

LAND COVER CLASSIFICATION  
USING MULTI-FREQUENCY SAR OVER  
SEMI-ARID AND FORESTED AFRICAN LANDSCAPES

Thesis submitted for the degree of Doctor of Philosophy  
at the University of Leicester

by

Bernardus Francois Spies  
Department of Geography  
University of Leicester

2016

Bernardus Francois Spies  
Land cover classification using multi-frequency SAR  
over semi-arid and forested African landscapes

**Abstract**

The potential of using multi-frequency Synthetic Aperture Radar (SAR) for land cover classification is becoming a reality, with multiple SAR satellites utilising different frequencies currently in orbit and more missions planned for the future. This study looks at combining SAR frequencies from L-band (ALOS PALSAR), C-band (ENVISAT ASAR) and X-band (TerraSAR-X) to find the optimum combination of the SAR data for land cover classification of forested and semi-arid ecoregions in Africa. The study site for forested areas is in Cameroon and the semi-arid study site is in Tanzania. Data from both the wet and dry seasons are available. Random forest models, with different combinations of input variables, are compared. Models with the top 30 variables are chosen from the mean decrease accuracy and mean decrease Gini variable importance measures, and compared with the classification accuracies using support vector machines. Some of the findings are that L-band is the best single frequency for land cover classifications for both ecoregions, with X-band the best single frequency if only forested regions are considered. Texture measures lead to an increase of between 15-25% overall classification accuracy compared to using only backscatter coefficients. The recommended dual-frequency combination are LX-bands, although L-band data give overall classification accuracies very close to LX-bands. The use of images from LCX data only marginally improves the classification accuracy from LX-images and L-band images. The benefit from acquisition of all three frequencies would therefore rarely outweigh the cost of acquiring and processing data from all three frequencies. The transferability of the random forest models to an additional geographic site did not produce satisfactory results, however the transferability of the random forest models to additional season data did give satisfactory results. The Kullback-Leibler divergence class difference measure showed potential to give an indication of transferability of the models, although refinement remains necessary.

## **Acknowledgements**

This research is funded by the European Commission, Marie Curie Programme, Initial Training Networks (EU-FP7-ITN), Grant Agreement number PITN-GA-2010-264509 and forms part of the GMES Initial Operations – Network for Earth Observation Research Training (GIONET). Data for ENVISAT ASAR, © ESA (2012) and ALOS PALSAR, © JAXA (2012), were obtained by proposal CP1-13326. TerraSAR-X data were obtained from the © DLR (2012) by proposal LAN1639 to utilize TSX archive data. To process most of the data used in this work, this research used the ALICE High Performance Computing Facility at the University of Leicester. I would like to thank the European Commission for the funding and for the great opportunity to be part of GIONET. Thanks to Heiko Balzter, from the University of Leicester Centre for Landscape and Climate Research, for the supervision and advice during the last four years and for initiating the GIONET research project in the first place. Thanks to Pete Fisher, who unfortunately passed away during the time of this project, for all the input and direction with regards to this research. Thanks to Virginia Nicolas-Perea and Rhiannon Harte-Chance for all the organisation of the project, especially the summer school every year during the duration of the project. Thanks to Alistair Lamb and Sarah Johnson for the supervision and guidance during the three years spent at Airbus Defence and Space, Geo-Information Division, in Leicester. Thanks to Andrew Anstee, Andrew Tewkesbury, Dave Turnbull, Tom Harling, Paul Brown and all the staff at the Airbus Defence and Space Leicester office. A special thanks to Helen Stannard for the administration during the project from Airbus Defence and Space. Thanks to Andreas Brink for the month spent at the European Commission Joint Research Centre in Ispra, Italy. Thanks to Elisabeth Schoepfer and Stefan Voigt for the two months spent at the DLR in Oberpfaffenhofen, Germany. Thanks to Matt Waldram for the introduction to bash scripting in Linux and for using his script as a starting point for processing many SAR images. Thanks to Andreas Wiesmann for guidance with the GAMMA software. Thanks to each of the fellow GIONET students: Dimitris, Stephanie, Linda, Christoff, Jessica, Penelope, Martyna, Maxim, Shailesh and Matthew with a special thanks to Sybrand, who was also based at Airbus Defence and Space, and to Pedro and James at the University of Leicester. Thanks for sharing this road and for the

friendships made that will long outlive the duration of the project. Lastly and by far not least, a big thanks, large indebtedness and deep appreciation to my wife Linda, and our daughter Bella who was born during this time, for all the tremendous amount of support during this research project.



## Table of contents

<b>ABSTRACT</b>	<b>II</b>
<b>ACKNOWLEDGEMENTS</b>	<b>III</b>
<b>TABLE OF CONTENTS</b>	<b>V</b>
<b>LIST OF TABLES</b>	<b>X</b>
<b>LIST OF FIGURES</b>	<b>XII</b>
<b>LIST OF ABBREVIATIONS</b>	<b>XXI</b>
<b>CHAPTER ONE INTRODUCTION</b>	<b>1</b>
1.1 OVERVIEW	1
1.2 HYPOTHESES FOR THE THESIS	8
1.3 THESIS OUTLINE	10
<b>CHAPTER TWO LITERATURE REVIEW ON MULTI-FREQUENCY SAR LAND COVER CLASSIFICATION</b>	<b>15</b>
2.1 SAR LAND COVER CLASSIFICATION	20
2.1.1 LAND COVER CLASSIFICATION SYSTEM (LCCS)	22
2.2 INTRODUCTION TO PRINCIPLES OF SYNTHETIC APERTURE RADAR (SAR)	24
2.3 KEY FACTORS THAT INFLUENCE SAR BACKSCATTER	29
2.3.1 TERRAIN GEOMETRY	29
2.3.2 SURFACE ROUGHNESS AND VEGETATION STRUCTURE	30
2.3.3 DIELECTRIC PROPERTIES / MOISTURE CONTENT	31
2.4 SAR SENSOR CHARACTERISTICS	32
2.4.1 ALOS PALSAR	32
2.4.2 ENVISAT ASAR	33
2.4.3 TERRASAR-X	34
2.5 MULTI-FREQUENCY SAR	35
2.6 CLASSIFICATION ALGORITHMS	37
2.6.1 RANDOM FORESTS	38
2.6.2 SUPPORT VECTOR MACHINES (SVM)	40
2.7 DISCUSSION OF LITERATURE REVIEW	44
<b>CHAPTER THREE STUDY SITES, DATA AND GENERAL METHODOLOGY</b>	<b>47</b>
3.1 STUDY SITES	47
3.1.1 CAMEROON	51
3.1.2 DRC	53
3.1.3 TANZANIA	56

3.1.4	CHAD	59
3.2	CHARACTERISTICS OF AVAILABLE IMAGES FOR EACH SITE	62
3.3	GENERAL METHODOLOGY FOR THE THREE EXPERIMENTS	65
3.4	SAMPLE SELECTION OF LAND COVER TYPES	69
<b>CHAPTER FOUR DATA PREPARATION</b>		<b>79</b>
4.1	ORTHORECTIFICATION OF IMAGES	81
4.2	CREATION OF IMAGE OVERLAP	86
4.3	CALCULATION OF DERIVED LAYERS	88
4.3.1	INTERCHANNEL RATIOS	88
4.3.2	TEXTURE MEASURES OF SAR IMAGES	90
4.3.3	ELEVATION DATA FROM DEM	100
4.4	COMBINING IMAGES INTO A LAYER STACK	100
<b>CHAPTER FIVE RANDOM FOREST AND SVM CLASSIFICATION OF MULTI-FREQUENCY SAR OVER CAMEROON AND TANZANIA STUDY SITES</b>		<b>103</b>
5.1	INTRODUCTION	104
5.2	SPECIFIC METHODOLOGY FOR DEVELOPING THE CLASSIFICATION ALGORITHMS	105
5.2.1	INPUT VARIABLE COMBINATION SCENARIOS	105
5.2.2	VARIABLE IMPORTANCE MEASURES FROM RANDOM FOREST MODELLING	107
5.2.2.1	MEAN DECREASE ACCURACY (MDA)	108
5.2.2.2	MEAN DECREASE GINI (MDG)	109
5.3	RESULTS	111
5.3.1	RANDOM FOREST CLASSIFICATION ACCURACY RESULTS	111
5.3.2	SELECTING VARIABLES BASED ON VARIABLE IMPORTANCE	117
5.3.2.1	VARIABLE IMPORTANCE OF ALL AVAILABLE VARIABLES	117
5.3.2.2	VARIABLE IMPORTANCE OF THE TOP 30 VARIABLES	129
5.3.3	MODEL BASED ON THE TOP 30 VARIABLES	135
5.3.4	CORRELATION BETWEEN TOP 30 VARIABLES FOR MDA AND MDG	138
5.3.5	SVM VS RANDOM FOREST CLASSIFICATION ACCURACIES	144
5.3.6	RANDOM FOREST MODEL APPLIED ON THE COMPLETE LAYER STACK OF IMAGES	145
5.3.7	SUMMARY OF RESULTS	148
5.4	DISCUSSION	152
5.5	CONCLUSION	155
<b>CHAPTER SIX TRANSFERABILITY OF RANDOM FOREST CLASSIFICATION MODELS ON ADDITIONAL STUDY SITES IN THE DRC AND CHAD</b>		<b>157</b>
6.1	INTRODUCTION	157
6.2	SPECIFIC METHODOLOGY ON TRANSFERABILITY AND GENERALIZATION OF CLASSIFICATION ALGORITHMS	158

6.3	RESULTS	162
6.3.1	TRANSFERABILITY OF CAMEROON WET SEASON RANDOM FOREST MODELS	162
6.3.1.1	CAMEROON WET SEASON MODELS APPLIED TO CAMEROON DRY SEASON DATA	162
6.3.1.2	CAMEROON WET SEASON MODELS APPLIED TO DRC WET SEASON DATA	164
6.3.2	TRANSFERABILITY OF TANZANIA DRY SEASON RANDOM FOREST MODELS	166
6.3.2.1	TANZANIA DRY SEASON RANDOM FOREST MODEL APPLIED TO TANZANIA WET SEASON DATA	167
6.3.2.2	TANZANIA DRY SEASON RANDOM FOREST MODEL APPLIED TO CHAD DRY SEASON DATA	169
6.4	SUMMARY OF RESULTS	171
6.5	DISCUSSION	178
6.5.1	GROUPS OF INCREASING VARIABLE COMPLEXITY	178
6.5.1.1	BACKSCATTER COEFFICIENTS	179
6.5.1.2	INTERCHANNEL RATIOS	180
6.5.1.3	ELEVATION DATA	180
6.5.1.4	TEXTURE MEASURES	181
6.5.2	SINGLE, DOUBLE OR THREE FREQUENCIES	183
6.5.3	FORESTED VS SEMI-ARID TRANSFERABILITY COMPARISON	185
6.6	CONCLUSION	187
<b>CHAPTER SEVEN SAR SIGNATURES AND KULLBACK-LEIBLER DIVERGENCE MEASURES OF MULTI-FREQUENCY SAR OVER SEMI-ARID AND FORESTED SITES</b>		<b>189</b>
7.1	INTRODUCTION	190
7.1.1	CAMEROON WET SEASON TRAINING VS TEST SAMPLES	192
7.1.2	TANZANIA DRY SEASON TRAINING VS TEST SAMPLES	198
7.2	SPECIFIC METHODOLOGY FOR SAR SIGNATURE EXTRACTION	202
7.3	RESULTS	204
7.3.1	VISUAL COMPARISON OF SAR SIGNATURES FOR THE FORESTED SITES (CAMEROON AND DRC)	204
7.3.1.1	CAMEROON WET VS DRY SEASON SAMPLES	204
7.3.1.2	CAMEROON VS DRC WET SEASON SAMPLES	208
7.3.2	VISUAL COMPARISON OF SAR SIGNATURES FOR THE SEMI-ARID SITES (TANZANIA AND CHAD)	212
7.3.2.1	TANZANIA DRY VS WET SEASON SAMPLES	213
7.3.2.2	TANZANIA VS CHAD DRY SEASON SAMPLES	217
7.3.3	KLD FOR LAND COVER CLASSES OF THE FORESTED ECOREGION (CAMEROON AND DRC)	221
7.3.3.1	KLD FOR CAMEROON WET TRAINING VS TEST SAMPLES	223
7.3.3.2	KLD FOR CAMEROON WET VS DRY SEASON SAMPLES	225
7.3.3.3	KLD FOR CAMEROON VS DRC WET SEASON SAMPLES	226
7.3.4	KLD FOR LAND COVER CLASSES OF THE SEMI-ARID ECOREGION (TANZANIA AND CHAD)	228
7.3.4.1	KLD FOR TANZANIA DRY SEASON TRAINING VS TEST SAMPLES	229
7.3.4.2	KLD FOR TANZANIA DRY VS WET SEASON SAMPLES	230

7.3.4.3	KLD FOR TANZANIA VS CHAD DRY SEASON SAMPLES	232
7.3.5	KLD CLASS DIFFERENCE VS RANDOM FOREST CLASSIFICATION USER'S ACCURACY	233
7.4	DISCUSSION	238
7.5	CONCLUSION	242
<b>CHAPTER EIGHT</b>	<b>OVERALL DISCUSSION AND CONCLUSION</b>	<b>243</b>
8.1	OVERALL DISCUSSION	244
8.1.1	FINDINGS FROM THE RANDOM FOREST AND SVM CLASSIFICATION OF MULTI-FREQUENCY SAR OVER CAMEROON AND TANZANIA STUDY SITES	244
8.1.1.1	SINGLE VS DUAL VS THREE FREQUENCIES	244
8.1.1.2	BACKSCATTER COEFFICIENTS ONLY	246
8.1.1.3	INTERCHANNEL RATIOS	246
8.1.1.4	TEXTURE MEASURES	247
8.1.1.5	ELEVATION DATA	247
8.1.1.6	DRY SEASON VS WET SEASON	248
8.1.1.7	FORESTED VS SEMI-ARID	249
8.1.1.8	MDA VS MDG	250
8.1.1.9	RANDOM FOREST VS SVM	250
8.1.2	FINDINGS FROM ASSESSING THE TRANSFERABILITY OF THE RANDOM FOREST CLASSIFICATION MODELS TO ADDITIONAL TEST SITES	250
8.1.3	FINDINGS FROM THE SAR SIGNATURES AND KLD MEASURES OF MULTI-FREQUENCY SAR IMAGES	252
8.2	PRACTICAL IMPLICATIONS OF A MULTI-FREQUENCY SAR APPROACH	253
8.3	CONCLUSION, FURTHER WORK IDENTIFIED AND FUTURE VIEW	254
8.3.1	FINAL CONCLUSION	254
8.3.2	FURTHER WORK IDENTIFIED	256
8.3.3	FUTURE VIEW	257
<b>APPENDICES</b>		<b>258</b>
<b>APPENDIX ONE: GAMMA ORTHORECTIFICATION SCRIPTS</b>		<b>258</b>
A1.1	OVERVIEW OF GAMMA ORTHORECTIFICATION SCRIPTS	258
A1.2	DEM PREPARATION STEPS – LANDSAT	261
A1.3	DEM PREPARATION – SRTM	261
A1.4	GAMMA ORTHORECTIFICATION SCRIPTS	263
A1.4.1	/PHD FOLDER SCRIPTS	263
A1.4.2	/DEM FOLDER SCRIPTS	297
A1.4.3	/DEM/LANDSAT FOLDER SCRIPTS	299
A1.4.4	/DEM/LANDSAT/LANDSAT_PREPARATION FOLDER	303
A1.4.5	/DEM/PALSAR	304

A1.4.6	/DEM/SRTM	304
A1.4.7	/DEM/SRTM/SRTM_PREPARATION FOLDER	307
A1.5	ADDITIONAL LINUX COMMANDS AND NOTES	308
<b>APPENDIX TWO: GDAL SCRIPTS TO CREATE IMAGE OVERLAP</b>		<b>310</b>
A2.1	OVERVIEW OF GDAL SCRIPTS	310
A2.2	GDAL BASH SCRIPTS	313
A2.2.1	UNFOLD MULTIPLE FOLDERS INTO A SINGLE FOLDER	313
A2.2.2	CREATING AN OVERLAP BETWEEN MULTIPLE SHAPEFILES SCRIPT	314
A2.2.3	ELEVATION DATA PREPARATION	323
A2.2.4	CALCULATE ADDITIONAL FEATURES LAYERS PROCESS FLOW SCRIPT	329
<b>R MODELLING AND GRAPHING SCRIPTS</b>		<b>386</b>
A3.1	OVERVIEW OF R MODELLING AND GRAPHING SCRIPTS	386
A3.2	R RANDOM FOREST MODELLING SCRIPTS	389
A3.2.1.	R DATA PREPARATION FOR MODELLING SCRIPTS	389
A3.2.2.	R FUNCTIONS FOR MODELLING	399
A3.2.3.	R FUNCTIONS FOR PLOTTING GRAPHS	406
<b>BIBLIOGRAPHY</b>		<b>430</b>

## List of tables

TABLE 2.1 THE SIX LEVELS OF THE MODULAR-HIERARCHIVAL PHASE OF THE NATURAL AND SEMI-NATURAL TERRESTRIAL VEGETATION DICHOTOMOUS CLASS OF LCCS.	24
TABLE 2.2 RANGE OF RADAR FREQUENCY BANDS WITH THEIR CORRESPONDING FREQUENCY AND WAVELENGTH RANGES (LEWIS ET AL. 1998; BARRETT ET AL. 2009).	25
TABLE 2.3 SAR SATELLITES IN ORBIT, APPROVED AND PLANNED FOR 2011-2022, FOR POTENTIAL SAR APPLICATIONS.	28
TABLE 3.1 LIST OF STUDY SITE LOCATIONS WITH CORRESPONDING UTM ZONES AND EPSG CODES.	50
TABLE 3.2 THE AVAILABLE OPTICAL IMAGES FOR THE FOUR STUDY SITES, WITH RESPECTIVE DATES AND SEASONS FOR EACH SITE.	61
TABLE 3.3 THE AVAILABLE IMAGES FOR EACH STUDY SITE, WITH COLOURS SHOWING IMAGES TO BE USED FOR AN INTER-SEASONAL COMPARISON BETWEEN THE SEMI-ARID AND FORESTED STUDY SITES.	63
TABLE 3.4 THE AVAILABLE IMAGES FOR EACH STUDY SITE, WITH COLOURS SHOWING THE AVAILABLE IMAGES USED FOR TESTING THE TRANSFERABILITY OF THE CLASSIFICATION MODEL TO AN ADDITIONAL STUDY SITE FOR BOTH THE FORESTED AND SEMI-ARID AREAS.	63
TABLE 3.5 THE LAND COVER TYPES IDENTIFIED IN THE CAMEROON AND TANZANIA STUDY SITES, WITH THEIR WOODLAND COVER ESTIMATION AND ADDITIONAL DESCRIPTION.	71
TABLE 3.6 NUMBER OF TRAINING AND TEST SAMPLES AVAILABLE, TRAINING AND TEST SAMPLE PERCENTAGE SPLIT, AND 3000 SAMPLES AS A PERCENTAGE OF THE AVAILABLE LAND COVER SAMPLES, FOR THE CAMEROON STUDY SITE.	77
TABLE 3.7 NUMBER OF TRAINING AND TEST SAMPLES AVAILABLE, TRAINING AND TEST SAMPLE PERCENTAGE SPLIT, AND 3000 SAMPLES AS A PERCENTAGE OF THE AVAILABLE LAND COVER SAMPLES, FOR THE TANZANIA STUDY SITE.	78
TABLE 4.1 THE IMAGES USED IN THIS RESEARCH WITH THE FINAL MODEL FIT STANDARD DEVIATION IN THE RANGE AND AZIMUTH DIRECTIONS.	83
TABLE 4.2 MULTI-LOOKING APPLIED IN THE RANGE AND AZIMUTH DIRECTION FOR EACH SENSOR TO ACHIEVE THE LISTED SPATIAL RESOLUTIONS.	85
TABLE 4.3 THE INTERCHANNEL RATIOS, SUMS AND DIFFERENCES USED IN THIS STUDY, EXTENDED FROM LÖNNQVIST ET AL. (2010).	89
TABLE 4.4 SUMMARY OF TEXTURE MEASURES USED IN OTHER STUDIES FOR THE CLASSIFICATION OF SAR IMAGES.	92
TABLE 4.5 THE FOOTPRINT AREA COVERED IN METERS BY THE TEXTURE CALCULATION, FOR EACH SENSOR, AT DIFFERENT WINDOW SIZES.	94
TABLE 5.1 LIST OF SCENARIOS INVESTIGATED FOR MODEL INPUT COMPARISON.	106
TABLE 5.2 OVERALL CLASSIFICATION ACCURACIES OF RANDOM FOREST MODELS USING (A) SINGLE FREQUENCIES AND (B) DUAL / THREE FREQUENCIES, WITH DIFFERENT VARIABLE COMBINATIONS FOR THE FORESTED AND SEMI-ARID STUDY SITES.	150
TABLE 5.3 THE RECOMMENDED SAR FREQUENCIES WITH HIGH CLASSIFICATION ACCURACIES FOR LAND COVER CLASSIFICATION WITHIN SPECIFIC ECOREGIONS. THIS ASSUMES ALL THE AVAILABLE VARIABLES OF BACKSCATTER COEFFICIENTS, INTERCHANNEL RATIOS, TEXTURE MEASURES AND ELEVATION DATA ARE INCLUDED IN THE MODELS.	152

TABLE 6.1 THE TRAINING DATA, TEST DATA AND DIFFERENT LAND COVER CLASS SCENARIOS FOR THE FORESTED SITES.	160
TABLE 6.2 THE TRAINING DATA, TEST DATA AND DIFFERENT LAND COVER CLASS SCENARIOS FOR THE SEMI-ARID SITES.	161
TABLE 6.3 SUMMARY TABLE OF THE OVERALL CLASSIFICATION ACCURACIES FROM THE RANDOM FOREST MODELS, WHICH TEST THE TRANSFERABILITY OF THE FORESTED SITES. THE (A) SINGLE FREQUENCY AND (B) DUAL / THREE FREQUENCY COMBINATIONS ARE SHOWN.	173
TABLE 6.4 SUMMARY TABLE OF THE OVERALL CLASSIFICATION ACCURACIES FROM THE RANDOM FOREST MODELS, WHICH TEST THE TRANSFERABILITY OF THE SEMI-ARID SITES. THE (A) SINGLE FREQUENCY AND (B) DUAL / THREE FREQUENCY COMBINATIONS ARE SHOWN.	174
TABLE 6.5 SUMMARY OF RECOMMENDED FREQUENCY AND VARIABLE RECOMMENDATIONS FOR TRANSFERRING RANDOM FOREST MODELS TO DIFFERENT SEASON AND ADDITIONAL SITE SCENARIOS. OVERALL CLASSIFICATION ACCURACIES ARE GIVEN IN BRACKETS FOR THE RESPECTIVE SCENARIOS.	184
TABLE 7.1 SCENARIOS FOR ASSESSING THE SEPARABILITY BETWEEN LAND COVER CLASSES.	191
TABLE 7.2 LIST OF LAND COVER SAMPLE DATA DISTRIBUTION COMPARISONS, BOTH VISUAL AND BY THE KLD, FOR THE FORESTED AND SEMI-ARID SITES.	203
TABLE 7.3 THE SITE COMPARISONS FOR THE FORESTED SITES WITH CORRESPONDING FIGURE NUMBERS.	204
TABLE 7.4 THE SITE COMPARISONS FOR THE SEMI-ARID SITES WITH CORRESPONDING FIGURE NUMBERS.	213
TABLE 7.5 THE SITE COMPARISON USING KLD OF THE FORESTED SITES WITH CORRESPONDING FIGURE NUMBERS.	222
TABLE 7.6 THE SITE COMPARISON USING KLD FOR THE SEMI-ARID SITES WITH CORRESPONDING FIGURE NUMBERS.	228
TABLE 7.7 A CONFUSION MATRIX BETWEEN REFERENCE VALUES AND PREDICTED VALUES.	235
TABLE A1.1 THE FALSE EASTING AND FALSE NORTHING VALUES FOR THE NORTHERN AND SOUTHERN HEMISPHERE.	300
TABLE A2.1 STRUCTURE OF BASH SCRIPTS FOR DATA PREPARATION.	312
TABLE A3.1 R DATA PREPARATION FOR MODELLING SCRIPTS (SECTION A3.3.1)	386
TABLE A3.2 R FUNCTIONS FOR MODELLING (SECTION A3.3.2)	387
TABLE A3.3 R FUNCTIONS FOR PLOTTING GRAPHS (SECTION A3.3.3)	388

## List of figures

FIGURE 1.1 AVAILABILITY OF SAR DATA FROM ALOS PALSAR, ENVISAT ASAR AND TERRASAR-X FOR THE FULL YEAR OF 2010 OVER THE AFRICAN CONTINENT.	7
FIGURE 1.2 SCHEMATIC OVERVIEW OF THIS RESEARCH IN TERMS OF THE CROSS INDUSTRY STANDARD PROCESS FOR DATA MINING.	12
FIGURE 2.1 THE LCCS EIGHT BROAD LAND COVER CLASSES FROM THE DICHOTOMOUS PHASE OF THE LCCS LAND COVER CLASSIFICATION.	23
FIGURE 2.2 ACQUISITION GEOMETRY OF A SATELLITE SAR SYSTEM, AMENDED FROM LEWIS ET AL. (1998).	27
FIGURE 2.3 THE COMPARISON BETWEEN THE INCIDENCE ANGLE $\theta$ AND THE LOCAL INCIDENCE ANGLE $\theta_l$ .	30
FIGURE 2.4 SCATTERING MECHANISMS AFFECTING THE SAR SIGNAL: VOLUME, SURFACE AND DOUBLE-BOUNCE SCATTERING.	30
FIGURE 2.5 SCHEMATIC DIAGRAM OF THE RANDOM FOREST CLASSIFIER.	39
FIGURE 2.6 CLASSIFICATION OF AN SVM, FOR LINEARLY SEPARABLE CLASSES.	41
FIGURE 2.7 AN SVM MAPPED TO A HIGHER DIMENSION USING KERNEL $K()$ FOR NON-LINEARLY SEPARABLE CLASSES.	42
FIGURE 3.1 THE LOCATIONS OF THE STUDY SITES, IN THE CONTEXT OF THE BIOME MAP OF AFRICA (WHITE, 1983B).	48
FIGURE 3.2 AVAILABLE DATA FROM THE SATELLITE ARCHIVES FOR THE CAMEROON SITE, WITH SELECTED IMAGES IN RED FOR THE DRY SEASON AND IN BLUE FOR THE RAINY WET SEASON. THE OPTICAL IMAGES USED IN THIS STUDY IS SHOWN IN DARK BROWN SQUARES (LANDSAT 8) AND A LIGHT BROWN SQUARE, © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE.	51
FIGURE 3.3 A) OVERLAP BETWEEN DIFFERENT SAR SENSORS FOR THE CAMEROON STUDY SITE. B) GOOGLE EARTH VIEW OF THE OVERLAPPING AREA AND WIDER SURROUNDS FOR THE CAMEROON STUDY SITE. © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE, IMAGE © 2015 CNES /ASTRIUM, © 2015 CNES /SPOT IMAGE.	52
FIGURE 3.4 THE SRTM ELEVATION, LANDSAT (DRY AND WET) AND MULTI-FREQUENCY SAR (DRY AND WET) IMAGES OF THE CAMEROON SITE.	53
FIGURE 3.5 AVAILABLE DATA FROM THE ARCHIVES FOR THE DRC SITE, WITH SELECTED IMAGES IN DARK BLUE FOR THE WET SEASON. © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE.	54
FIGURE 3.6 A) OVERLAP AREA FROM THE DIFFERENT SAR SENSORS FOR THE DRC SITE. B) GOOGLE EARTH VIEW OF THE OVERLAPPING AREA AND WIDER SURROUNDINGS FOR THE DRC STUDY SITE. © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE, IMAGE LANDSAT.	55
FIGURE 3.7 SRTM ELEVATION, LANDSAT 8 AND MULTI-FREQUENCY SAR IMAGES OF THE DRC STUDY SITE.	56
FIGURE 3.8 AVAILABLE DATA FROM THE ARCHIVES FOR THE TANZANIA SITE, WITH SELECTED IMAGES IN RED FOR THE DRY SEASON AND IN BLUE FOR THE RAINY WET SEASON. © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE.	57
FIGURE 3.9 A) OVERLAP AREA FROM THE DIFFERENT SAR SENSORS FOR THE TANZANIA STUDY SITE. B) GOOGLE EARTH VIEW OF THE OVERLAPPING AREA AND WIDER SURROUNDINGS FOR THE TANZANIA SEMI-ARID STUDY SITE. © 2015 GOOGLE, IMAGE ©2015 DIGITALGLOBE, ©2015 CNES/SPOT IMAGE, IMAGE © 2015 CNES / ASTRIUM.	58



FIGURE 3.10 SRTM ELEVATION, LANDSAT (DRY AND WET SEASON IMAGES) AND MULTI-FREQUENCY SAR (DRY AND WET SEASON IMAGES) FOR THE TANZANIA SITE.	58
FIGURE 3.11 AVAILABLE DATA FROM THE ARCHIVES FOR THE CHAD SITE, WITH SELECTED IMAGES IN RED FOR THE DRY IMAGES. © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE.	59
FIGURE 3.12 A) OVERLAP AREA FROM THE DIFFERENT SAR SENSORS FOR THE CHAD SITE. B) GOOGLE EARTH VIEW OF THE OVERLAPPING AREA AND WIDER SURROUNDINGS FOR THE CHAD SEMI-ARID STUDY SITE. © 2015 GOOGLE, © 2015 CNES/SPOT IMAGE, IMAGE © 2015 DIGITALGLOBE, IMAGE © 2015 CNES / ASTRIUM.	60
FIGURE 3.13 SRTM ELEVATION, LANDSAT AND MULTI-FREQUENCY SAR FOR THE CHAD STUDY SITE.	60
FIGURE 3.14 THE INTER-SEASON AND INTER-SITE COMPARISON BETWEEN THE MAIN AND ADDITIONAL SITES FOR THE FORESTED AND SEMI-ARID ECOREGIONS.	66
FIGURE 3.15 A) CAMEROON STUDY SITE WITH LETTERS A-F REFERRING TO THE LAND COVER CLASSES SHOW IN FIGURE 3.16. B) TANZANIA STUDY SITE WITH LETTERS A-F REFERRING TO THE LAND COVER CLASSES IN FIGURE 3.17. © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE.	72
FIGURE 3.16 SCREENSHOTS FROM GOOGLE EARTH OF SELECTED SAMPLE AREAS OF IDENTIFIED LAND COVER CLASSES FOR THE CAMEROON STUDY SITE. EACH IMAGE IS 1KM X 1KM IN SIZE. © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE.	73
FIGURE 3.17 SCREENSHOTS FROM GOOGLE EARTH OF SELECTED SAMPLE AREAS OF IDENTIFIED LAND COVER CLASSES FOR THE TANZANIA STUDY SITE. EACH IMAGE IS 1KM X 1KM IN SIZE. © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE.	74
FIGURE 3.18 SAMPLE SELECTION PROCESS FOR THE CAMEROON AND TANZANIA SITES. © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE.	75
FIGURE 4.1 OVERVIEW OF THE PRE-PROCESSING STEPS, LEADING UP TO THE MODEL DEVELOPMENT. NUMBERS CORRESPOND TO CHAPTER AND SECTION NUMBERS.	80
FIGURE 4.2 FIRST STEP OF THE PRE-PROCESSING WORKFLOW, TO CREATE ORTHORECTIFIED IMAGES.	84
FIGURE 4.3 SPATIAL RESOLUTION COMPARISON FOR THE MULTI-LOOKED AND SPATIALLY RESAMPLED IMAGES USED IN THIS STUDY, FROM ENVISAT ASAR, ALOS PALSAR AND TERRASAR-X.	85
FIGURE 4.4 SECOND STEP OF THE PRE-PROCESSING WORKFLOW, CREATING OVERLAP AREAS FROM THE SAR IMAGES FOR EACH SITE, CREATING A LAYER STACK AND CONVERTING FROM LINEAR TO DB SCALE.	87
FIGURE 4.5 COMPARISON OF THE FOOTPRINTS OF THE DIFFERENT WINDOW-SIZES USED FOR THE TEXTURE CALCULATIONS FOR TERRASAR-X, ALOS PALSAR AND ENVISAR ASAR.	95
FIGURE 4.6 TEXTURE MEASURES AT VARIOUS WINDOW SIZES FOR ALOS PALSAR (L-HH), HIGHLIGHTING DENSE TREES AND BARE SOIL.	96
FIGURE 4.7 TEXTURE MEASURES AT VARIOUS WINDOW SIZES FOR TERRASAR-X (X-VV), HIGHLIGHTING DENSE TREES AND BARE SOIL.	97
FIGURE 4.8 TEXTURE MEASURES AT VARIOUS WINDOW SIZES FOR ALOS PALSAR (L-HH), HIGHLIGHTING SETTLEMENTS.	98
FIGURE 4.9 TEXTURE MEASURES AT VARIOUS WINDOW SIZES FOR TERRASAR-X (X-VV), HIGHLIGHTING SETTLEMENTS.	99
FIGURE 4.10 LAYER STACK CREATION AND ADDITIONAL LAYER CALCULATION OF THE CAMEROON AND TANZANIA SAR IMAGES.	102

- FIGURE 5.1 RANDOM FOREST CLASSIFICATION ACCURACIES USING ALL VARIABLES FOR DIFFERENT FREQUENCY COMBINATIONS ( $\sigma 0$ ) FOR THE CAMEROON DRY SEASON STUDY SITE (TOP LEFT), THE CAMEROON WET SEASON STUDY SITE (TOP RIGHT), THE TANZANIA DRY SEASON STUDY SITE (BOTTOM LEFT) AND THE TANZANIA WET SEASON STUDY SITE (BOTTOM RIGHT). 112
- FIGURE 5.2 RANDOM FOREST CLASSIFICATION ACCURACIES USING BACKSCATTER COEFFICIENTS ONLY FOR DIFFERENT FREQUENCY COMBINATIONS ( $\sigma 0$ ) FOR THE CAMEROON DRY SEASON STUDY SITE (TOP LEFT), THE CAMEROON WET SEASON STUDY SITE (TOP RIGHT), THE TANZANIA DRY SEASON STUDY SITE (BOTTOM LEFT) AND THE TANZANIA WET SEASON STUDY SITE (BOTTOM RIGHT). 112
- FIGURE 5.3 RANDOM FOREST CLASSIFICATION ACCURACIES USING ONLY INTERCHANNEL RATIOS FOR DIFFERENT FREQUENCY COMBINATIONS ( $\sigma 0$ ) FOR THE CAMEROON DRY STUDY SITE (TOP LEFT), THE CAMEROON WET STUDY SITE (TOP RIGHT), THE TANZANIA DRY STUDY SITE (BOTTOM LEFT) AND THE TANZANIA WET STUDY SITE (BOTTOM RIGHT). 114
- FIGURE 5.4 RANDOM FOREST CLASSIFICATION ACCURACIES USING TEXTURE MEASURES ONLY FOR DIFFERENT FREQUENCY COMBINATIONS ( $\sigma 0$ ) FOR THE CAMEROON DRY STUDY SITE (TOP LEFT), THE CAMEROON WET STUDY SITE (TOP RIGHT), THE TANZANIA DRY STUDY SITE (BOTTOM LEFT) AND THE TANZANIA WET STUDY SITE (BOTTOM RIGHT). 114
- FIGURE 5.5 RANDOM FOREST CLASSIFICATION ACCURACIES USING ONLY ELEVATION DATA FROM SRTM (ELEVATION, SLOPE AND ASPECT) FOR THE CAMEROON DRY STUDY SITE (TOP LEFT), THE CAMEROON WET STUDY SITE (TOP RIGHT), THE TANZANIA DRY STUDY SITE (BOTTOM LEFT) AND THE TANZANIA WET STUDY SITE (BOTTOM RIGHT). 115
- FIGURE 5.6 RANDOM FOREST CLASSIFICATION ACCURACIES USING BACKSCATTER COEFFICIENTS AND INTERCHANNEL RATIOS FOR DIFFERENT FREQUENCY COMBINATIONS ( $\sigma 0$ ) FOR THE CAMEROON DRY STUDY SITE (TOP LEFT), THE CAMEROON WET STUDY SITE (TOP RIGHT), THE TANZANIA DRY STUDY SITE (BOTTOM LEFT) AND THE TANZANIA WET STUDY SITE (BOTTOM RIGHT). 115
- FIGURE 5.7 RANDOM FOREST CLASSIFICATION ACCURACIES USING BACKSCATTER COEFFICIENTS AND ELEVATION DATA FOR DIFFERENT FREQUENCY COMBINATIONS ( $\sigma 0$ ) FOR THE CAMEROON DRY STUDY SITE (TOP LEFT), THE CAMEROON WET STUDY SITE (TOP RIGHT), THE TANZANIA DRY STUDY SITE (BOTTOM LEFT) AND THE TANZANIA WET STUDY SITE (BOTTOM RIGHT). 116
- FIGURE 5.8 RANDOM FOREST CLASSIFICATION ACCURACIES USING BACKSCATTER COEFFICIENTS AND TEXTURE MEASURES FOR DIFFERENT FREQUENCY COMBINATIONS ( $\sigma 0$ ) FOR THE CAMEROON DRY STUDY SITE (TOP LEFT), THE CAMEROON WET STUDY SITE (TOP RIGHT), THE TANZANIA DRY STUDY SITE (BOTTOM LEFT) AND THE TANZANIA WET STUDY SITE (BOTTOM RIGHT). 116
- FIGURE 5.9 RANDOM FOREST CLASSIFICATION ACCURACIES USING BACKSCATTER COEFFICIENTS, INTERCHANNEL CALCULATIONS AND TEXTURE MEASURES FOR DIFFERENT FREQUENCY COMBINATIONS ( $\sigma 0$ ) FOR THE CAMEROON DRY STUDY SITE (TOP LEFT), THE CAMEROON WET STUDY SITE (TOP RIGHT), THE TANZANIA DRY STUDY SITE (BOTTOM LEFT) AND THE TANZANIA WET STUDY SITE (BOTTOM RIGHT). 117
- FIGURE 5.10 THE MDA AND MDG VARIABLE IMPORTANCE MEASURES FOR CAMEROON DRY SEASON (TOP LEFT), CAMEROON WET SEASON (TOP RIGHT), TANZANIA DRY SEASON (BOTTOM LEFT) AND TANZANIA WET SEASON (BOTTOM RIGHT), GROUPED BY VARIABLE TYPE (BACKSCATTER COEFFICIENTS, INTERCHANNEL RATIOS, TEXTURE MEASURES AND

ELEVATION DATA). THE NUMBER OF VARIABLES FOR EACH GROUPED ARE SHOWN IN BRACKETS TO THE LEFT OF EACH GROUP.	120
FIGURE 5.11 THE ORDER OF VARIABLE IMPORTANCE BASED ON MDA AND MDG USING ONLY BACKSCATTER COEFFICIENTS, FOR CAMEROON DRY SEASON (TOP LEFT), CAMEROON WET SEASON (TOP RIGHT), TANZANIA DRY SEASON (BOTTOM LEFT) AND TANZANIA WET SEASON STUDY SITES (BOTTOM RIGHT). THE NUMBER OF VARIABLES FOR EACH GROUPED ARE SHOWN IN BRACKETS TO THE LEFT OF EACH GROUP.	121
FIGURE 5.12 VARIABLE IMPORTANCE (MDA AND MDG) OF ONLY INTERCHANNEL RATIOS, FOR EACH OF THE SITE / SEASON SCENARIOS.	122
FIGURE 5.13 MDA AND MDG VARIABLE IMPORTANCE OF TEXTURE MEASURES, FOR X-HH, C-VV AND L-HH FREQUENCY-POLARISATION COMBINATIONS FOR THE CAMEROON DRY STUDY SITE.	123
FIGURE 5.14 MDA AND MDG VARIABLE IMPORTANCE OF TEXTURE MEASURES, FOR X-HH, C-VV AND L-HV FREQUENCY-POLARISATION COMBINATIONS FOR THE CAMEROON WET STUDY SITE.	124
FIGURE 5.15 MDA AND MDG VARIABLE IMPORTANCE OF TEXTURE MEASURES, FOR X-VH, C-VV AND L-HV FREQUENCY-POLARISATION COMBINATIONS FOR THE TANZANIA DRY STUDY SITE.	125
FIGURE 5.16 MDA AND MDG VARIABLE IMPORTANCE OF TEXTURE MEASURES, FOR X-VH, C-VH AND L-HV FREQUENCY-POLARISATION COMBINATIONS FOR THE TANZANIA WET STUDY SITE.	126
FIGURE 5.17 MDA AND MDG VARIABLE IMPORTANCE FOR ELEVATION, SLOPE AND ASPECT FOR THE CAMEROON DRY (TOP LEFT), CAMEROON WET (TOP RIGHT), TANZANIA DRY (BOTTOM LEFT) AND TANZANIA WET (BOTTOM RIGHT) STUDY SITES.	128
FIGURE 5.18 TOP 30 VARIABLES BASED ON MDA AND MDG FOR THE CD SITE RANDOM FOREST MODEL. THE TOP TWO GRAPHS SHOW VARIABLE 1-30 IN RANKING ORDER, WHEREAS THE BOTTOM TWO GRAPHS ARE GROUPED BY THE ORIGINAL FREQUENCY-POLARISATION CHANNELS.	131
FIGURE 5.19 TOP 30 VARIABLES BASED ON MDA AND MDG FOR THE CW SITE RANDOM FOREST MODEL. THE TOP TWO GRAPHS SHOW VARIABLE 1-30 IN RANKING ORDER, WHEREAS THE BOTTOM TWO GRAPHS ARE GROUPED BY THE ORIGINAL FREQUENCY-POLARISATION CHANNELS.	132
FIGURE 5.20 TOP 30 VARIABLES BASED ON MDA AND MDG FOR THE TD SITE RANDOM FOREST MODEL. THE TOP TWO GRAPHS SHOW VARIABLE 1-30 IN RANKING ORDER, WHEREAS THE BOTTOM TWO GRAPHS ARE GROUPED BY THE ORIGINAL FREQUENCY-POLARISATION CHANNELS.	133
FIGURE 5.21 TOP 30 VARIABLES BASED ON MDA AND MDG FOR THE TW SITE RANDOM FOREST MODEL. THE TOP TWO GRAPHS SHOW VARIABLE 1-30 IN RANKING ORDER, WHEREAS THE BOTTOM TWO GRAPHS ARE GROUPED BY THE FREQUENCY-POLARISATION GROUPS.	134
FIGURE 5.22 THE OVERALL AND CLASS-SPECIFIC CLASSIFICATION ACCURACIES FOR THE TOP 30 VARIABLES BASED ON THE MDA VARIABLE IMPORTANCE MEASURE (CONSIDERING BACKSCATTER COEFFICIENTS, INTERCHANNEL RATIOS AND TEXTURE MEASURES) FOR EACH SITE / SEASON SCENARIO.	136
FIGURE 5.23 THE OVERALL AND CLASS-SPECIFIC CLASSIFICATION ACCURACIES FOR THE TOP 30 VARIABLES BASED ON MDG VARIABLE IMPORTANCE MEASURE (CONSIDERING BACKSCATTER COEFFICIENTS, INTERCHANNEL RATIOS AND TEXTURE MEASURES) FOR EACH SITE / SEASON SCENARIO.	137

FIGURE 5.24 CORRELATION PLOTS FOR THE TOP 30 VARIABLES BASED ON MDA AND MDG FOR THE CAMEROON DRY STUDY SITE. ORDERING OF THE VARIABLES ARE BASED ON HIERARCHICAL CLUSTERING.	140
FIGURE 5.25 CORRELATION PLOTS FOR THE TOP 30 VARIABLES BASED ON MDA AND MDG FOR THE CAMEROON WET STUDY SITE. ORDERING OF THE VARIABLES ARE BASED ON HIERARCHICAL CLUSTERING.	141
FIGURE 5.26 CORRELATION PLOTS FOR THE TOP 30 VARIABLES BASED ON MDA AND MDG FOR THE TANZANIA DRY STUDY SITE. ORDERING OF THE VARIABLES ARE BASED ON HIERARCHICAL CLUSTERING.	142
FIGURE 5.27 CORRELATION PLOTS FOR THE TOP 30 VARIABLES BASED ON MDA AND MDG FOR THE TANZANIA WET STUDY SITE. ORDERING OF THE VARIABLES ARE BASED ON HIERARCHICAL CLUSTERING.	143
FIGURE 5.28 OVERALL AND CLASS-SPECIFIC CLASSIFICATION ACCURACIES FOR THE SUPPORT VECTOR MACHINE CLASSIFICATION BASED ON THE TOP 1 TO 30 VARIABLES SELECTED BY THE MDA VARIABLE IMPORTANCE MEASURE IN THE RANDOM FOREST CLASSIFICATION.	144
FIGURE 5.29 OVERALL AND CLASS-SPECIFIC CLASSIFICATION ACCURACIES FOR THE SUPPORT VECTOR MACHINE CLASSIFICATION BASED ON THE TOP 1 TO 30 VARIABLES SELECTED BY THE MDG VARIABLE IMPORTANCE MEASURE IN THE RANDOM FOREST CLASSIFICATION.	145
FIGURE 5.30 THE CAMEROON STUDY SITE OPTICAL IMAGE AVAILABLE ON GOOGLE EARTH ON THE LEFT COMPARED THE RANDOM FOREST CLASSIFICATION OF THE WHOLE IMAGE USING THE CAMEROON DRY TOP 30 MDA VARIABLES ON THE RIGHT. © 2015 GOOGLE, IMAGE © 2015 DIGITALGLOBE, IMAGE © 2015 CNES /ASTRIUM.	146
FIGURE 5.31 ZOOMED IN SECTION OF THE RANDOM FOREST CLASSIFICATION BASED ON THE MDA TOP 30 VARIABLES (TOP), COMPARED TO THE VERY HIGH RESOLUTION IMAGE OF THE SAME AREA OF THE CAMEROON DRY STUDY SITE. © GOOGLE, IMAGE © 2015 DIGITALGLOBE.	147
FIGURE 5.32 A SUMMARY OF THE RESULTS OF THE OVERALL CLASSIFICATION ACCURACIES FOR THE MODELS BASED ON DIFFERENT VARIABLE COMBINATIONS FOR THE CAMEROON DRY AND CAMEROON WET SITE. ALL MODELS ARE RANDOM FOREST MODELS APART FROM THE TWO MENTIONED TOP 30 SVM MODELS.	149
FIGURE 5.33 A SUMMARY OF THE RESULTS OF THE OVERALL CLASSIFICATION ACCURACIES FOR THE MODELS BASED ON DIFFERENT VARIABLE COMBINATIONS FOR THE TANZANIA DRY AND TANZANIA WET SITES. ALL MODELS ARE RANDOM FOREST MODELS APART FROM THE TWO MENTIONED TOP 30 SVM MODELS.	149
FIGURE 6.1 CLASSIFICATION ACCURACIES OF RANDOM FOREST MODELS BASED ON CAMEROON WET SEASON TRAINING SAMPLES, APPLIED DIRECTLY TO CAMEROON DRY SEASON SAMPLES. SEVERAL COMBINATIONS OF INPUT VARIABLES FOR THE VARIOUS FREQUENCY COMBINATIONS ARE SHOWN.	163
FIGURE 6.2 CLASSIFICATION ACCURACIES OF RANDOM FOREST MODELS BASED ON CAMEROON WET SEASON DATA, APPLIED TO CAMEROON DRY SEASON DATA, WITH SITE-SPECIFIC CLASSES. SEVERAL COMBINATIONS OF INPUT VARIABLES FOR THE VARIOUS FREQUENCY COMBINATIONS ARE SHOWN.	164
FIGURE 6.3 CLASSIFICATION ACCURACIES OF THE RANDOM FOREST MODELS BASED ON CAMEROON WET SEASON DATA, APPLIED DIRECTLY TO DRC WET SEASON DATA. SEVERAL COMBINATIONS OF INPUT VARIABLES FOR THE VARIOUS FREQUENCY COMBINATIONS ARE SHOWN.	165
FIGURE 6.4 CLASSIFICATION ACCURACIES OF RANDOM FOREST MODELS BASED ON CAMEROON WET SEASON DATA, APPLIED TO DRC WET SEASON DATA, WITH SITE-SPECIFIC CLASSES. SEVERAL COMBINATIONS OF INPUT VARIABLES FOR THE VARIOUS FREQUENCY COMBINATIONS ARE SHOWN.	166

FIGURE 6.5 CLASSIFICATION ACCURACIES OF RANDOM FOREST MODELS BASED ON TANZANIA DRY SEASON DATA, APPLIED DIRECTLY TO TANZANIA WET SEASON DATA. SEVERAL COMBINATIONS OF INPUT VARIABLES FOR THE VARIOUS FREQUENCY COMBINATIONS ARE SHOWN.	167
FIGURE 6.6 CLASSIFICATION ACCURACIES OF RANDOM FOREST MODELS BASED ON TANZANIA DRY SEASON DATA, APPLIED TO TANZANIA WET SEASON DATA, WITH SITE-SPECIFIC CLASSES. SEVERAL COMBINATIONS OF INPUT VARIABLES FOR THE VARIOUS FREQUENCY COMBINATIONS ARE SHOWN.	168
FIGURE 6.7 CLASSIFICATION ACCURACIES OF RANDOM FOREST MODELS BASED ON TANZANIA DRY SEASON DATA, APPLIED DIRECTLY TO CHAD DRY SEASON DATA. SEVERAL COMBINATIONS OF INPUT VARIABLES ARE SHOWN. ONLY L-BAND DATA OVERLAPPED BETWEEN THE TWO SITES.	169
FIGURE 6.8 CLASSIFICATION ACCURACIES OF RANDOM FOREST MODELS BASED ON TANZANIA DRY SEASON DATA APPLIED TO CHAD DRY SEASON DATA, WITH SITE-SPECIFIC CLASSES. SEVERAL COMBINATIONS OF INPUT VARIABLES ARE SHOWN. ONLY L-BAND DATA OVERLAPPED BETWEEN THE TWO SITES.	170
FIGURE 6.9 THE OVERALL CLASSIFICATION ACCURACIES FROM THE RANDOM FOREST MODELS, TO TEST THE TRANSFERABILITY OF THE MODELS FOR THE FORESTED SITES, USING SINGLE FREQUENCIES. THIS IS THE SAME DATA FROM TABLE 6.3A REPRESENTED AS A BAR CHARTS.	175
FIGURE 6.10 THE OVERALL CLASSIFICATION ACCURACIES FROM THE RANDOM FOREST MODELS, TO TEST THE TRANSFERABILITY OF THE MODELS FOR THE FORESTED SITES, USING DUAL/THREE FREQUENCIES. THIS IS THE SAME DATA FROM TABLE 6.3B REPRESENTED AS A BAR CHARTS.	176
FIGURE 6.11 THE OVERALL CLASSIFICATION ACCURACIES FROM THE RANDOM FOREST MODELS, TO TEST THE TRANSFERABILITY OF THE MODELS FOR THE SEMI-ARID SITES, USING SINGLE FREQUENCIES. THIS IS THE SAME DATA FROM TABLE 6.4A REPRESENTED AS A BAR CHARTS.	177
FIGURE 6.12 THE OVERALL CLASSIFICATION ACCURACIES FROM THE RANDOM FOREST MODELS, TO TEST THE TRANSFERABILITY OF THE MODELS FOR THE SEMI-ARID SITES, USING DUAL/THREE FREQUENCIES. THIS IS THE SAME DATA FROM TABLE 6.4B REPRESENTED AS A BAR CHARTS.	178
FIGURE 7.1 DISTRIBUTION OF BACKSCATTER COEFFICIENTS FOR THE CAMEROON WET SEASON TRAINING VS TEST SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE BACKSCATTER COEFFICIENTS.	194
FIGURE 7.2 DISTRIBUTION OF THE TOP 10 INTERCHANNEL RATIOS FOR THE CAMEROON WET SEASON TRAINING VS TEST SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE INTERCHANNEL RATIOS.	195
FIGURE 7.3 DISTRIBUTION OF THE TOP 10 TEXTURE MEASURES FOR THE CAMEROON WET SEASON TRAINING VS TEST SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE TEXTURE MEASURES.	196
FIGURE 7.4 DISTRIBUTION OF ELEVATION, SLOPE AND ASPECT DATA FOR THE CAMEROON WET SEASON TRAINING VS TEST SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE ELEVATION VARIABLES.	197
FIGURE 7.5 DISTRIBUTION OF BACKSCATTER COEFFICIENTS FOR THE TANZANIA DRY TRAINING VS TEST SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE BACKSCATTER COEFFICIENTS.	198

FIGURE 7.6 DISTRIBUTION OF THE INTERCHANNEL RATIOS FOR THE TANZANIA DRY TRAINING TEST SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE INTERCHANNEL RATIOS.	199
FIGURE 7.7 DISTRIBUTION OF THE TOP 10 TEXTURE MEASURES FOR THE TANZANIA DRY TRAINING VS TEST SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE TEXTURE MEASURES.	200
FIGURE 7.8 DISTRIBUTION OF THE ELEVATION, SLOPE AND ASPECT FOR THE TANZANIA DRY TRAINING VS TEST SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE ELEVATION DATA.	201
FIGURE 7.9 DISTRIBUTION OF BACKSCATTER COEFFICIENTS FOR THE CAMEROON WET VS DRY SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE BACKSCATTER COEFFICIENTS.	205
FIGURE 7.10 DISTRIBUTION OF THE TOP 10 INTERCHANNEL RATIOS FOR THE CAMEROON WET VS DRY SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE INTERCHANNEL RATIOS.	206
FIGURE 7.11 DISTRIBUTION OF THE TOP 10 TEXTURE MEASURES FOR THE CAMEROON WET VS DRY SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE TEXTURE MEASURES.	207
FIGURE 7.12 DISTRIBUTION OF ELEVATION, SLOPE AND ASPECT DATA FOR THE CAMEROON WET VS DRY SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE ELEVATION VARIABLES.	208
FIGURE 7.13 DISTRIBUTION OF BACKSCATTER COEFFICIENTS FOR THE CAMEROON VS DRC WET SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE BACKSCATTER COEFFICIENTS.	209
FIGURE 7.14 DISTRIBUTION OF THE TOP 10 INTERCHANNEL RATIOS FOR THE CAMEROON VS DRC WET SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE INTERCHANNEL RATIOS.	210
FIGURE 7.15 DISTRIBUTION OF THE TOP 10 TEXTURE MEASURES FOR THE CAMEROON VS DRC WET SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE TEXTURE MEASURES.	211
FIGURE 7.16 DISTRIBUTION OF THE ELEVATION, SLOPE AND ASPECT FOR THE CAMEROON VS DRC WET SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE ELEVATION DATA.	212
FIGURE 7.17 DISTRIBUTION OF BACKSCATTER COEFFICIENTS FOR THE TANZANIA DRY VS WET SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE BACKSCATTER COEFFICIENTS.	214
FIGURE 7.18 DISTRIBUTION OF THE INTERCHANNEL RATIOS FOR THE TANZANIA DRY VS WET SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE INTERCHANNEL RATIOS.	215
FIGURE 7.19 DISTRIBUTION OF THE TOP 10 TEXTURE MEASURES FOR THE TANZANIA DRY VS WET SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE TEXTURE MEASURES.	216
FIGURE 7.20 DISTRIBUTION OF THE ELEVATION, SLOPE AND ASPECT FOR THE TANZANIA DRY VS WET SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE ELEVATION DATA.	217
FIGURE 7.21 DISTRIBUTION OF BACKSCATTER COEFFICIENTS FOR THE TANZANIA VS CHAD DRY SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE BACKSCATTER COEFFICIENTS.	218
FIGURE 7.22 DISTRIBUTION OF THE TOP 10 INTERCHANNEL RATIOS FOR THE TANZANIA VS CHAD DRY SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE INTERCHANNEL RATIOS.	219
FIGURE 7.23 DISTRIBUTION OF THE TOP 10 TEXTURE MEASURES FOR THE TANZANIA VS CHAD DRY SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE TEXTURE MEASURES.	220

FIGURE 7.24 DISTRIBUTION OF THE ELEVATION, SLOPE AND ASPECT FOR THE TANZANIA VS CHAD DRY SEASON SAMPLES. ORDER BASED ON MDG VARIABLE IMPORTANCE OF A RANDOM FOREST MODEL OF THE ELEVATION DATA.	221
FIGURE 7.25 THE IDEAL SCENARIO OF KLD BETWEEN TWO SITES THAT WOULD ALLOW MODELS TO BE COMPLETELY TRANSFERABLE BETWEEN SITE 1 AND SITE 2. THE CLASSES ARE SIMILAR BETWEEN THE TWO SITES, WHEREAS THE DIFFERENCES BETWEEN CLASSES ARE COMPLETELY SEPARABLE.	223
FIGURE 7.26 THE KLD OF THE BACKSCATTER COEFFICIENTS FOR THE LAND COVER CLASSES OF THE CAMEROON WET TRAINING VS TEST SAMPLES.	224
FIGURE 7.27 THE KLD OF THE TOP 8 TEXTURE MEASURES FOR THE LAND COVER CLASSES OF THE CAMEROON WET TRAINING VS TEST SAMPLES. THE ORDER OF THE TOP 8 VARIABLES IS BASED ON THE MDG VARIABLE IMPORTANCE MEASURE FROM THE RANDOM FOREST MODEL.	224
FIGURE 7.28 THE KLD OF THE BACKSCATTER COEFFICIENTS FOR THE LAND COVER CLASSES OF THE CAMEROON WET VS DRY SEASON SAMPLES.	225
FIGURE 7.29 THE KLD OF THE TOP 8 TEXTURE MEASURES FOR THE LAND COVER CLASSES OF THE CAMEROON WET VS DRY SEASON SAMPLES. THE ORDER OF THE TOP 8 VARIABLES IS BASED ON THE MDG VARIABLE IMPORTANCE MEASURE FROM THE RANDOM FOREST MODEL.	226
FIGURE 7.30 THE KLD OF THE BACKSCATTER COEFFICIENTS FOR THE LAND COVER CLASSES OF THE CAMEROON VS DRC WET SEASON SAMPLES.	227
FIGURE 7.31 THE KLD OF THE TOP 8 TEXTURE MEASURES FOR THE LAND COVER CLASSES OF THE CAMEROON VS DRC WET SEASON SAMPLES. THE ORDER OF THE TOP 8 VARIABLES IS BASED ON THE MDG VARIABLE IMPORTANCE MEASURE FROM THE RANDOM FOREST MODEL.	227
FIGURE 7.32 THE KLD OF THE BACKSCATTER COEFFICIENTS FOR THE LAND COVER CLASSES OF THE TANZANIA DRY SEASON TRAINING VS TEST SAMPLES.	229
FIGURE 7.33 THE KLD OF THE TOP 8 TEXTURE MEASURES FOR THE LAND COVER CLASSES OF THE TANZANIA DRY SEASON TRAINING VS TEST SAMPLES. THE ORDER OF THE TOP 8 VARIABLES IS BASED ON THE MDG VARIABLE IMPORTANCE MEASURE FROM THE RANDOM FOREST MODEL.	230
FIGURE 7.34 THE KLD OF THE BACKSCATTER COEFFICIENTS FOR THE LAND COVER CLASSES OF THE TANZANIA DRY VS WET SEASON SAMPLES.	231
FIGURE 7.35 THE KLD OF THE TOP 8 TEXTURE MEASURES FOR THE LAND COVER CLASSES OF THE TANZANIA DRY VS WET SEASON SAMPLES. THE ORDER OF THE TOP 8 VARIABLES IS BASED ON THE MDG VARIABLE IMPORTANCE MEASURE FROM THE RANDOM FOREST MODEL.	231
FIGURE 7.36 THE KLD OF THE BACKSCATTER COEFFICIENTS FOR THE LAND COVER CLASSES OF THE TANZANIA VS CHAD DRY SEASON SAMPLES.	232
FIGURE 7.37 THE KLD OF THE TOP 8 TEXTURE MEASURES FOR THE LAND COVER CLASSES OF THE TANZANIA VS CHAD DRY SEASON SAMPLES. THE ORDER OF THE TOP 8 VARIABLES IS BASED ON THE MDG VARIABLE IMPORTANCE MEASURE FROM THE RANDOM FOREST MODEL.	233
FIGURE 7.38 AN OUTLINE OF THE DIAGONAL AND OFF-DIAGONAL ELEMENTS OF THE MATRIX OF THE KLD AVERAGES.	234

FIGURE 7.39 THE KLD CLASS DIFFERENCE COMPARED TO THE USER'S ACCURACY OF THE RANDOM FOREST MODELS USING AT MOST THE TOP 8 BACKSCATTER COEFFICIENT CHANNELS AND THE TOP 8 TEXTURE MEASURES FOR THE FORESTED SITES.	236
FIGURE 7.40 THE KLD CLASS DIFFERENCE COMPARED TO THE USER'S ACCURACY OF THE RANDOM FOREST MODELS USING AT MOST THE TOP 8 BACKSCATTER COEFFICIENT CHANNELS AND TOP 8 TEXTURE MEASURES FOR THE SEMI-ARID SITES.	237
FIGURE A1.1 DIRECTORY STRUCTURE TO BE CREATED FOR THE GAMMA ORTHORECTIFICATION SCRIPTS.	259
FIGURE A1.2 THE DIRECTORY STRUCTURE OF THE /DEM FOLDER, WITH LOCATIONS OF SCRIPTS	260
FIGURE A2.1 FOLDER STRUCTURE FOR THE GDAL SCRIPTS DATA PREPARATION STEPS.	311



## List of abbreviations

AGB	Above-ground Biomass
ALOS	Advanced Land Observing Satellite
ANN	Artificial Neural Network
ANOVA	ANalysis Of VAriance
AP	Alternating Polarisations
ASAR	Advanced Synthetic Aperture Radar
ASI	Agenzia Spaziale Italiana (Italian Space Agency)
CC	Canopy Cover
CD	Cameroon dry season
ChD	Chad dry season
CLL	CORINE Land Cover
CNES	Centre national d'études spatiales / National Centre of Space Studies (French space agency)
CORINE	Coordination of Information on the Environment
COSMO-Skymed	COnstallation of small Satellites for the Mediterranean basin Observatory
CRISP-DM	Cross Industry Standard Process for Data Mining
CW	Cameroon wet season
DEM	Digital Elevation Model
DLR	Deutsches Zentrum für Luft- und Raumfahrt (German Aerospace Center)
DRC	Democratic Republic of the Congo
DW	DRC wet season
EADS	European Aeronautic Defence and Space Company (renamed to Airbus Defence and Space)
EC	European Commission
ECV	Essential Climate Variables
EEA	European Environment Agency
EEC	Enhanced Ellipsoid Corrected

EIONET	European Environment Information and Observation Network
ENVISAT	Environmental satellite
EPSG	European Petroleum Survey Group
ERS	European Remote Sensing Satellite
ESA	European Space Agency
EUMETSAT	European Organisation for the Exploitation of Meteorological Satellites
EU	European Union
FAO	Food and Agriculture Organization of the United Nations
FBD	Fine Beam Double polarisation mode
FBS	Fine Beam Single polarisation mode
FEWSNET	Famine Early Warning Systems Network
GCP	Ground Control Point
GIO	GMES Initial Operations
GIONET	GMES Initial Operations - Network for Earth Observation Research Training
GLC2000	Global Land Cover database for the year 2000
GLONASS	GLOBAL NAVIGATION Satellite System
GMES	Global Monitoring of Environment and Security
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HSPol	High Sensitive Polarimetric mode
IDP	Internally Displaced Persons
IM	Image Mode
IUCN	International Union for Conservation of Nature
JAXA	Japanese Aerospace Exploration Agency
JERS	Japanese Earth Resources Satellite
JPL	Jet Propulsion Laboratory
JRC	Joint Research Centre
KLD	Kullback-Leibler Divergence
LCC	Land Cover Change

LCCS	Land Cover Classification System
MaxEnt	Maximum Entropy
MDA	Mean Decrease Accuracy
MDG	Mean Decrease Gini
MERIS	Medium Resolution Imaging Spectrometer
MIMICS	Michigan Microwave Canopy Scattering model
MLC	Maximum Likelihood Classifier
MLR	Multivariate Linear Regression
NASA	National Aeronautics and Space Administration
NDVI	Normalised Difference Vegetation Index
NEST	Next ESA SAR Toolbox
NIR	Near Infrared
NRC	National Reference Centre
OBIA	Object-based Image Analysis
OOB	out-of-bag sample
OSH	Optimal linear Separating Hyperplane
PALSAR	Phased Array L-band Synthetic Aperture Radar
POL	Quad-Polarimetric mode
$R^2$	coefficient of determination of regression analysis
RBF	Radial Basis Function
REDD	Reducing Emissions from Deforestation and forest Degradation
RE	Relative Entropy
RF	Random Forests
RGB	Red Green Blue
RVI	Radar Vegetation Index
SAFER	Services and Applications For Emergency Response
SAR	Synthetic Aperture Radar
SLC	Single-Look Complex
SRTM	Shuttle Radar Topography Mission
SSTL	Surrey Satellite Technology Limited
SVM	Support Vector Machine

SVR	Support Vector Regression
TCV	Total Canopy Volume
TD	Tanzania dry season
TW	Tanzania wet season
UFS	Ultra Fine Single polarimetric mode
UNEP	United Nations Environment Programme
USGS	United States Geological Survey
UTM	Universal Transverse Mercator
WGS84	World Geodetic System 1984

## **Chapter One     Introduction**

This chapter presents an overview of the thesis and sets an outline of the structure and flow of the document. A brief history of Synthetic Aperture Radar (SAR) satellites is presented to introduce the context of multi-frequency SAR, classification algorithms, land cover classification of Africa and the relevance of this thesis to the Copernicus programme.

### **1.1     Overview**

The knowledge of land cover and land cover change is important for a wide variety of applications and is being used by government departments, non-governmental organisations and private institutions to make decisions regarding planning for land resource management, conservation, biodiversity management and climate change, to name a few. Land cover is on the list of essential climate variables (ECV) that need to be monitored to assess the state of climate change as defined by the Global Climate Observing System (GCOS) in support of the United Nations Framework Convention on Climate Change (UNFCCC) and the International Panel on Climate Change (IPCC) (GCOS, 2010). Biomass and burnt area are two more ECVs related to land cover. Biomass in forests is a great store of carbon, and therefore monitoring of any deforestation can be used to account for carbon released into the atmosphere (Le Toan et al., 2011). Other uses of land cover and land cover change are: landscape ecology and the impacts of land cover change on biodiversity and ecosystems (Newton et al., 2009; Duro et al., 2007), conservation monitoring (Nagendra et al., 2013), mangrove forests monitoring (Heumann, 2011), desertification and water erosion (Albalawi and Kumar, 2013; Vrieling, 2006), agriculture and horticulture-intensive farming (Cotlier et al., 2011), wetlands monitoring (White et al., 2015; Evans and Costa, 2013) and emergency management (Joyce et al., 2009; Voigt, 2011). This is just an extract of the wide application area of land cover maps and land cover change maps used by scientists,

organisations and governments to make informed decisions ranging from the local environment to the global climate.

A big challenge for remote sensing remains to be cloud cover in many regions of the world. This is one of the main benefits of Synthetic Aperture Radar (SAR) as opposed to from optical remote sensing satellites: the ability to acquire data in any weather condition, and during the day and night. This is especially the case for regions with regular cloud cover (Kuntz, 2010), and during and after stormy weather associated with certain emergency response situations (Voigt et al., 2007). In addition to using single frequency SAR data for land cover classification, multi-frequency SAR have been investigated in a handful of studies (Li et al., 2012; Turkar et al., 2012). Turkar et al. (2012) showed very high classification accuracies of land cover classes in an area of Mumbai, India, using fully polarimetric, multi-frequency SAR. Peng et al. (2005) has shown that the addition of texture measures and elevation data can improve the land cover classification accuracies significantly. The use of ensemble classifiers, including random forests is shown to be by Waske and Braun (2009), to be successful to classify multi-temporal SAR images.

SAR satellites have been in orbit since the late 1970s. SEASAT, an L-band horizontally transmitted and horizontally received (HH) polarization SAR sensor, was the first civilian SAR mission from the United States, and operated for three months during 1978. Since then several missions followed during the 1980s with the SIR-A and SIR-B (both L-band, HH-polarization) Space Shuttle missions in 1981 and 1984 acquiring data in orbit for 2.5 days and 8 days respectively. At the time Russia was also an active contender and operated Kosmos 1870 (S-band, HH polarization) in 1987 and Almaz (S-band, HH-polarization) in 1991. Since 1991 more international participants joined in with the launch of several SAR satellites. In 1991 the European Remote Sensing satellite, ERS-1, a C-band vertically transmitted and vertically received (VV) polarization SAR sensor was launched, with the Japanese Earth Resources Satellite, J-ERS-1 (L-band, HH polarization) in 1992 (Lewis et al., 1998).

During April and October 1994 a ground-breaking mission, SIR-C / X-SAR was undertaken to obtain simultaneous measurements from L, C (quad-polarization, i.e. HH, HV, VH, VV) and X-band SAR (VV polarization) from the Space Shuttle. SIR-C / X-SAR was a collaboration between National Aeronautics and Space Administration (NASA)'s Jet Propulsion Laboratory (JPL) of the United States, the German Aerospace Center (DLR, Deutsches Zentrum für Luft- und Raumfahrt) and the Italian Space Agency (ASI, Agenzia Spaziale Italiana). JPL managed the L- and X-band SAR and the DLR and ASI managed the X-band SAR (Lewis et al., 1998). This was the first time that L-, C- and X-band images were obtained simultaneously from space and the potential of a multi-frequency SAR system was realised. The potential to use multi-frequency SAR for land cover classification was recognised and several studies were undertaken by the researchers involved. Some very high classification accuracies (90%+) were found for their Michigan test site (Pierce et al., 1995), using a knowledge-based hierarchical classifier. Classes used were flat areas, short vegetation and tall vegetation, with tall vegetation split into upland conifers, lowland conifers and deciduous trees. Other studies found more varying classification accuracy results, varying from 17% for rapeseed, 51% for built-up areas, 84% for winter cereals and open soil combined, 72% for meadows and 93% for forests (Stolz and Mauser, 1995). Classification accuracies using both the April and October data from SIR-C / X-SAR were found to give very high accuracies (90%+, with higher class-specific accuracies) using multi-temporal land cover classification (Pierce et al., 1998).

Since the SIR-C / X-SAR mission, several SAR missions have been put in orbit during the 1990s. Both ERS-2 (C-band, VV polarization) from the European Space Agency (ESA) and RADARSAT-1 (C-band, HH polarization) from Canada were launched in 1995 (Lewis et al., 1998) and ESA's Environmental Satellite, ENVISAT (C-band, VV+HH polarization) in 2002. ENVISAT was a remarkable satellite and the largest and most likely the most complex satellite built in Europe to date (Louet and Bruzzi, 1999). ENVISAT was a scientific mission, weighed 8000kg and had 10 instruments on board, including the Advanced Synthetic Aperture Radar (ASAR) instrument (Desnos et al., 2000). ENVISAT acquired data for over ten years, which was in excess of its design lifetime of 5 years, until

communication to the satellite was lost on 8 April 2012 (ESA, 2012a). Similar to ENVISAT, RADARSAT-1 outlived its design lifetime by many years. Its design lifetime was also 5 years, but the mission continued for 17 years. The RADARSAT-1 mission was ended in May 2013, when a technical anomaly was detected (Mahmood, 2015). In 2007 RADARSAT-2 (C-band, quad-polarization) was launched and is still active at the time of writing in mid-2015. RADARSAT-2 is an operational commercial satellite run by the private sector, with a large focus on sea-ice monitoring (Morena et al., 2004; Flett et al., 2008).

In addition to the satellites mentioned in the previous paragraph, which are all C-band satellites (ERS-2, ENVISAT, RADARSAT-1, RADARSAT-2), X-band and L-band SAR satellites were also launched during the mid-2000s. In 2006, the Japanese Aerospace Exploration Agency (JAXA) launched the Advanced Land Observing Satellite (ALOS), with three sensors, including the Phased Array type L-band Synthetic Aperture Radar (PALSAR) (L-band, with quad-polarisation capability) (JAXA, 2006). The ALOS mission was ended by JAXA on 12 May 2011, when it developed a power generation anomaly after five years of operation. TerraSAR-X (X-band, HH, VV, HV, VH) was launched in June 2007 by the DLR in a public-private partnership with Airbus Defence and Space, which at the time was European Aeronautic Defence and Space Company (EADS) Astrium (Werninghaus and Buckreuss, 2010). At the same time, between June 2007 and 2010 ASI launched the COntellation of small Satellites for Mediterranean basin Observatory (COSMO-SkyMed) which is a constellation of four X-band SAR satellites.

This means that for the window period between June 2007 and May 2011, SAR satellites were acquiring images from L-band, C-band and X-band SAR sensors. This was for the first time since the SIR-C / X-SAR mission in 1994 that all three SAR frequencies were acquired at the same time from space. However, each SAR satellite had its own mission objectives and the acquisitions were not coordinated as a multi-frequency mission. It did create a scenario where high resolution images were acquired from all three frequencies for the same geographic area and relatively close in time (within the same season) for several places around the world.



In 1998 the European Space Agency (ESA), the European Commission (EC), the European Organisation for the Exploitation of Meteorological Satellites (EUMETSAT) and the main European national space agencies met to establish a European-wide Earth Observation programme which came to be known as Global Monitoring for Environment and Security (GMES) (Aschbacher and Milagro-Pérez, 2012). In 2007 the Resolution on the European Space Policy (EU 4th Space Council, 2007) was adopted in the Lisbon Treaty (EU Lisbon Treaty, 2007) and GMES and Galileo were acknowledged as the two flagship programmes of the newly adopted policy (EC COM, 2007). Galileo is the European contribution to the Global Navigation Satellite System (GNSS), similar to the Global Positioning System (GPS) owned by the United States and GLObal NAVigation Satellite System (GLONASS), owned by Russia (Flament, 2004).

Funding for GMES Initial Operations (GIO) was provided by the 6<sup>th</sup> and 7<sup>th</sup> Framework Programmes of the European Union (EU). This included funding to build the first series of satellites which will provide satellite imagery for operational use, known as the Sentinel satellites. The period from 2011-2013 was the GIO phase and in May 2013 GMES was renamed Copernicus (European Commission, 2013). The Copernicus programme will provide six services, namely the Land Monitoring service, Marine service, Atmosphere service, Emergency Management service, Security Support service and Climate Change service. Of these services the Land Monitoring service (<http://land.copernicus.eu>) and Emergency Management service (<http://emergency.copernicus.eu>) are in operational phase and with funding secured for 2014-2020. All six services should become operational within the next few years (European Commission, 2013).

The first of the Sentinel satellites, Sentinel 1A, is a C-band dual-polarisation (HH+HV, VV+VH) SAR satellite and was successfully launched on 3 April 2014. Sentinel 1 will be used for both the Land Monitoring and the Emergency Management Copernicus services (Torres et al., 2012). In current land cover and emergency management applications, most SAR applications focus on using single frequency applications, but the possibility of using information from more than one SAR sensor, and therefore more than one

frequency has not been fully investigated. There are currently many SAR missions planned for the next 5-10 years for different SAR frequencies. As mentioned, Sentinel 1 (C-band) was launched on 4 April 2014 with data available from 3<sup>rd</sup> October 2014 (ESA, 2014). The Japanese Advanced Land Observing Satellite-2 (ALOS-2) (L-band) was launched on 24 May 2014 (JAXA, 2014b), with data available from 25 November 2014 (JAXA, 2014a). Other SAR satellites that are being developed are TerraSAR-X2 (X-band) (Janoth et al., 2012), NovaSAR-S (S-band) (SSTL, 2011) and BIOMASS (P-band) (Le Toan et al., 2011), to name a few, and the trend seems set to continue in the foreseeable future. This means that from 25 November 2014, multi-frequency SAR is again available by combining images from several SAR satellites, and the opportunity to utilise multi-frequency SAR from L-, C- and X-band frequencies is reinstated.

The African continent is a relevant geographic context for the emergency response management services, such as the Copernicus Emergency Management service. To investigate the use of multi-frequency SAR data for land cover classification for the forested and semi-arid ecoregions, the data archive of TerraSAR-X of the African continent was first queried. Secondly, archive data from the Environmental Satellite (ENVISAT) Advanced Synthetic Aperture Radar (ASAR) and ALOS Phased Array L-band Synthetic Aperture Radar (PALSAR) were overlayed over the available TerraSAR-X images. The number of images from the archives of TerraSAR-X, ENVISAT ASAR and ALOS PALSAR for the year 2010 are shown in Figure 1.1. Starting with the TerraSAR-X images, areas where the images overlap geographically and close in time were selected, based on data availability in the respective archives. As a result, study sites representing forested ecoregions were identified in Cameroon and the Democratic Republic of the Congo (DRC), and study sites representing semi-arid ecoregions were identified in Tanzania and Chad (see Section 3.1 and 3.2 for a detailed description of the study sites). For both the semi-arid and forested sites, data are available for the wet and dry seasons, which were identified using rainfall data obtained from the Famine Early Warning Systems Network ([www.fews.net](http://www.fews.net)).

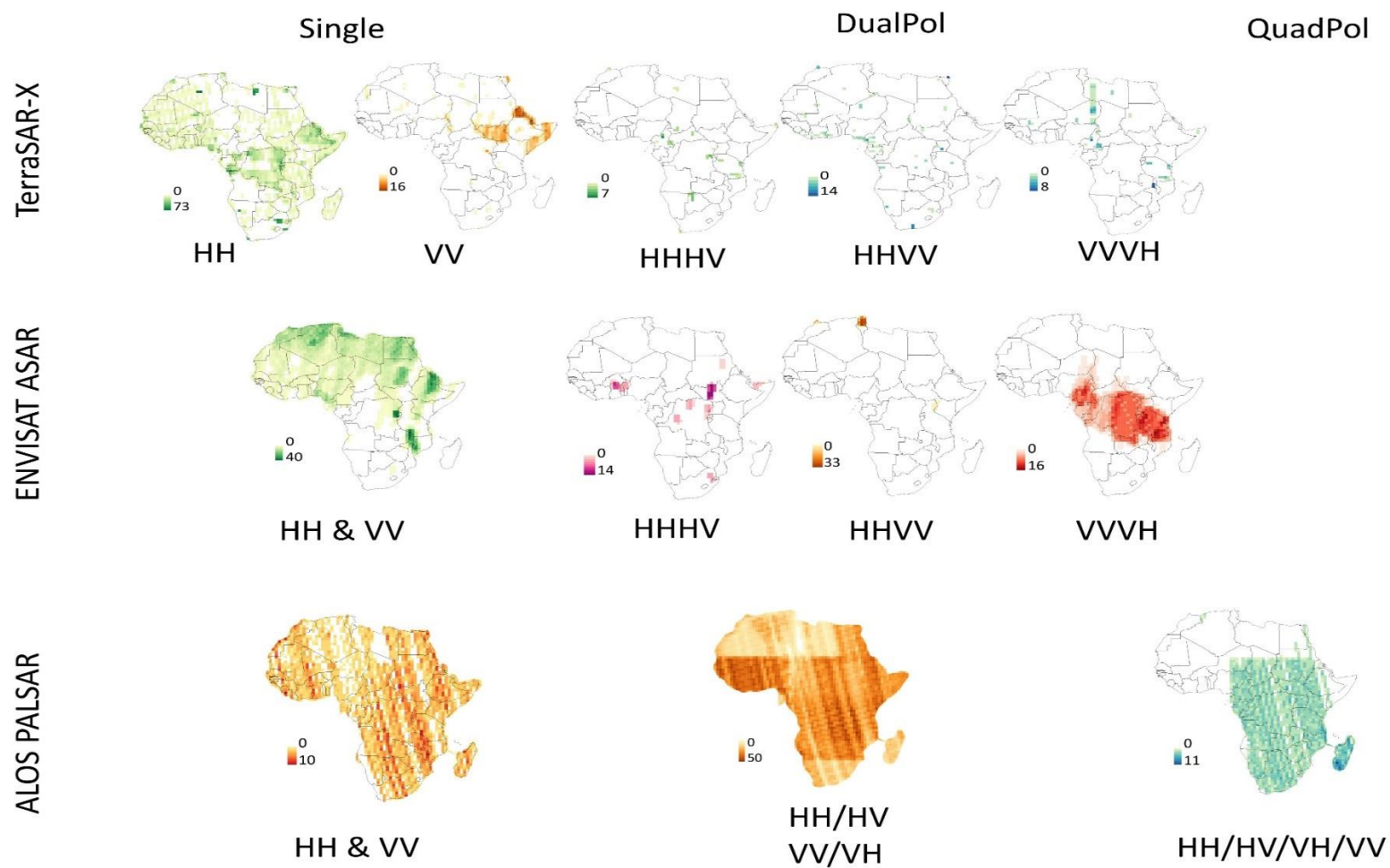


Figure 1.1 Availability of SAR data from ALOS PALSAR, ENVISAT ASAR and TerraSAR-X for the full year of 2010 over the African continent.

## 1.2 Hypotheses for the thesis

The overall objective of this study is to enhance current available land cover classification techniques to classify landscapes and quantify change, regardless of what data is used. Within this context of available multi-frequency SAR data, to address this overall objective, the following hypotheses are tested in this thesis:

- 1a. Multi-frequency SAR images provide a higher classification accuracy than single frequency SAR for land cover classification.
- 1b. Random forest classification algorithm gives higher classification accuracies than the support vector machine algorithm.

To test these hypotheses, combinations of single, dual and three frequencies are compared for:

- a. The forested ecoregion wet and dry season images.
- b. The semi-arid ecoregion wet and dry season images.

Features that are calculated in addition to the SAR backscatter coefficient images are:

- Interchannel ratios
- Texture measures
- Elevation data

Classification accuracy results of support vector machine (SVM) models are compared to the random forest models, using the variables selected by the random forest model.

2. The SAR models are transferable between different seasons of the same site, and to different sites of similar ecoregions.

To test this hypothesis, the classification accuracies are compared for the:

- a. Forested ecoregion:
    - Site1 wet vs dry season images.
    - Site1 wet vs site2 wet season images.
  - b. Semi-arid ecoregion:
    - Site1 dry vs wet season images.
    - Site1 dry vs site2 dry season images.
3. The land cover classes are similar between different seasons, and similar between different sites of the same ecoregion.

To test this hypothesis, the SAR signatures of land cover classes are compared, both visually and with a statistical separability measure, for the:

- a. Forested ecoregion:
  - Site1 wet vs dry season images.
  - Site1 wet vs site2 wet season images.
- b. Semi-arid ecoregion:
  - Site1 dry vs wet season images.
  - Site1 dry vs site2 dry season images.

By testing these hypotheses, the use of multi-frequency SAR are tested as a viable classification strategy to classify images in regions that would otherwise be affected by cloud cover. Testing hypothesis 1a, a comparison will be done for single (L-, C-, X-band), double (LC, LX, CX-bands) and all three frequencies (LCX-bands). This will create a set of recommendations as to which frequency or frequency combinations are most suitable for forested and semi-arid ecoregions, as well for the wet and dry seasons for both ecoregions. Testing hypothesis 1b will result in a comparison between random forest modelling and SVM modelling, for the multi-frequency SAR data of the forested and semi-arid ecoregions.

By addressing hypothesis 2, a guideline will be established as to whether these models can be transferred between seasons and to additional sites. This is crucial for applications such as emergency response management monitoring, where the transferability of models can improve the response time and enable the application of a model developed for a certain region and season to another region or season.

By testing hypothesis 3, a fundamental question will be answered with the regards to the SAR signature and how different the SAR signal are between the wet and dry seasons, and between the main and additional sites for both the forested and semi-arid ecoregions.

### 1.3 Thesis outline

For clarity of the overall flow between the different chapters and the three hypotheses, this research project is shown as a schematic diagram (Figure 1.2). This is similar to a Cross Industry Standard Process for Data Mining (CRISP-DM) project (Sharma et al., 2012; Piatetsky-Shapiro, 2007). First the context is set and the literature review given in Chapter Two. Chapter Three explains the the data availability and the chosen study sites. Chapter Four lists the data preparation steps, land cover class identification and sample selection for each of the study sites. The data preparation includes the additional data feature calculations of interchannel ratios, texture measures and elevation data.

Chapter Five addresses the first hypothesis of comparing multi-frequency SAR vs single frequency SAR for land cover classification. In the same chapter the random forest algorithm is compared to the SVM algorithm. The challenge quickly becomes a multi-dimensional problem, which is well handled by machine learning algorithms in the field of data mining. The machine learning algorithms compared in this research were random forests (Breiman, 2001) and SVMs (Cortes and Vapnik, 1995). The random forest algorithm, as implemented in the *randomForest* package in R (Liaw and Wiener, 2002), gives measures of variable importance as part of its output. This allows the researcher to select a smaller set of variables with possibly only a marginal decrease in classification

accuracy. The approach in this research is to first develop a model, using all the input variables for several scenarios. The scenarios include single frequencies (L vs C vs X), compared with two frequencies (LC, LX and CX) and then with using all the data available (LCX). Using the variable importance measures from the random forest model, a smaller set of variables is chosen to have a more agile model. This smaller set of variables, containing the variables that carry the largest weight in the random forest model, is then used to train an SVM model. The classification accuracy results are then compared between the random forest and SVM models. In this comparison the classification algorithm that gives the highest classification accuracy result is then selected for each of the semi-arid (wet vs dry) and forested (wet vs dry) scenarios.

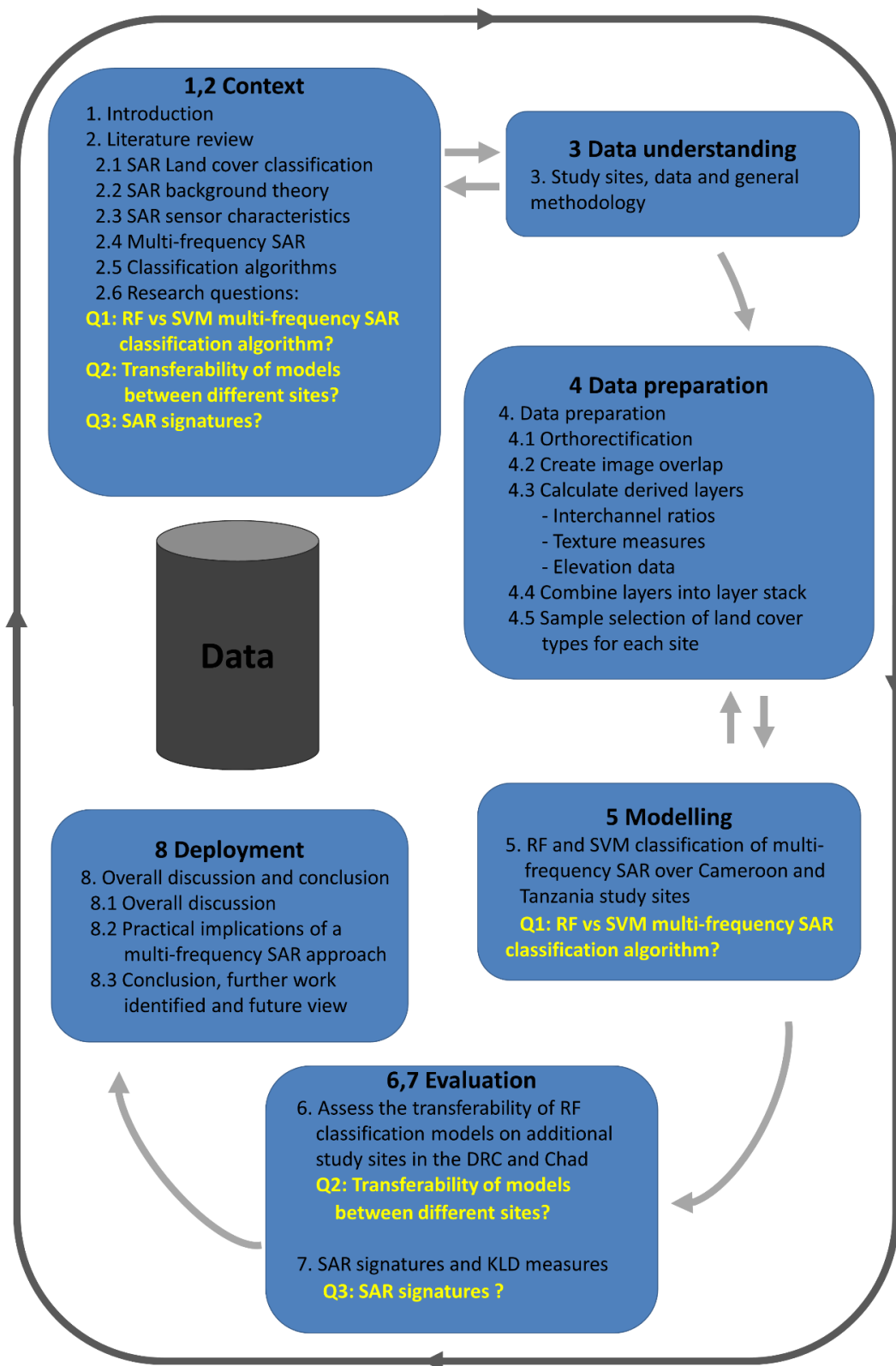


Figure 1.2 Schematic overview of this research in terms of the Cross Industry Standard Process for Data Mining.



Chapter Six addresses the second hypothesis to test the transferability of the models to a different season, for the same study site, and to a different sit location. This is done for both the forested and semi-arid ecoregions. The second hypothesis looks into the transferability of the random forest model for each of the semi-arid (wet vs dry) and forested (wet vs dry) landscape scenarios. This is a crucial part of the research project, since many studies often only focus on one study site, resulting in the classification model being study site specific, and possibly overtrained. The intention is to assess whether the classification models generalize well enough to be used on another set of input images from semi-arid or forested areas within Africa.

Chapter Seven addresses the third hypothesis, to test how different the SAR response is for the wet and dry seasons, and for different sites for both the forested and semi-arid ecoregions. The statistical distributions of each of the SAR backscatter coefficient channels and derived information layers (interchannel ratios, texture measures and elevation data) for each land cover class are visually compared. This is done for the training vs test data, inter-season and inter-site comparison for both the forested and semi-arid study sites. Apart from the visual comparison of the statistical distributions of the land cover classes, the Kullback-Leibler divergence (KLD) separability index measure are also compared for the backscatter coefficients and most prominent texture measures. From the KLD measures a KLD class difference measure is derived, and the relation between this newly derived KLD class difference measure and the user's accuracy from a random forest model using the same input variables are assessed. This will therefore address the hypothesis of how separable the SAR signatures of different land cover types are for the different landscape scenarios.

Finally Chapter Eight gives an overall discussion and conclusion, identification of further work and a future view. With the identification of which frequencies and derived feature layers separate each land cover class from the other land cover classes most effectively, and with a set of classification algorithms for each landscape scenario, recommendations are made for a possible monitoring mission of a constellation of multi-frequency SAR satellites aimed at land cover monitoring and emergency response

situations across Africa and other regions around the world with similar semi-arid and forested landscapes.

A selection of the computer code used for this research are shown in the Appendices. The scripts to orthorectify the SAR images are given in Appendix One (GAMMA orthorectification scripts). The scripts to calculate the additional information layers of interchannel ratios, texture measures and elevation data; and creating an overlay and layer stack of the images are given in Appendix Two (GDAL scripts to create image overlay). The scripts to do the R modelling and plotting graphs are given in Appendix Three (R modelling and Graphing scripts).

## **Chapter Two      Literature review on multi-frequency SAR land cover classification**

In Europe, the main land cover mapping initiative is the Coordination of Information on the Environment (CORINE) Land Cover project that was initiated in 1985 and is periodically produced by the European Environment Information and Observation Network (EIONET) National Reference Centres (NRCs) for Land Cover and coordinated by the European Environment Agency (EEA). The CORINE Land Cover (CLC) maps produced so far are CLC1990, CLC2000 and CLC2006, with CLC2012 in the process of being completed. As part of the CLC maps, Land Cover Change (LCC) maps are also produced, resulting in LCC1990-2000, LCC2000-2006 and LCC2006-2012 (EEA, 2007, 2015). The CLC as well as other similar mapping initiatives around the world, uses images from multi-spectral optical sensors such as Landsat, SPOT, RapidEye or similar sensors with spatial resolution varying from 5-30m. For the CLC project the minimum mapping unit is 25ha and minimum width of 100m. All changes greater than 5ha are mapped on the LCC maps (EEA, 2015).

The strengths of the CLC approach to creating land cover maps are that it is well coordinated and produced by the EEA, and that there are regional experts partaking in the validation of the land cover maps from each country involved. However, this resolution of land cover mapping is not readily available, for a large portion of the world. There are a collection of global land cover products available, but they are at a much coarser resolution. The Global Land Cover database for the year 2000 (GLC2000) used SPOT VEGETATION data at 1km spatial resolution (Bartholomé and Belward, 2005), GLOBCOVER using Medium Resolution Imaging Spectrometer (MERIS) data from ENVISAT at 300m (Arino et al., 2008) and the Modis global land cover product at 500m spatial resolution (Friedl et al., 2010).

One shortcoming of this approach is that it requires a huge amount of time and effort to produce and validate these land cover maps. CLC maps are currently produced every

6 years, even though the map production time has been reduced from 10 years to 2 years (EEA, 2015). Another shortcoming of using optical images for land cover classification is that it is difficult to obtain multiple clear and cloud-free optical images throughout the year for some regions of the world. This is particularly so in the tropical regions (Asner, 2001; Wylie et al., 2005), but also for some higher latitude regions (Ju and Roy, 2008; Armitage et al., 2013). One of the main advantages of SAR, over optical images, is that the SAR images can be acquired in any weather condition, and is therefore largely unaffected by cloud cover (Kuntz, 2010).

Two areas of research are therefore identified to improve the land cover classification process: the development of a more automated process to not be reliant on this intensive manual process, and the use of SAR images to eliminate the dependence on utilising only cloud-free images (Kuntz, 2010). Utilising SAR is furthermore of particular interest to emergency response situations when images are needed quickly, and where the areas affected are often in stormy environments with overcast cloud conditions (Voigt et al., 2007; Covella et al., 2010).

Several studies have been done to use SAR for land cover classification. Most of the studies look at single-frequency SAR applications (McNairn and Brisco, 2004; Rosenqvist and De Grandi, 2009; Longepe et al., 2010; Marti-Cardona et al., 2010; Li et al., 2012). A detailed comparison of optimal SAR parameters for various application disciplines are given in (Schmullius and Evans, 1997). Schmullius and Evans (1997) found that each frequency and polarisation has its strengths for specific applications.

To maintain a sustainable change monitoring approach, a systematic acquisition strategy is needed. A systematic acquisition strategy is when images are acquired every set number of days for a specific location. This enables the monitoring of changes for these locations. Examples of satellite sensors that follow a systematic acquisition strategy are Landsat and Sentinel 1. The Landsat is an optical sensor which acquires images continuously, resulting in a revisit time of every 16 days. The latest Landsat satellite is Landsat 8, and builds on 30-40 years of systematic data acquisition from the

Landsat programme. Sentinel 1 is a C-band SAR sensor, and acquires images continuously, with a revisit time of every 12 days around the globe. Sentinel 1B will complement Sentinel 1A, allowing a revisit time for the Sentinel 1 constellation of every 6 days. A systematic acquisition strategy is important for applications such as monitoring deforestation. This allows for the monitoring of very large areas, such as the whole Amazon rainforest, or the Congo basin in Africa. This enables a regular acquisition of a specific location, and allows for the building up of a set of historical data that would not be available if data was only selected irregularly.

Some research papers focused on using full quad-polarimetric images, which allow for polarimetric processing of the images (Ainsworth et al., 2009; Qi et al., 2010; Moran et al., 2012). This current research excludes the full polarimetric processing, since it is not readily available on all SAR systems, and is a focus area in its own right. Any interchannel ratios from available dual-polarimetric images are included in this research. Another aspect of SAR is the ability to perform interferometric calculations between two images. This is usually used for displacement mapping such as earthquakes and landslides (Metternicht et al., 2005; Joyce et al., 2009). Interferometric processing has been applied to land cover mapping (Strozzi et al., 2000; Okhimamhe, 2003) though any interferometric processing is excluded from this research.

Studies conducted using the multi-frequency SAR images from the SIR-C / X-SAR shuttle mission in 1994 showed the potential of utilising multi-frequency SAR for land cover classification (Pierce et al., 1995, 1998). Multi-frequency SAR applications have been studied from airborne sensors and are of particular interest to the current study, because the SAR signals from multiple frequencies can be acquired concurrently. There have been several airborne multi-frequency campaigns, such as the NASA / JPL AirSAR (Quegan et al., 2003; Lucas et al., 2006) the DLR's E-SAR (Rowland et al., 2008), and the latest F-SAR multi-frequency SAR airborne campaign (Horn et al., 2009). The F-SAR instrument enables the simultaneous capturing of P-, L-, C- and X-band images. For a multi-frequency system, this would be the ideal scenario, however this would be limited to airborne instruments. The global reach of satellite missions for a multi-frequency SAR

setup is still currently lacking. Therefore, to make use of multi-frequency SAR measurements, one presently needs to combine measurements from multiple sensors on different satellites.

Recently, more studies have looked at the use of multi-frequency SAR for land cover classification using a combination of SAR frequencies from satellite (Turkar et al., 2012; Naidoo et al., 2015). Turkar (2012) compared the classification accuracies of L-band (ALOS PALSAR), C-band (RADARSAT-2), X-band (TerraSAR-X) and combinations thereof, of land cover classes: water, mangrove, urban, forest, saltpans, sewage plant, wetland and grassland of a study site in Mumbai using ANN and Wishart classifiers. Fully polarimetric data were used from ALOS PALSAR and RADARSAT and DualPol from TerraSAR-X data. Overall classification accuracies of up to 98% were achieved using all three frequencies with polarimetric decompositions. The main conclusions from Turkar (2012) were that (1) classification accuracies using co-polarised data are marginally lower than fully-polarised data, (2) classification using complex data compared to only backscatter coefficients leads to higher classification accuracy, (3) L-band is in general the best frequency for land cover classification, and (4) combining multi-frequency data (L, C and X-band) improves classification accuracy.

Naidoo (2015) compared the use of L-band (ALOS PALSAR), C-band (RADARSAT-2) and X-band (TerraSAR-X) data to classify above ground biomass (AGB), canopy cover (CC) and total canopy volume (TCV). Naidoo concluded that, although the use of all three frequencies yielded the best results ( $R^2 = 0.83$  for AGB and CC, and  $R^2 = 0.85$  for TCV, with  $R^2$  being the coefficient of determination of a regression analysis), it was marginal above using only L-band data ( $R^2 = 0.78$  for AGB, 0.77 for CC and 0.79 for TCV). Naidoo made use of the random forest machine learning algorithm to obtain these results.

From the SAR backscatter coefficient images, additional feature layers can be calculated, such as interchannel ratios and texture measures. Interchannel ratios of SAR images have been investigated in several studies (Buckley, 2002; Quegan et al., 2003; Dierking et al., 2003; Simental et al., 2005; Lönnqvist et al., 2010). Lönnqvist found that the

addition of the interchannel ratio layers gave an increase of at least 2.2% on using only the basic backscatter coefficient channels.

A set of texture measure of images was presented by Haralick (1973) which have been used to enhance SAR images for land cover classification in many studies (Anys, 1995; Clausi, 2002; Herold et al., 2004; Peng et al., 2005). Peng found that using texture information increase the overall accuracy from 25.6% to 57.6% for a mountainous region in southern Patagonia. In addition, Peng found that using elevation, slope and aspect data from a DEM increases the overall land cover classification significantly.

In the last 10-15 years there has been an increase in the use of ensemble classifiers instead of using a single classifier (Waske and Braun, 2009). An ensemble classifier is when multiple models are run, and the results of the models are combined to form a consensus view on the classification result, often by a simple majority vote. One of the most well-known ensemble classifiers is random forests (Breiman and Cutler, 2001). Random forests were used to classify multi-temporal SAR imagery with 79.8% and 83.8% overall classification accuracies for two test sites in Bonn and Jena (Waske and Braun, 2009). In the PhD thesis of Waske (2007), where different classification algorithms to classify multisource imagery were compared, the final conclusion of the thesis was the following:

*“In summary, the following main subjects are worth to investigate in detail: (1) the derivation of additional temporal information from high temporal resolution imagery, such as RapidEye and TerraSAR-X data, (2) the integration of textural information, e.g. derived from spatial high resolution SAR data, as TerraSAR-X or Cosmo-SkyMed, (3) the utilization of polarimetric satellite imagery, as provided by ALOS PALSAR and TerraSAR-X, and (4) the use of multi-frequency approaches, using for example, ALOS PALSAR, TerraSAR-X or Cosmo-SkyMed, and Radarsat-2 data.”* (Waske, 2007)

This study will address points 2, 3 and 4 mentioned above by incorporating textural information, using features from polarimetric satellite imagery (although not

polarimetric decompositions), and the use of multi-frequency SAR imagery, in the context of semi-arid and forested biomes of Africa.

## 2.1 SAR land cover classification

Land cover classification is one of the core applications of remote sensing. Land cover classification has been studied since the advent of satellites in orbit around the earth. The ability to classify what is observed in a satellite image is very valuable and can assist decision makers to make informed decisions regarding a particular field of interest for a particular area of the world. This could vary in scale from planning and monitoring of large regions for carbon emissions e.g. as part of the United Nations' program on Reducing Emissions from Deforestation and forest Degradation (UN-REDD) (FAO et al., 2008), to land cover classification of a small area before and after a specific disaster event (Voigt et al., 2007).

The generation of accurate land cover maps is about interpreting the satellite signal correctly, whether from optical or SAR images. A standardised classification system sets a standard for decision makers who interpret the classified land cover map. Several land cover classification schemes have been defined over the last 35 years, since the dawn of widely used remote sensing technology. The first was the *Land Use and Land Cover Classification System for Use with Remote Sensor Data* (Anderson et al., 1976), developed by the United States Geological Survey (USGS). To establish a system which is transferable and applicable to any mapping scale or possible land cover type, the Land Cover Classification System (LCCS) has been developed by the Food and Agriculture Organization of the United Nations (FAO) and the United Nations Environment Programme (UNEP) (Di Grigorio, 2005). In this current study, the land cover classes identified are related to the LCCS framework. A more detailed description of LCCS is covered in Section 2.1.1.

Applications of land cover classification range depending on the scale and size of the area being studied. This ranges from very high resolution optical imagery, which focuses



at very fine detail (<1m) on a small area e.g. mapping urban areas (Pacifiçi et al., 2009), to coarser 500m to 1km resolution imagery used for global land cover mapping (Loveland & Belward 1997; Bartholomé & Belward 2005). This research will focus on high resolution (5-30m pixel spacing) land cover mapping from SAR satellites. This is therefore a local scale, which is often limited to the size of the swath of the high resolution satellite image. High resolution SAR land cover mapping can be used in the context of emergency response situations for flood monitoring (Martinez & Le Toan 2007; Cruz 2010; Pulvirenti et al. 2011), burn scar mapping (Ruecker & Siegert 2000), oil spills (Solberg et al., 2007), landslides and earthquakes (Massonnet and Feigl, 1998; Metternicht et al., 2005) as well as for carbon inventories (Gibbs et al., 2007).

SAR land cover classification has the potential to be used within the Copernicus programme, which is the European Earth Observation program (EuropeanCommission, 2013). Two of the services are already operational, namely Land Monitoring ([land.copernicus.eu](http://land.copernicus.eu)) and Emergency Management ([emergency.copernicus.eu](http://emergency.copernicus.eu)). The main satellite data source for Copernicus will be the Sentinel satellites (Malenovský et al., 2012), of which Sentinel 1A (SAR, C-band) (Torres et al., 2013) was launched on 3 April 2014 and Sentinel 2A (high resolution optical, multispectral) (Drusch et al., 2012) was launched on 23 June 2015. The Copernicus core services and downstream services presents a significant opportunity for utilising SAR and multi-frequency SAR for land cover classification applications.

SAR satellites are relatively unaffected by atmospheric conditions such as rain or cloud in most conditions and are therefore some of the most useful systems for emergency response situations, allowing a fast response time for mapping a disaster area (Voigt et al., 2007). The monitoring of emergency situations is crucial to effectively give aid to people affected by the disaster on the ground. The effect of disasters in developing countries is much greater than in developed countries as there are proportionally many more associated casualties in developing countries (Peduzzi et al., 2009). The Copernicus / GMES pre-operation emergency response projects, Respond (ESA, 2011) and Services and Applications For Emergency Response (SAFER) (SAFER, 2012), focused on providing

reference mapping during the first stages of a disaster, but also throughout the monitoring and reconstruction phases of the disaster. Land cover mapping and classification systems can be used for reference mapping and change detection of pre and post disaster events.

A further application of SAR to land cover mapping is in estimating the amount of above-ground biomass (AGB) in the vegetation. AGB can then be correlated to an estimate of total carbon content of the vegetation using ecosystems specific conversions (Patenaude et al., 2005). This is particularly useful in the context of UN-REDD and REDD+. REDD+ is an extension to UN-REDD, and include the focus on sustainable management of forests, the role of conservation and enhancement of forest carbon stocks, going further than deforestation and forest degradation (Gibbs et al., 2007; FAO et al., 2008; UNFCCC, 2011). It has been shown in several studies that there is a strong relationship between SAR backscatter and AGB (Lucas and Armston, 2007; Lucas et al., 2010; Mitchard et al., 2011; Carreiras et al., 2012) and that the best polarisations to use for biomass retrieval are L-band HV (Mitchard et al. 2011; Tsui et al. 2012) . This study will not focus directly on estimating biomass, but it is good to know that land cover classes relate to different biomass estimates, and are therefore related. AGB of semi-arid areas was estimated by Eisfelder et al. (2012) and can be used to refine land cover classification.

### 2.1.1 Land cover classification system (LCCS)

LCCS was initially developed through the AFRICOVER project (Di Gregorio and Latham, 2002). The basic principle of LCCS is that a land cover class is defined by a set of independent classifiers. As more classifiers are added, a more detailed class is defined. Therefore the emphasis is no longer on the class name, as in most other classification systems, but on the set of classifiers used to classify the land cover class.

The first step of LCCS is to identify the land cover class from a list of eight broad classes, called the dichotomous phase, given in Figure 2.1.

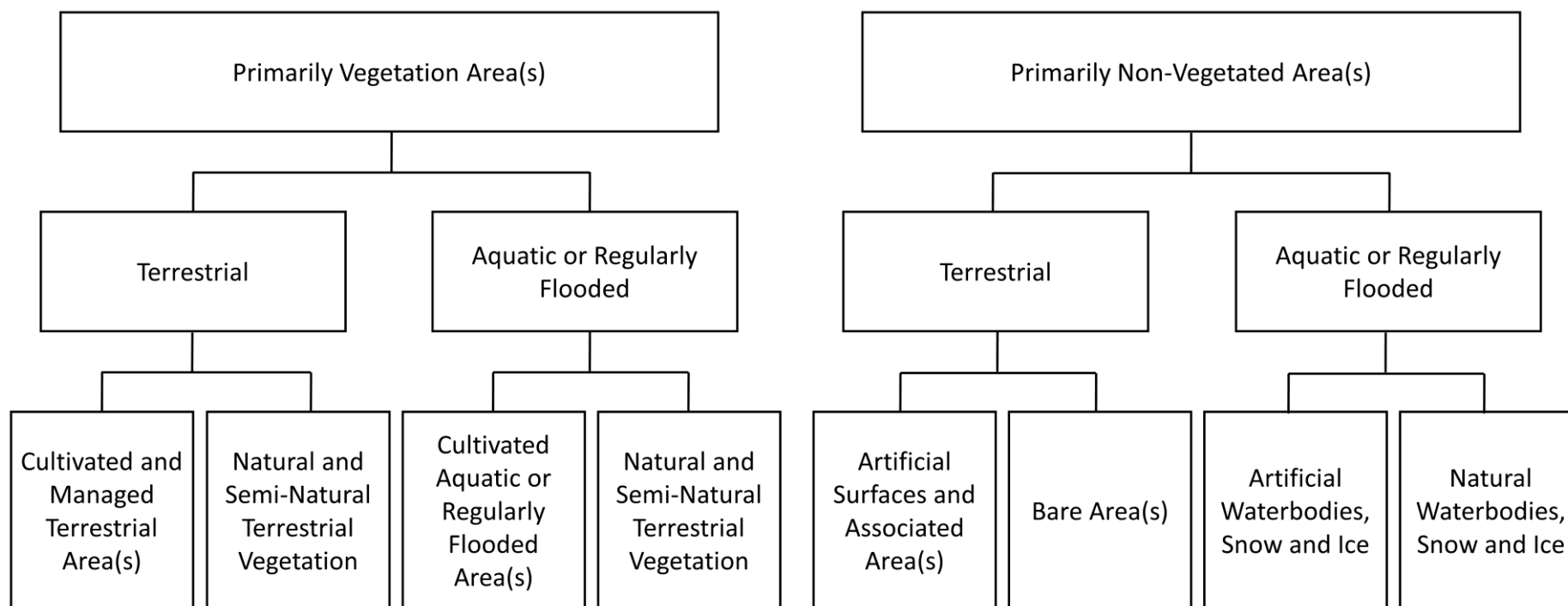


Figure 2.1 The LCCS eight broad land cover classes from the dichotomous phase of the LCCS land cover classification.

LCCS is implemented in a software package where the user can step through as many steps necessary to classify the land cover class. The second phase is called the modular-hierarchical phase, where more detail is added to refine the specific land cover class (Di Grigorio, 2005). For the dichotomous class of natural and semi-natural terrestrial vegetation, the modular-hierarchical phase drills down to six more levels as listed in Table 2.1.

*Table 2.1 The six levels of the modular-hierarchical phase of the natural and semi-natural terrestrial vegetation dichotomous class of LCCS.*

Level	Description
1	Life form and cover / Height / Spatial distribution
2	Leaf type and life cycle / Leaf phenology
3	Stratification
4	Landform / Lithology and Soils
5	Climate / Altitude / Erosion
6	Floristic Aspect

Similarly for each of the eight broad dichotomous classes, LCCS allows the user to add as many aspects of the class as deemed necessary to describe the land cover class to be classified. The land cover classes that were extracted individually from the SAR data for this research were closed woody vegetation (referred to as *dense trees*), open woody vegetation (*open woodland*), sparse woody vegetation / herbaceous closed to open vegetation (*sparse vegetation*), bare soil and/or other unconsolidated materials (*bare soil*), built up areas (*settlements*), artificial and natural waterbodies (*water*) and herbaceous crops (*agriculture*). This is expanded on in Section 3.4.

## 2.2 Introduction to principles of Synthetic Aperture Radar (SAR)

The beneficial properties of SAR remote sensing are the ability of SAR sensors to observe the land surface through cloud cover and that it uses its own power source, independent of sun illumination (i.e., it can image throughout day and night). There is, however, an

influence of the atmosphere on shorter wavelengths (<4cm, i.e. X-band). Rainfall, and especially heavy downpours, has a considerable influence on short wavelength radar (X-band and shorter wavelengths), but has less of an effect on longer wavelengths (C-band and longer wavelengths) (Lillesand et al., 2004). Nonetheless, the ability of SAR sensors to observe through cloud cover remains one of its biggest benefits over optical systems which are completely hindered by clouds (Kuntz, 2010).

Here follows a brief background on the theory of SAR from Ulaby and Dobson (1989), Lewis et al. (1998) and Woodhouse (2006). Radar remote sensing is the measurement of electromagnetic waves in the microwave part of the electromagnetic spectrum, which covers frequencies < 40000 MHz or wavelengths > 1cm. Electromagnetic wavelength ( $\lambda$ ) and frequency ( $f$ ) are inter-related by Equation 2-1:

$$c = f\lambda \quad \text{Equation 2-1}$$

where  $c$  is the speed of light, given by  $2.99792458 \times 10^8 \text{ ms}^{-1}$ . An outline of radar frequency bands, with corresponding frequency and wavelength ranges, is given in Table 2.2 (Lewis et al. 1998; Barrett et al. 2009).

*Table 2.2 Range of radar frequency bands with their corresponding frequency and wavelength ranges (Lewis et al. 1998; Barrett et al. 2009).*

<b>Radar frequency band</b>	<b>Frequency, <math>f</math> (GHz)</b>	<b>Wavelength, <math>\lambda</math> (cm)</b>
P	0.22-0.39	136-77
UHF	0.3-1	100-30
L	1-2	30-15
S	2-4	15-7.5
C	4-8	7.5-3.75
X	8-12.5	3.75-2.4
Ku	12.5-18	2.40-1.67
K	18-26.5	1.67-1.18
Ka	26.5-40	1.18-0.75
mm	>40	<0.75

Table 2.2 shows the SAR sensors that are currently in orbit, along with those approved and planned for in the next seven years. The SAR frequencies covered are P-, L-, C- and

X-band. The principal owner, polarisation, revisit time, swath width, spatial resolution and expected mission life from 2011 onwards is included in this table. This table shows the potential to utilise multi-frequency SAR into the foreseeable future. Two satellites to mention from this table are Sentinel-1A and NovaSAR-S. The Sentinel-1A satellite is the first of a series of operational satellites as part of the Copernicus programme. NovaSAR-S will be an S-band satellite developed and built by Surrey Satellite Technology Limited (SSTL) based in the United Kingdom.

The acquisition geometry of a SAR system is given in Figure 2.2. SAR refers to imaging radar which requires image pre-processing to synthesize a much longer effective radar antenna length to construct an image. This improves the resolution in the azimuth direction (Woodhouse 2006). The incidence angle ( $\theta$ ) is the angle between the radar wave and the vertical.

Satellite based SAR are active microwave instruments which transmit a pulse at a specific frequency and measures the signal that is scattered back to the instrument. The received power ( $P_r$ ) is related to the transmitted power ( $P_t$ ) by the radar equation:

$$P_r = \frac{P_t G A_e}{(4\pi R^2)^2} \sigma \quad \text{Equation 2-2}$$

where  $G$  is the antenna gain,  $R$  is the slant range distance between the instrument and the measured object,  $A_e$  is the effective area of the antenna and  $\sigma$  (sigma) is the radar cross section of the measured object. The backscatter coefficient  $\sigma^0$  (sigma-nought) is defined as the radar cross section per unit area. The  $\sigma^0$  values are essentially what the SAR instrument measures, and are used to determine biophysical parameters from SAR measurements.

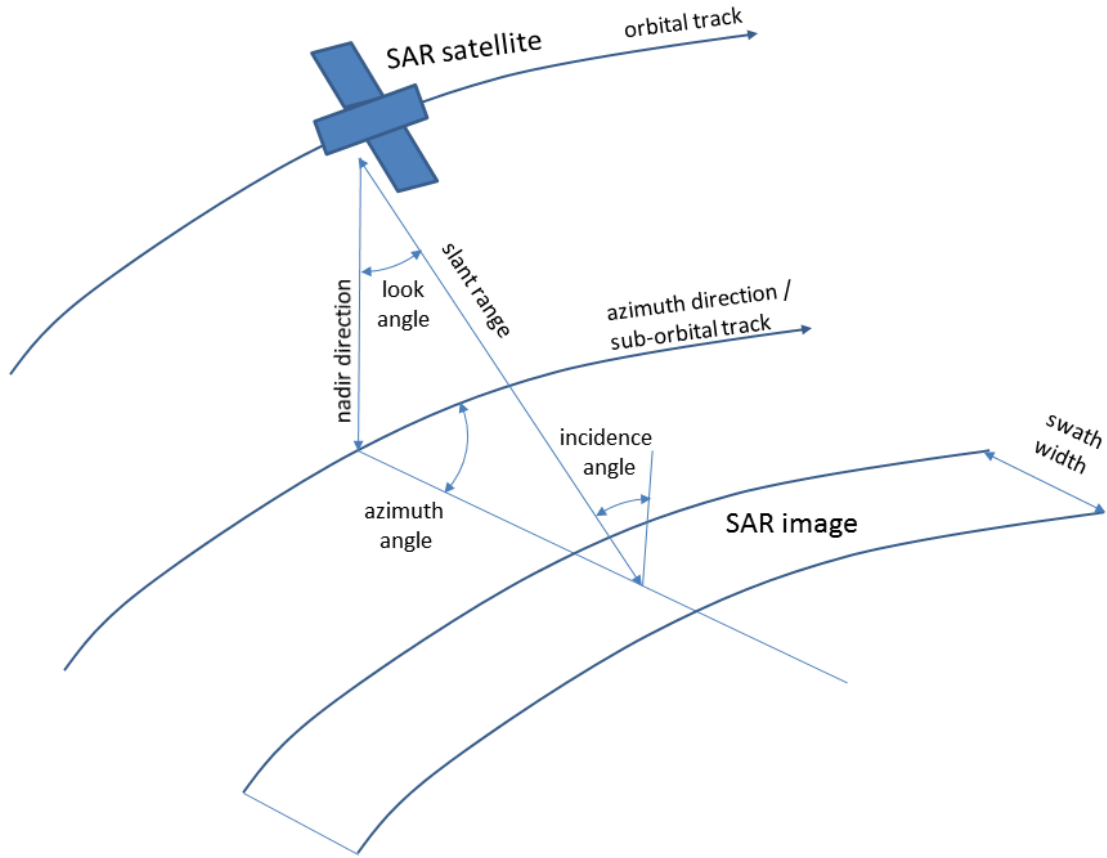


Figure 2.2 Acquisition geometry of a satellite SAR system, amended from Lewis et al. (1998).

Another important concept when dealing with SAR remote sensing images is polarimetry. All SAR sensors transmit the radar pulse in a certain polarization. This can be either horizontal (H) or vertical (V) with respect to the slant range direction. Many SAR sensors can also receive the signal in either H or V direction which leads to backscatter measurements of  $\sigma_{hh}^0$ ,  $\sigma_{vv}^0$ ,  $\sigma_{hv}^0$  and  $\sigma_{vh}^0$ . The representation used most commonly to describe fully polarimetric SAR measurements is the Sinclair scattering matrix [S], which is given by

$$[S] = \begin{bmatrix} S_{HH} & S_{HV} \\ S_{VH} & S_{VV} \end{bmatrix} \quad \text{Equation 2-3}$$

where matrix [S] is symmetrical with  $S_{HV} = S_{VH}$ . Different scatterers on the ground surface modify the polarisation to varying degrees, and the polarisation mode therefore influences how the object looks on the resulting imagery (Lillesand et al., 2004). A good overview of recent advancements in polarimetry is given by Lee and Ainsworth (2010).




Table 2.3 SAR satellites in orbit, approved and planned for 2011-2022, for potential SAR applications.

SAR missions <sup>1</sup>	Principal Owner	Polarisation	Revisit time (days)	Swath width* (km)	Spatial res.* (m)	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
<b>P-Band SAR</b>																	
BIOMASS <sup>2</sup>	ESA	full	25-45d	40-60	50												
<b>L-Band SAR</b>																	
ALOS PALSAR <sup>3</sup>	JAXA	full	46	40-70	7-44												
ALOS-2 <sup>4</sup>	JAXA	full	14	25-70	3/6/10												
SAOCOM 1A <sup>5</sup>	CONAE (Arg)	full	16	30-350	10-100												
SAOCOM 1B <sup>5</sup>	CONAE (Arg)	full	8, A&B	30-350	10-100												
<b>S-Band SAR</b>																	
HJ-1C (Huan Jing) <sup>6,7</sup>	CRESDA (China)	VV	4	40-100	5-25												
NovaSAR-S <sup>8</sup>	SSTL (UK)	full	01-Apr	15-20	6												
<b>C-Band SAR</b>																	
Envisat (ASAR) <sup>1</sup>	ESA	DualPol	35	56-100	30												
Sentinel 1A <sup>9,10</sup>	ESA	DualPol	12	80-250	5-20												
Sentinel 1B <sup>9,10</sup>	ESA	DualPol	6, A&B	80-250	5-20												
Sentinel 1C <sup>1</sup> , ...	ESA	DualPol	6	80-250	5-20												
Radarsat-2 <sup>11,1</sup>	MDA	full	24	20-100	1x3												
Radarsat Constellation Mission (RCM) <sup>12</sup>	MDA/CSA	full	1	5-125	1x3												
RISAT-1 <sup>13</sup>	ISRO (India)	full	25	30-240	2-6												
<b>X-Band SAR</b>																	
TerraSAR-X <sup>1,14</sup>	DLR / Astrium	DualPol	11	10-50	1-3												
TanDEM-X <sup>15</sup>	DLR / Astrium	DualPol	11	10-50	1-3												
TerraSAR-X2 <sup>16</sup>	DLR / Astrium	full	11	5-24	0.25-3												
Cosmo-Skymed-1,2,3 <sup>1,17</sup>	ASI (Italy)	single/AP	3-36hr	10-40	1-5												
Cosmo-Skymed-4 <sup>1,17</sup>	ASI (Italy)	single/AP	2-12hr	10-40	1-5												
Cosmo-Skymed 2nd Gen A,B <sup>1</sup>	ASI (Italy)	single/AP	2-12hr	10-60	1-4												
SeoSAR/PAZ <sup>18,1</sup>	Hisdesat	DualPol	11, 5 (+ TSX)	10-30	1-3												
SeoSAR/PAZ-2 <sup>1</sup>	Hisdesat	DualPol	?	?	?												

\*Swath width and spatial resolution for high-resolution imaging modes.

Wider swaths and lower spatial resolution also available.

AP: Alternate Polarisation

 In orbit  
 Approved  
 Planned

ASI – Italian Space Agency (Agenzia Spaziale Italiana), CONAE – Argentinian Space Agency (Comision Nacional de Actividades Espaciales), CRESDA – China Center for Resource Satellite Data and Application, CSA – Canadian Space Agency, DLR – German Aerospace Centre (Deutsches Zentrum für Luft- und Raumfahrt), ESA – European Space Agency, ISRO – Indian Space Research Organisation, JAXA – Japan Aerospace Exploration Agency, MDA – MacDonald, Dettwiler and Associates Ltd., Canada, SSTL – Surrey Satellite Technology Limited, UK



References for Table 2.2:

<sup>1</sup>(Aschbacher and Milagro-Pérez, 2012), <sup>2</sup>(ESA, 2013a; Le Toan et al., 2011), <sup>3</sup>(JAXA, 2011), <sup>4</sup>(JAXA, 2013a, 2013b), <sup>5</sup>(CONAE, 2011), <sup>6</sup>(Huang et al., 2010), <sup>7</sup>(Barbosa, 2012), <sup>8</sup>(SSTL, 2011), <sup>9</sup>(ESA, 2013b), <sup>10</sup>(Torres et al., 2012), <sup>11</sup>(CSA, 2011), <sup>12</sup>(CSA, 2013), <sup>13</sup>(Chakraborty et al., 2013), <sup>14</sup>(Werninghaus and Buckreuss, 2010), <sup>15</sup>(Zink, 2012), <sup>16</sup>(Janoth et al., 2012), <sup>17</sup>(Covella et al., 2010), <sup>18</sup>(Hisdesat, 2013)

## 2.3 Key factors that influence SAR backscatter

The key factors that influence SAR backscatter are terrain geometry; the physical structure of the scatterers on the surface of the earth, such as vegetation structure; and the dielectric constant, which is related to water content, of the scatterers on the surface of the earth (Lewis et al., 1998). The SAR signal can therefore be related to the biophysical parameters of the scatterers on the ground, such as biomass in vegetation and soil moisture.

### 2.3.1 Terrain geometry

The terrain geometry affects the SAR signal by modifying the incidence angle ( $\theta$ ) of the SAR signal by the effect of a local incidence angle ( $\theta_l$ ). The local incidence angle is influenced by the local slope of the terrain (Figure 2.3). The measured backscatter coefficient ( $\sigma^0$ ) varies with relation to  $\theta$ , as the scattering geometry changes. For all polarisations the SAR backscatter coefficient values decrease as the incidence angle increases (Ulaby and Dobson, 1989). This can be explained in that there is a larger observed backscatter to the sensor at a steeper incidence angle, and a lower observed return backscatter at a more oblique incidence angle.

To be able to extract biophysical parameters from a SAR image, the effect of terrain geometry first needs to be removed. This is done during the orthorectification steps to rectify an image using a DEM. This is particularly necessary for images with hilly terrain, where the effect of different angled slopes are greatest. Although the effect of terrain geometry is was minimal for the sites used in this research, due to the relatively flat terrain, a SRTM DEM was still used during the orthorectification steps of the SAR images.

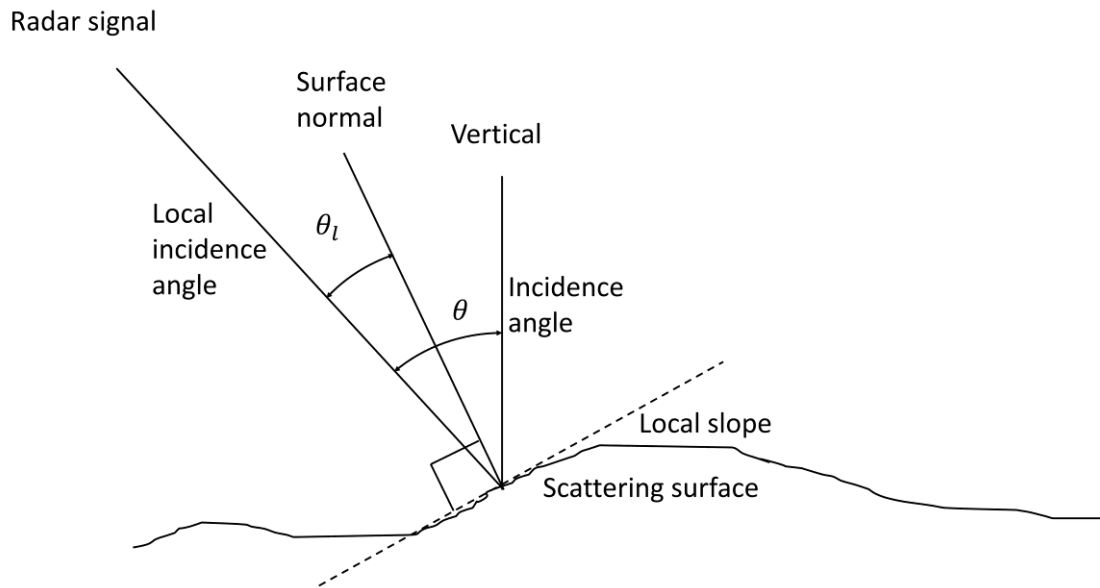


Figure 2.3 The comparison between the incidence angle  $\theta$  and the local incidence angle  $\theta_l$ .

### 2.3.2 Surface roughness and vegetation structure

There are three main scattering mechanisms at work between that influences the SAR signal as it interacts with the surface and vegetation: surface scattering, volume (or canopy) scattering and double bounce scattering (Freeman and Durden, 1998) (see Figure 2.4).

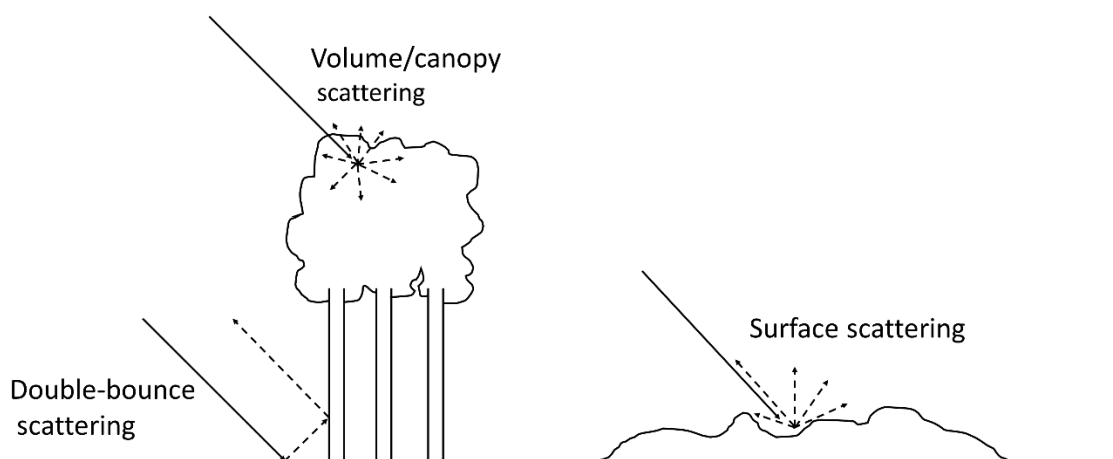


Figure 2.4 Scattering mechanisms affecting the SAR signal: volume, surface and double-bounce scattering.

Surface scattering is affected by the roughness of the surface in relation to the SAR frequency. A rough surface in relation to an X-band and therefore be a smooth surface in relation to a longer SAR signal, such as L-band. Volume scattering is the interaction of the SAR signal where are large volume of vegetation is combined together. This is especially prevalent to dense forest areas, but also shorter vegetation such as shrubs and grassland. Double-bounce scattering is where the SAR signal interacts with objects orthogonal to the surface, such as the vertical stems of trees or the sides of buildings.

The SAR signals returned are often a combination of the scattering mechanisms. In a forest there are SAR backscatter returned from the canopy, through volume scattering, and double-bounce from the ground and the stems of the trees. For vegetation with smaller stems, such as shrubs or tall grass, a combination of volume scattering and surface scattering is returned.

However, since the scattering mechanisms are frequency dependent, different scattering interactions can be separated to a large degree. Within the forest example, the X-band will interact largely with the canopy, through volume scattering, whereas longer wavelengths such as L-band will penetrate the canopy, and return scattering from the stems and large branches through double bounce scattering and some surface scattering.

### 2.3.3 Dielectric properties / moisture content

The backscatter coefficient  $\sigma^0$  is determined by both the surface roughness and the dielectric properties of the surface. The dielectric constant is heavily influenced by the moisture content of the material. One of the main challenges in retrieving soil moisture measurements from SAR  $\sigma^0$  values is therefore to separate the influence of surface roughness from the influence of soil moisture on the  $\sigma^0$  values (Paloscia, 2012; van der Velde et al., 2012).

The reflectivity of a rough surface increases as a response to a higher dielectric constant. Therefore, e.g. a wet agricultural field will result in a brighter image during and after rainfall. The effect of increased moisture on the surface reduces surface roughness in addition to increasing its dielectric constant. As a result this leads to greater forward scattering, with a smaller part of the signal reflected back to the SAR sensor (Raney, 1998). The increased moisture has therefore a two-fold effect, and depending if the increased dielectric constant or the resulting smooth surface is more prevalent, will determine whether it is a lighter or darker area on the SAR image.

For volume scattering, increased volume moisture such as a wet canopy, increases internal reflections towards, but also away from the SAR sensor. Additionally, as observed by the SAR sensor, the total reflection includes the surface scattering from the surface as well as the volume scattering from the canopy (Raney, 1998). In general, in forested areas the effect of rainfall events before or during the SAR acquisition leads to an increased backscatter scatter (Lucas et al., 2010).

## 2.4 SAR sensor characteristics

A brief introduction to each of the SAR sensors used in this study is presented in this section. Images from ALOS PALSAR, ENVISAT ASAR and TerraSAR-X were used in this study.

### 2.4.1 ALOS PALSAR

The Phased Array L-band Synthetic Aperture Radar (PALSAR) instrument was on board the Japanese Space Exploration Agency's (JAXA) Advance Land Observing Satellite (ALOS) launched in January 2006 (Rosenqvist et al., 2007), and came to the end of its life after five years in May 2011 (JAXA, 2011). The ALOS satellite had a 46-day repeat cycle and the PALSAR instrument has 4 acquisition modes: Fine beam mode, which was split into Fine beam single polarization (FBS) (HH or VV) or Fine beam double polarization (FBD) (HH/HV or VV/VH), Polarimetric or Quad-polarization mode (POL)

(HH/HV/VH/VV), ScanSAR mode and Direct transmission mode (Rosenqvist et al., 2007). The data used in this study were acquired using the FBS, FBD and POL modes. The swath width for the FBS mode is 70km with 10m x 10m spatial resolution on the ground. The swath width for the FBD mode is 70km with 20m x 10m spatial resolution, and for the POL mode, a swath width of 30km with 31m x 10m spatial resolution. ALOS PALSAR has a variable off-nadir incidence angle of 9.9° - 50.8° for FBS and FBD modes, and 9.7° - 26.2° for the POL mode (Rosenqvist et al., 2007).

One of the main products developed from the ALOS PALSAR images is the Kyoto and Carbon Initiative global mosaic (Shimada and Ohtaki, 2010). The processing of PALSAR scenes into a mosaic of the African continent was undertaken by the European Commission's Joint Research Centre (JRC) (Rosenqvist and De Grandi, 2009; De Grandi et al., 2011). Additionally, the full polarimetric ability from the ALOS PALSAR POL mode has been investigated in several studies (Ainsworth et al., 2009; Bagan et al., 2012).

ALOS-2 was launched on 24 May 2014 and carries the PALSAR-2 instrument, with upgrades that include a much shorter revisit cycle of 14 days and new modes of ultra fine single mode (UFS) at 3m spatial resolution and high sensitive polarimetric mode (HSPol) at 6m spatial resolution, to name a few (Rosenqvist et al., 2014).

#### 2.4.2 ENVISAT ASAR

ENVISAT is one of ESA's most ambitious satellites to fly in space to date, with ten instruments on board and weighing eight tons (Gardini et al., 1995). The satellite was launched in March 2002 and contact with the satellite was lost on 8 April 2012 (ESA, 2012a). One of the instruments on board is the Advanced Synthetic Aperture Radar (ASAR), which is a C-band SAR instrument. ASAR followed on the previous European SAR sensors ERS-1 and ERS-2. The ASAR instrument has five acquisition modes (Desnos et al., 2000), of which the Image mode (IM) (VV or HH) and Alternating polarisation mode (AP) (VV/HH, HH/HV or VV/VH) were used in this study. Both the IM and AP modes present spatial resolution of about 30m x 30m. The remaining three modes are Wide swath, Global monitoring and Wave mode.

With data acquired from ten sensors on board ENVISAT during ten years in orbit, numerous applications were researched. A selection of applications using ASAR data are flood extent mapping (Henry et al., 2006; Marti-Cardona et al., 2010), oil spill detection (Solberg et al., 2007), landslide detection (Hölbling et al., 2012) and land cover mapping (Park and Chi, 2008).

To continue the global archive of C-band SAR data of nearly 25 years (since 1991) from ERS-1, ERS-2, ENVISAT ASAR, RADARSAT-1 and RADARSAT-2, the next generation of C-band SAR satellites are Sentinel 1A and Sentinel 1B. As mentioned in Section 2.1, the Sentinel satellites are the main data source for the European Earth Observation programme Copernicus. One of the major improvements of the Sentinel data, compared to its predecessors is that the data are free and accessible to the public (ESA, 2012b). This is similar to NASA's Landsat data policy which is used extensively all around the world as a results of its free and open data policy.

#### 2.4.3 TerraSAR-X

TerraSAR-X, Germany's first commercial X-band satellite, was launched in June 2007 and was developed as part of a public-private partnership between the DLR and EADS Astrium GmbH (Werninghaus and Buckreuss, 2010). For interest, EADS Astrium has, since July 2013, been re-branded as Airbus Defence and Space. EADS Astrium developed, built and launched the satellite, while the DLR developed the corresponding ground segment and the science service segment to provide data to the science community. TerraSAR-X offer 4 imaging modes, namely Stripmap mode single or dual polarization (at 3.3m or 6.6m azimuth resolution), High Resolution Spotlight mode in single or dual polarization (at 1.1m or 2.2m azimuth resolution), Spotlight mode in single or dual polarization (at 1.7m or 3.4m azimuth resolution) and ScanSAR mode in single polarization (at 18.5m azimuth resolution) (Fritz and Eineder, 2010).

The TanDEM-X satellite is a replication of the TerraSAR-X satellite and was launched in June 2010 (Martone et al., 2013). TanDEM-X is in the similar orbit as TerraSAR-X and able to fly in different formations, which enables interferometric data acquisition. The primary objective of the TanDEM-X mission was to complete a global DEM of unmatched

accuracy, quality and coverage. The processing of the global DEM data were completed in 2014, and is commercially available as the WorldDEM™ product from Airbus Defence and Space (Riegler et al., 2015). The WorldDEM product has a relative vertical accuracy of 2m and is available at 12m x 12m spatial resolution. This is a large improvement from the SRTM DEM which had a relative vertical accuracy of 12m and was available at 30m x 30m (90m x 90m before September 2014) spatial resolution (Martone et al., 2013; JPL, 2014).

A few applications of TerraSAR-X related to land cover classification are mapping soil texture (Zribi et al., 2012), burn severity (Tanase et al., 2010a), sugarcane fields (Baghdadi et al., 2010) and flood mapping (Cruz, 2010).

## 2.5 Multi-frequency SAR

Early studies using multi-frequency SAR are from Chen et al. (1996), who demonstrated using a neural network as a classification algorithm for multi-frequency SAR data. Images from ERS-1 (C-band) and JERS-1 (L-band) were used for land cover classification of different vegetation structures using hierarchical knowledge-based decision rules (Dobson et al. 1996). This was achieved through a manual process of decision rule selection, assisted by modelling the SAR backscatter values in the Michigan microwave canopy scattering model (MIMICS).

The combination of using more than one SAR frequency in land cover classification has been explored in several studies. The combination of L- and X-band for AGB retrieval for a forest area of central Kalimantan was successfully used by Englhart et al. (2011). L-band and C-band were combined in a study of land cover classification over central Africa, with a focus on Malawi (Holecz et al., 2009). The influence of burn scars on co- and cross-polarised L-, C-, and X-band data were compared in Tanase et al., (2010b). The highest sensitivity to burn severity was found to be at L- and C-band.

In a study on large-scale mapping of central Africa, L- and C-band SAR data were used to map forest extent within the Congo basin (Mayaux et al., 2002). The study concluded that, although the SAR data can provide thematic information of the forest composition for the whole ecosystem, SAR data alone could not consistently map all the vegetation features, such as secondary forests, in this tropical ecosystem. A synergistic use of SAR and optical data were proposed for future regional scale land cover projects. It is noted here that in this case, the SAR image data were downscaled to 100m pixel spacing, so the conclusion might be different with higher resolution SAR images.

Most of the examples thus far have only made use of single frequency data; most regularly C-band, but also X- and L-band. Even though there are SAR satellites of different frequencies in orbit and even more planned for the future (Table 2.2), synergies with other frequencies are not regularly used or part of the current Copernicus core service activities.

Another innovative way to make use of multi-frequency SAR was suggested by Eckardt et al. (2013). Here it is proposed to use data from multi-frequency SAR satellites to supplement the optical thick cloud cover areas in optical imagery, using a closest feature vector. This was used to fuse the SAR and optical imagery together, in the areas of the optical image which are affected by clouds. Eckardt et al. also used mono-temporal, multi-frequency data.

The satellites mentioned in Table 2.2 show the various data opportunities for potential applications of multi-frequency SAR, by combining data from different SAR sensors. Over the next 5 years, there is the potential to obtain data from L-, S-, C- and X-band SAR sensors, over the same geographical area, and relatively close in time. The addition of the BIOMASS P-band SAR mission from 2020 onwards adds another frequency to the possible SAR images available. It is notable that the spatial resolution keeps improving with the newer satellites, and more satellites are being used in constellations to improve revisit times.



## 2.6 Classification algorithms

Classification methods for remote sensing data are usually taken from the machine learning / pattern recognition methods for various statistical applications. Some of the most commonly used methods are the standard supervised maximum likelihood classifier (MLC), decision trees and artificial neural networks (ANN) (Jensen, 2005). More recently, techniques applied to multisource remote sensing data that have proved superior to the standard supervised classification techniques include SVMs (Huang et al. 2002) and random forests (Rodriguez-Galiano et al., 2012b). SVMs and random forests are both non-parametric classification algorithms, similar to ANN and decision trees, which mean that the algorithms do not rely on the assumption that the input variables are from a specific statistical distribution. This is contrary to parametric classification algorithms, such as MLC, which are based on the assumption that the input variables are normally distributed (Moser et al., 2006). Non-parametric classification algorithms are therefore preferred for classifying multi-source data. Additionally, SAR backscatter values of different land cover classes do not all follow the same distribution functions (Ulaby and Dobson, 1989; Moser et al., 2006).

In the last 15 years much research has gone into ensemble classifiers. Ensemble classifiers are learning algorithms that construct a set of classifiers. New data points are then classified using a weighted vote between the different classifier results (Dietterich, 2000). Some of the most well established ensemble classifier methods are bagging, and boosting.

Bootstrap aggregation a.k.a. bagging, was introduced by Breiman (1996). Bagging works as follows: the training dataset are sampled multiple times by selecting a random sample with replacement from the training set, to produce  $m$  training datasets. The  $m$  newly sampled training datasets are known as bootstrap replicates of the original training dataset. The size of each bootstrap replicate is 63.2% of the training set, with instances that can appear more than once. A model is then trained on each of the  $m$  bootstrap replicates, to produce  $m$  model results. A majority vote on the  $m$  model results is used

to determine the overall classification result by creating a consensus view from all the trained models.

Boosting is similar to bagging, with the difference that instead of creating  $m$  models concurrently, each model is developed based on the results from the previous model. At the first iteration, all training instances are given the same weight. Then, with each iteration, the weight for each of the training instances are adjusted, based on the results from the previous model training. A larger weight is given to instances classified incorrectly in the previous model. This is done for  $m$  iterations, to derive a final set of weights, on which the final model is trained. This final model, based on the weights-distribution after  $m$  iterations, are then used to classify any new data. Boosting was first proposed by Schapire (1990) and Freund (1995). The most commonly used boosting algorithm named adaptive boosting or AdaBoost as presented by Freund and Schapire (1997, 1996). Based on the bagging and boosting techniques, a specific classifier was developed by Breiman (2001) using multiple decision trees, called random forests.

Therefore, based on the literature mentioned above, the classifiers compared in this study were random forests and SVMs. They are both techniques that outperform traditional classifiers in most cases. Random forests and SVMs are expanded on in the next sections.

### 2.6.1 Random forests

Random forests was formulated by Breiman (2001) and originally coded by Breiman and Cutler (2001). Random forests is essentially an ensemble model of many decision trees, where the dataset used to build each tree is randomly selected from the full training set, with replacement. At each random selection of the training data, one third of the instances are left out, referred to as the out-of-bag (OOB) sample. The OOB sample is used to get an unbiased estimate of the classification error as more trees are added to the forest. At each node, a small group of features are chosen at random, which are then available to be chosen by the decision tree algorithm, to split the tree on. The decision

trees are built to maximum size, with no pruning of the trees. Finally a majority vote is used between all the trees, to decide the final class for each instance. A schematic diagram is shown in Figure 2.5.

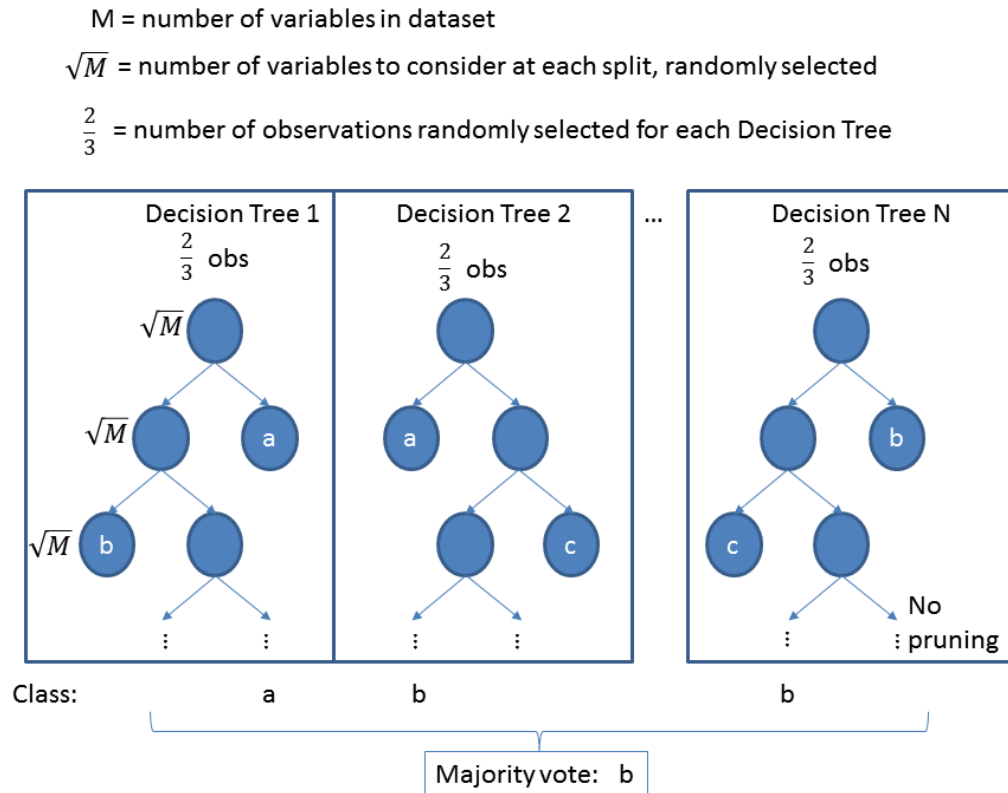


Figure 2.5 Schematic diagram of the random forest classifier.

The following pros and cons of random forests have been identified:

Pros (+) (Breiman, 2001):

- The accuracy of random forests is as good as and sometimes better than Adaboost.
- Random forests is relatively robust to outliers and noise.
- Random forests is faster than bagging or boosting.
- Random forests gives useful internal estimates of strength, error, correlation and variable importance.
- Random forests is simple and easy to parallelize on implementation.

Cons (-):

- The interpretability of random forest results can be difficult, since there are often hundreds of trees in the model, compared to a single decision tree (Prasad et al., 2006).
- The variable importance measures give a bias towards predictor variables that are highly correlated (Strobl and Zeileis, 2008).
- Cases were identified when a random forest model may overfit the training dataset, in the situation when there are many noisy variables to consider for the model (Segal, 2004).

The classification accuracies of land cover using multi-temporal SAR data of two sites in Bonn and Jena in Germany were compared by using standard decision trees, MLC, boosted decision trees and random forests in Waske and Braun (2009a). The results showed that random forests outperformed the standard classification techniques, including the boosted decision tree ensemble classifier.

The use of random forests in the classification of SAR images has been demonstrated in Loosvelt et al. (2012) and Corcoran et al. (2013). Naidoo et al. (2015) used random forests to classify AGB, CC and TCV from multi-frequency SAR images.

## 2.6.2 Support vector machines (SVM)

The mention of a SVM first appeared in (Vapnik, 1979), with the main paper calling SVMs support vector networks in (Cortes and Vapnik, 1995).

In short, SVMs is a non-parametric binary classifier that intends to find an optimal separating hyperplane between the two classes by maximizing the margin between the closest points of the classes, called the support vectors. Figure 2.6 shows a representation of an SVM for linearly separable classes. The support vectors and the separating hyperplane are highlighted in this figure.

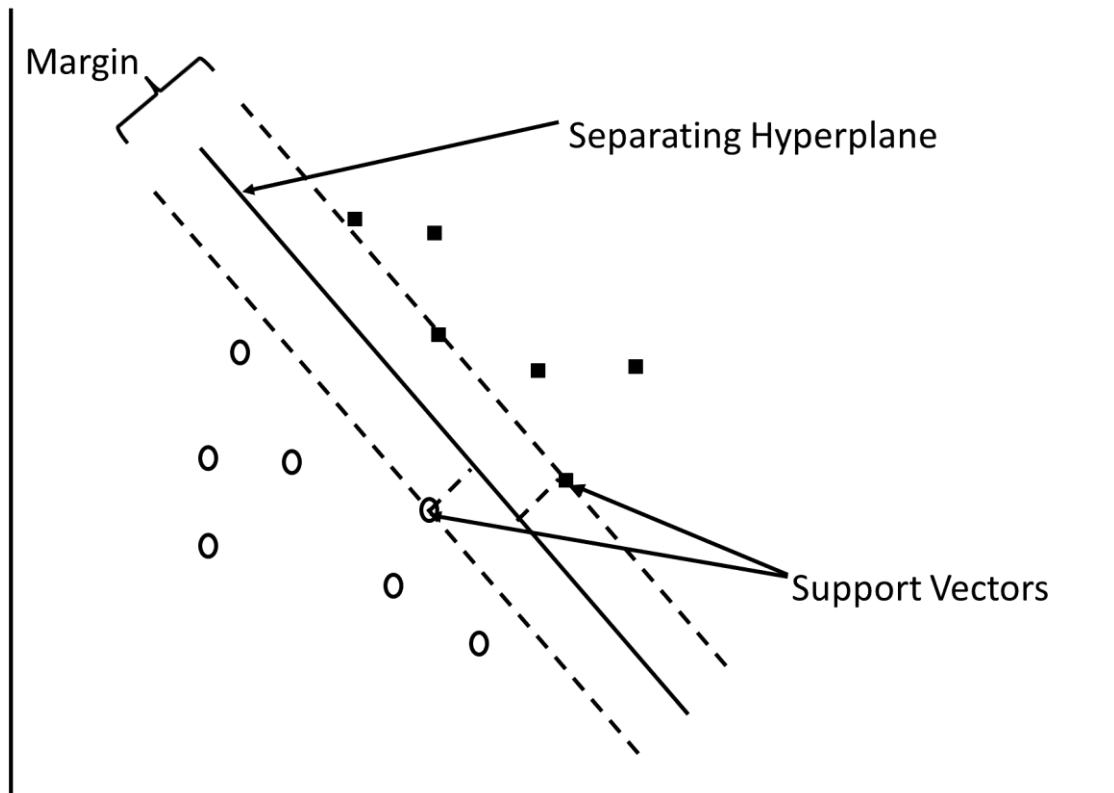


Figure 2.6 Classification of an SVM, for linearly separable classes.

For the situation where the classes are not linearly separable, the data points are mapped to a higher dimensional space using a kernel function, where the points are linearly separable. Figure 2.7 shows how the kernel function is applied to a dataset with classes that is not linearly separable.

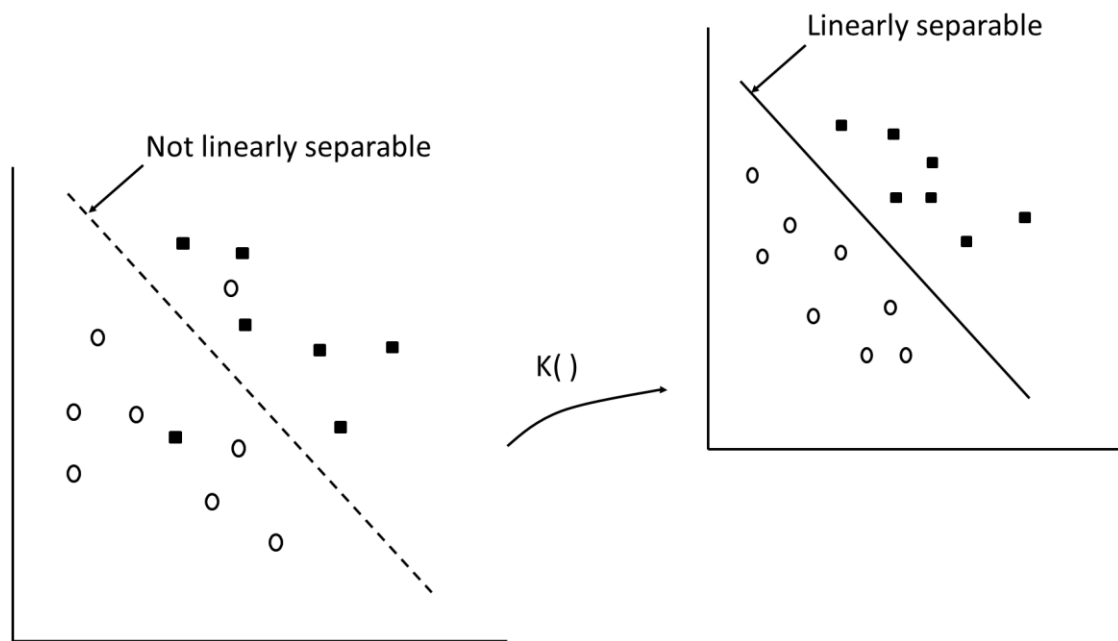


Figure 2.7 An SVM mapped to a higher dimension using kernel  $K()$  for non-linearly separable classes.

This problem can be formatted as a quadratic optimization problem, which can be solved by known optimization techniques. For SVMs, a regular implemented optimizer is the sequential minimization optimization algorithm, which solves the SVM quadratic problem without using any numerical quadratic optimization steps, and instead aims to solve the smallest possible optimization problem reducing the computation and memory requirements (Platt, 1998; Karatzoglou et al., 2006).

Since the support vectors are the training data points closest to the separating hyperplane, only a small amount of data are actually used to train the model. For a multi-class problem, SVMs use a technique called *one-to-many*, and then find the correct class by a voting mechanism.

A major implementation of SVMs is the library Libsvm (Chang and Lin, 2011). An implementation in the R statistical programming language is with the package called 'e1071' (Karatzoglou et al., 2006). There are many different kernel functions available for SVMs, namely Gaussian, polynomial, linear, sigmoid, Laplace, Bessel, analysis of variance (ANOVA), spline and user-defined. For the SVM implementation in the 'e1071'

package the Gaussian radial basis function (RBF), polynomial, linear and sigmoid functions are available.

The pros and cons of SVMs can be summarized as follows from Mountrakis et al. (2011):

Pros(+):

- SVMs generalize well from a small amount and / or quality of training data. For remote sensing this is especially important due to the cost of collecting ground truth data.
- As part of the optimization problem, SVMs are resilient against getting trapped in a local minimum and always get to global minima.
- There is no need for repeated training of the classifier using different random starting points or architectures.
- SVMs are non-parametric and therefore no assumption is made on the statistical distribution of the data.

Cons(-):

- The choice of which kernel function to use is not always known.
- The choice of the parameter value  $C$ , which denotes the trade-off between minimizing the training error and maximizing the margin, needs to be determined. This often leads to a trial and error approach.
- As dimensionality increases, SVMs typically have dimensionality issues such as how to address outliers and increased computational demands.
- Noisy data can be a problem for SVMs, as they are not optimized for noisy data with many outliers.
- The performance of an SVM can decrease as a result of only a small number of mislabelled training samples.

SVMs have been used in many studies during the last ten years to classify remote sensing images. One study where the SVMs focused on classifying AIRSAR images shows that SVMs perform much better than the supervised Wishart classifier (Lardeux et al., 2009).

The overall mean producer's accuracy using fully polarimetric P-, L- and C-band images were 99% for the SVM, compared to 78% for the Wishart classifier, for seven classes varying between four forest classes, two low vegetation classes and bare soil. This was for 5.5m spatial resolution for all three frequencies, with several polarimetric decompositions added to the feature space. In contrast to the Wishart classifier, no a priori data are needed with regards to the distribution of the different classes. For the SVM classifier, the Gaussian RBF kernel was used (Lardeux et al., 2009).

In another study, SVMs were used to determine the AGB of a study site in Borneo, using a variant of SVMs known as support vector regression (SVR), in comparison to ANN and multivariate linear regression (MLR). The Gaussian RBF kernel was used to train the SVR. For AGB estimation in the Indonesian forest, the ANN was preferred to SVR and MLR, and not the SVR classifier. This is due to AGB saturation levels being reached at about 260 t/ha for the SVR compared to the 400 t/ha for the ANN and MLR classifiers (Englhart et al., 2012). This shows the importance to compare several models for a specific application.

In Waske and Benediktsson (2007), a technique was shown to fuse data from different sources, in this case SAR and optical data. Each data source was first classified on its own using SVMs. Then, instead of combining the classification directly, it was combined using another SVM. This method of fusing the data with an SVM outperformed all other parametric and nonparametric methods tested. It also outperformed majority and absolute maximum voting methods to combine an ensemble of classifiers. Similar to the other SVM studies reviewed so far, the Gaussian RBF kernel was used as a kernel function.

## 2.7 Discussion of literature review

From the introduction and literature review it is shown that SAR has been used for land cover classification and some multi-frequency SAR applications have been attempted. Many research papers focus on classifying AGB (Ferrazzoli et al., 1997; Svoray and



Shoshany, 2002; Mitchard et al., 2009; Englhart et al., 2011) or soil moisture (Barrett et al., 2009; Baghdadi et al., 2012). Some studies have focused on using SAR for land cover, but most have either used interferometry (Strozzi et al., 2000; Okhimamhe, 2003; Park and Chi, 2008) or polarimetric decompositions (Qi et al., 2010; Niu, X., Ban, 2010; Bagan et al., 2012) to classify land cover.

In recent years, the use of ensemble classifiers and especially random forests have been proven to outperform traditional classification algorithms (Waske and Braun, 2009; Rodriguez-Galiano et al., 2012a). With the increase in available SAR data from satellite and from several frequencies: L-, C-, X-band, with the addition of S- and P-band in the near future, some researchers have investigated the combination of the SAR frequencies into multi-frequency SAR applications. Waske (2007) highlighted the use of classifier ensembles for land cover classification and mentioned that the use of texture measures and multi-frequency SAR data needs to be further investigated. Turkar (2012) used multi-frequency SAR, which included fully polarimetric data and polarimetric data for a study site in India. Texture measures were not included in Turkar's study. Texture measures were included in Pierdicca et al. (2011), but only L-band and X-band images were used. Multi-frequency SAR using C-band and L-band for land cover classification of an Arctic coastal ecosystem was studied in Banks et al. (2012). The use of multi-frequency SAR to classify AGB, CC and TCV of savannah woody structures was studied in Naidoo et al. (2015), and found that although the combination of L-, C, and X-band give the highest classification accuracy, the use of L-band alone is only marginally lower than using all three frequencies. The use of random forests and SVMs to classify land cover using multi-frequency SAR, including derived interchannel ratios, texture measures and elevation data of the wet and dry seasons of both forested and semi-arid regions of Africa has not been attempted before.

Most land cover studies focus on areas in Europe or North America where reliable and regularly updated land cover maps are available (Wegmuller et al., 2004; Kouskoulas et al., 2004; Waske and Benediktsson, 2007). As a result there are many areas in Africa, which have not been the focus of research studies, and where it would be useful to focus

future research efforts. Moreover, there are many humanitarian crises in central Africa, and this research has the potential to benefit the aid organisations who supply aid and support to these areas of conflict, by having better knowledge of what is happening on the ground. The sites that are chosen in this study are based on data availability, due to the sparse overlapping of high resolution images from multiple SAR satellites during the last five years. The forested sites that are chosen, are located in Cameroon and the DRC. The semi-arid sites are based in Tanzania and Chad.

The three hypotheses are investigated in Chapter Five, Chapter Six and Chapter Seven, but first the study sites, data and methodology are given in Chapter Three, with an expansion on data preparation in Chapter Four.

## Chapter Three Study sites, data and general methodology

This chapter discusses the broader context of the study sites, the process of choosing the study sites and data, and the general methodology for the three experiments. The characteristics of the chosen SAR images are discussed and tabulated.

### 3.1 Study sites

The study sites chosen from the available data are in Cameroon, DRC, Tanzania and Chad. The Cameroon and DRC sites are forested areas whereas the Tanzania and Chad sites are semi-arid. For the forested areas, the Cameroon site was selected as the model development site, while the DRC site was an additional site to test the transferability of the model. For the semi-arid sites, the Tanzania site was the model development site, while the Chad site was used to test transferability of the model. The geographic locations of each site within the biome map of Africa are shown in Figure 3.1. The Cameroon site is within the *mosaics of forest* biome and the DRC site is within the *tropical lowland forest* biome. Both *mosaics of forest* and *tropical lowland forest* are types of forests, and the Cameroon and DRC sites were therefore grouped together within a wider group of forested ecoregions. The Tanzania study site is within the *moist-infertile savanna* and the Chad site within *arid-fertile savanna*, which are both types of semi-arid areas. The Tanzania and Chad sites were grouped into a wider group of semi-arid ecoregions.

The location of the study sites, especially those in Chad and the DRC, were currently difficult to visit due to internal civil and political conflict. For this reason, it was decided not to perform fieldwork, but to rather use available very high resolution optical imagery on Google Earth and moderate resolution images from Landsat to identify land cover classes. This is not the ideal situation, but fieldwork would also not necessarily have corresponded with the satellite acquisition dates to provide an accurate account of the land cover in the SAR images. Several other studies have used very high resolution

images available on Google Earth as ground reference data, due to its easy and free access (Brink and Eva, 2011; Evans et al., 2010; Turkar et al., 2012).

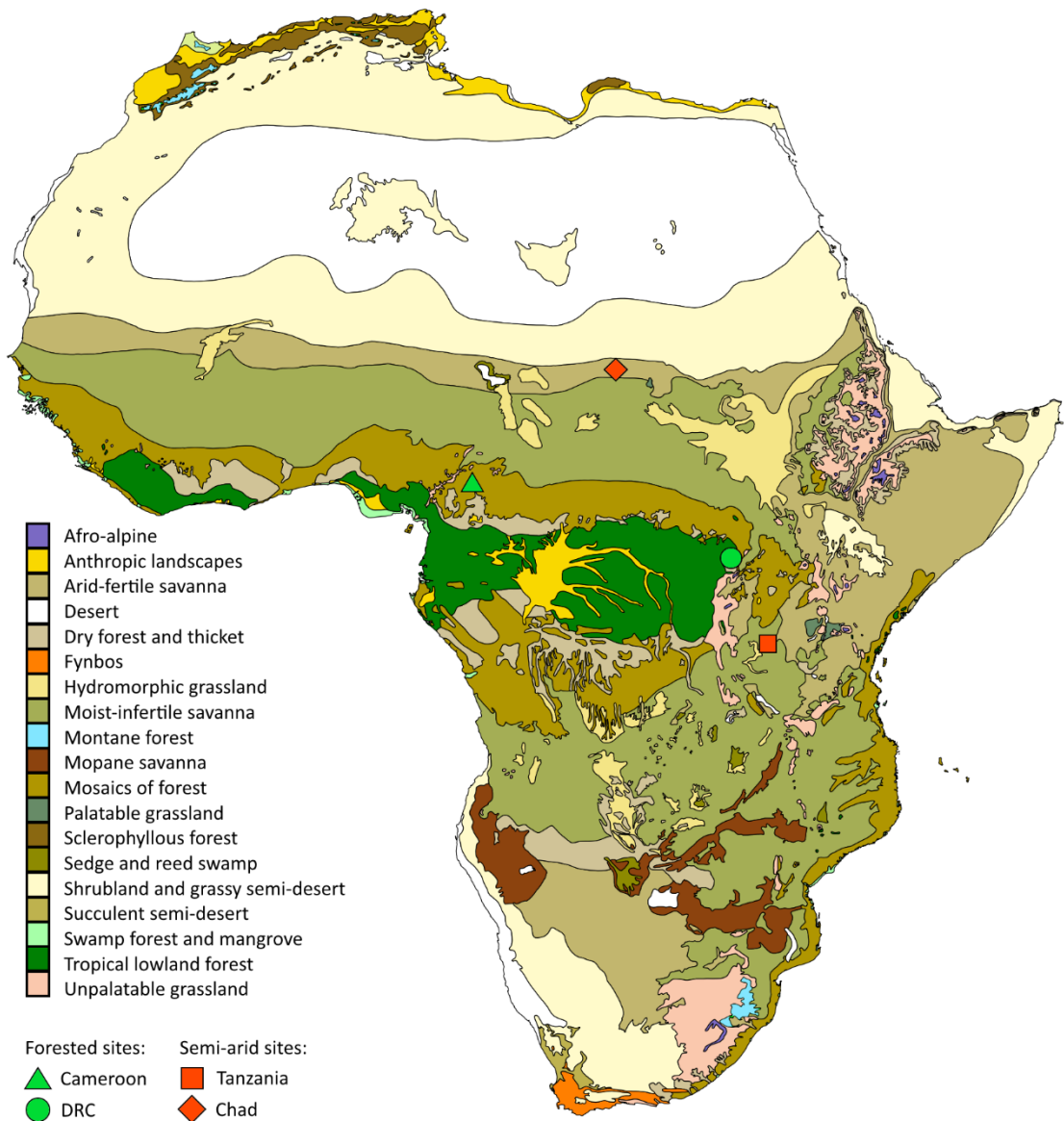


Figure 3.1 The locations of the study sites, in the context of the biome map of Africa (White, 1983b).

Before choosing study sites for this research, a handful of constraints were set out to restrict which study sites could be chosen:

- Images for each site should overlap geographically.
- Two sites within forested ecoregions and two sites within semi-arid ecoregions.
- Images should be close in time (same season), and if not available, then for the same season from a different year.

- The more available polarisations the better. This is to increase the number of information layers from the available input channels, and to build a comprehensive data stack with the data dimensionality as close to its full potential scenario as possible. This allows for models to be trained with all the possible input variables.
- Only high resolution SAR data were considered ( $\leq 15\text{m}$  azimuth or range resolution), which corresponds to the high resolution imaging modes of the different satellite sensors.
- The sites were chosen to be within Africa. Some preliminary work was done in Africa within the SAFER project (Hello, 2010) and therefore background knowledge and initial datasets are available from selected locations in Africa.

With these constraints in mind, the study sites for this research are chosen from images available in the archives of TerraSAR-X, ALOS PALSAR and ENVISAT ASAR. The data were screened manually, starting with data from the TerraSAR-X archive, since it is the least populated of the SAR archives.

Table 3.1 lists the Universal Transverse Mercator (UTM) projection zones and the European Petroleum Survey Group (EPSG) codes to easily convert between standard latitude / longitude projection and the UTM projection for each site. A site abbreviation along with the upper left and bottom right hand corner coordinates are listed for each site. Each of these sites are now discussed in turn in more detail.

The TerraSAR-X images were requested from the DLR, whereas the ENVISAT ASAR and ALOS PALSAR images were requested via a Category 1 proposal from ESA. ALOS PALSAR is owned by JAXA and is an ESA third party mission.

Table 3.1 List of study site locations with corresponding UTM zones and EPSG codes.

Biome / Ecoregion	Site	Season	Abbreviation	Training samples / test samples	Model development / Additional season / site	Coordinates upper left corner (latlong)	Coordinates bottom right corner (latlong)	UTM zone	EPSG code
Forested	Cameroon	Wet	CW	Training	Forested model development	6°36′33″N 12°31′60″E	6°11′18″N 12°56′27″E	33 North (N)	32633
				Test					
		Dry	CD	Training	Additional season				
				Test					
	DRC	Wet	DW	Training	Additional site	1°38′35″N 29°41′19″E	1°12′12″N 29°52′02″E	35 North (N)	32635
				Test					
Semi-arid	Tanzania	Dry	TD	Training	Semi-arid model development	3°48′34″S 31°57′26″E	4°15′28″S 32°20′47″E	36 South (M)	32736
				Test					
		Wet	TW	Training	Additional season				
				Test					
	Chad	Dry	ChD	Training	Additional site	13°44′46″N 22°02′56″E	13°31′10″N 22°15′23″E	34 North (P)	32634
				Test					
-	All (latitude / longitude)	-	-	-	-	-	-	-	4326
-	All (WGS84 Pseudo Mercator for Google Earth)	-	-	-	-	-	-	-	3857

### 3.1.1 Cameroon

The starting point of the data availability search is the high resolution TerraSAR-X images in Africa. The available data from the ALOS PALSAR and ENVISAT ASAR archives for the overlapping location of the TerraSAR-X archive images, for the site in Cameroon, are shown in Figure 3.2. The rainfall estimation was downloaded from the Famine Early Warning Systems Network ([www.fews.net](http://www.fews.net)), as daily precipitation, and then summed into monthly totals.

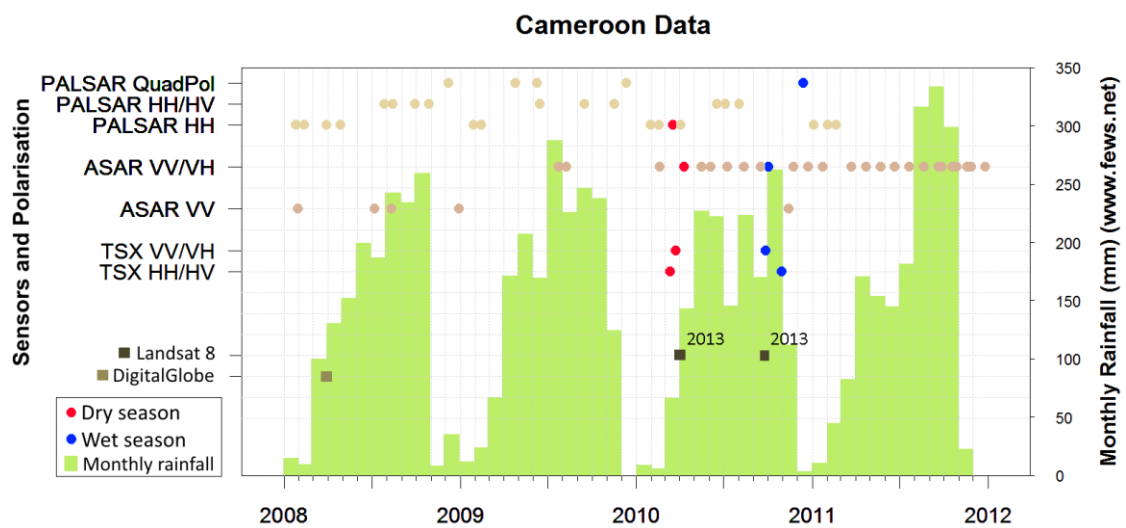


Figure 3.2 Available data from the satellite archives for the Cameroon site, with selected images in red for the dry season and in blue for the rainy wet season. The optical images used in this study is shown in dark brown squares (Landsat 8) and a light brown square, © 2015 Google, Image © 2015 DigitalGlobe.

Figure 3.2 differentiates between the different polarisation and modes (single / dual / quad-polarisation) for images that are available from the archives for each of the three sensors. Each of the sensors and their polarisations are shown on the left-hand axis, with the corresponding available dates on the archive on the x-axis. The red dots highlight the images chosen for the dry season and the blue dots show the images chosen for the wet season from the available images for the Cameroon test site. The optical images used in this study to identify and validate land cover types are the very high resolution image from DigitalGlobe (available on Google Earth) along with two recent images from Landsat 8. To fit on the scale of the graph, the dates of the two Landsat images from 2013 are superimposed on the same days of the year during 2010 in Figure 3.2, which overlap with the available SAR images.

Figure 3.3a shows the footprints of the available images from ALOS PALSAR, ENVISAT ASAR and TerraSAR-X, with the overlap area highlighted in yellow. Figure 3.3b shows the study site as viewed in Google Earth. The wider surrounds are also shown to see the general landscape surrounding the study site.

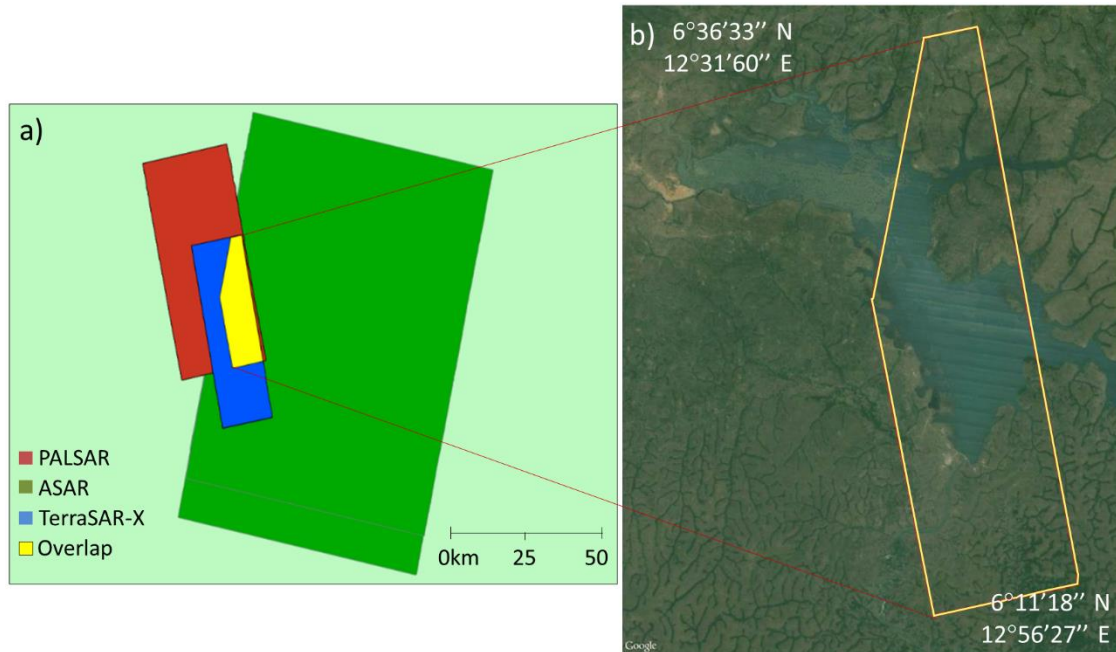


Figure 3.3 a) Overlap between different SAR sensors for the Cameroon study site. b) Google Earth view of the overlapping area and wider surrounds for the Cameroon study site. © 2015 Google, Image © 2015 DigitalGlobe, Image © 2015 CNES /Astrium, © 2015 CNES /Spot Image.

The SRTM elevation along with dry and wet season optical images and dry and wet season composites of SAR images are shown in Figure 3.4. The Landsat 8 images were taken on the 12<sup>th</sup> of April 2013 (dry season) and on the 19<sup>th</sup> of September 2013 (wet season). The multi-frequency SAR image for the dry season is a combination of L-HH, C-HV and X-VH. For the wet season the multi-frequency SAR image is displayed as L-HV, C-VH and X-VH. The large lake in the study site is lake Mbakaou which forms part of the Djerem department (one of five departments), part of the Adamawa region (4<sup>th</sup> largest of ten regions) in Cameroon. The size of the Cameroon study site is 15km by 40km. The elevation of the majority of the study site is between 820m and 940m above sea level.



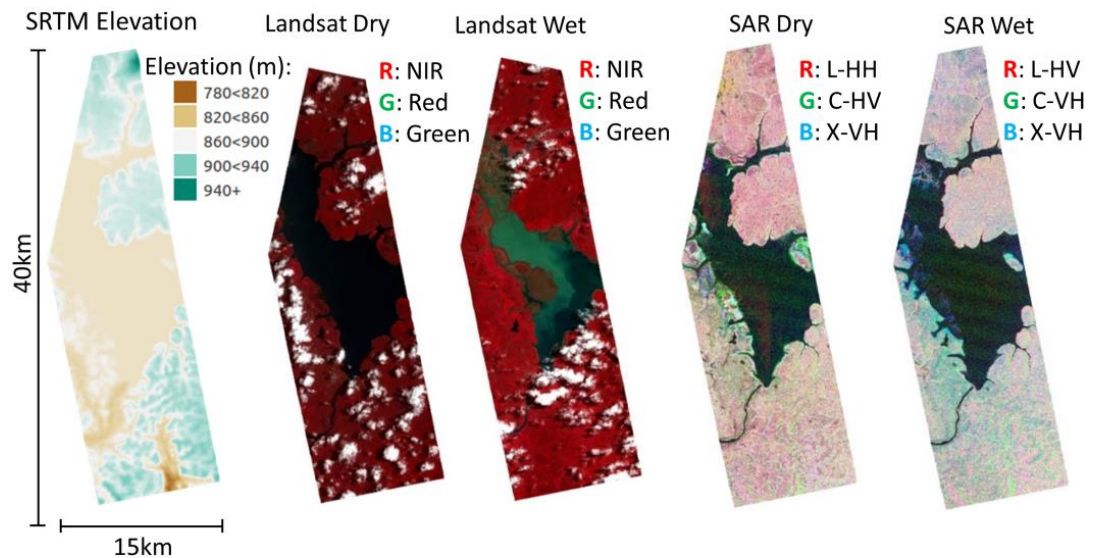


Figure 3.4 The SRTM elevation, Landsat (dry and wet) and multi-frequency SAR (dry and wet) images of the Cameroon site.

Most of this region is being converted to agricultural land from the natural habitat of submontane forests. The soil are from volcanic rock, which is productive soils for farmland, but also susceptible to dry out easily during the dry season, becoming non-vegetated areas. Vegetation structure are therefore a mixture of remaining forest areas are open areas which consist of a mixture of grass and other vegetation. Species from this forested montane zone include *Nuxia congesta*, *Podocarpus latifolius*, *Prunus africana*, *Rapanea melanophloeos*, and *Syzygium guineense bamendae* (WWF, 2016c).

The Cameroon study site is the main study site used to develop the model for the forested region. The DRC study site, which will be discussed next, will be used to test the transferability of the model to an additional area.

### 3.1.2 DRC

The DRC study was used to determine the transferability of the model developed using the Cameroon data. The available SAR images for this site are shown in Figure 3.5. The sensors and available polarisations are given on the left-hand axis with the available dates of the images in the archives on the x-axis. The monthly rainfall data are also plotted in Figure 3.5 in green and the rainfall pattern differs quite significantly to the

Cameroon site. As a result, only one set of images were chosen, which will be taken as the wet season. With a nearly continuous rainfall throughout the year, this study site does not present a dry season, though there seems to be a six-monthly cyclical rainfall pattern of a wet season and slightly less wet season. This rainfall pattern highlights the benefit of SAR imagery over optical imagery, with the associated consistent cloud cover with the regular rainfall of the tropical regions. The relevant Landsat 8 image for this study site from 2013 is overlain in dark brown onto Figure 3.5 to fit within the scale of the figure and the very high resolution image from DigitalGlobe, as available on Google Earth is shown in light brown. The DigitalGlobe image is used to identify the land cover samples to test the transferability of the model for the forested areas.

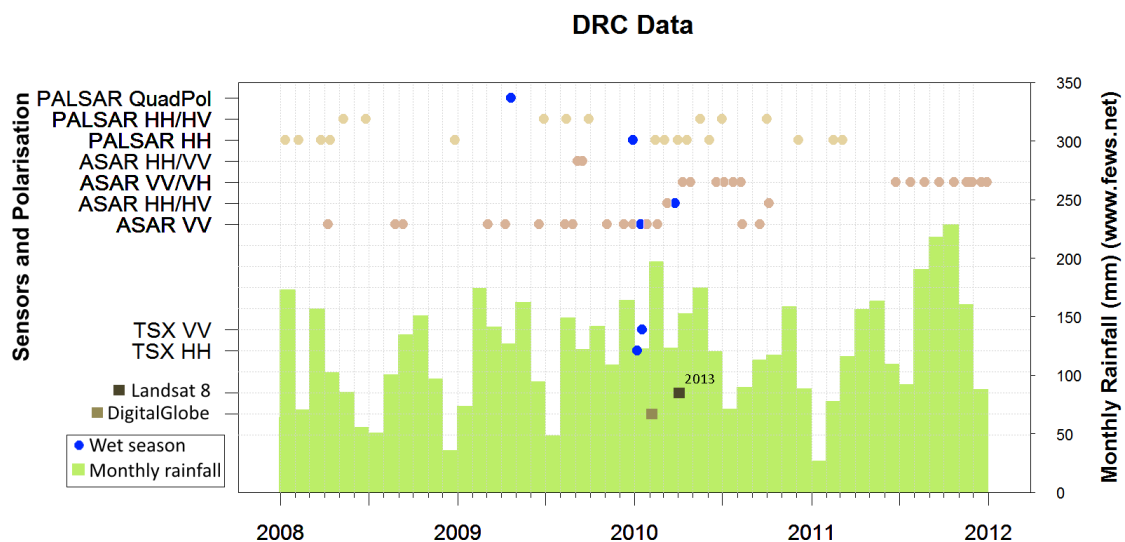


Figure 3.5 Available data from the archives for the DRC site, with selected images in dark blue for the wet season.  
© 2015 Google, Image © 2015 DigitalGlobe.

The overlapping area between the TerraSAR-X, ASAR and PALSAR images is shown in Figure 3.6a. Similar to the other sites, the area of overlap between the three SAR sensors is relatively small, compared to footprint sizes of the individual images. The overlap area are 20km by 50km at its widest dimensions. The Google Earth image for the study site and surrounding area is shown in Figure 3.6b. Areas of dense forest and deforested areas are clearly visible in this image.

Figure 3.7 shows more detail regarding the DRC study site. The SRTM elevation data are shown with the DRC study site being in the order of 1000m above sea level. This is at a similar elevation to the Cameroon study site, which is at around 900m above sea level. The Landsat 8 image is shown in false colour with red as the short wave infrared (SWIR) band, green as the near infrared (NIR) band and blue as the red band. This highlights the forested areas in black, the settlement area in white and the river in orange. A multi-frequency SAR image is also shown, with the L-band HV channel as red, C-band HH as green and X-band VV as blue. This shows the forest /non-forest clearly with the forested areas as red and the non-forested areas ranging between blue and green.

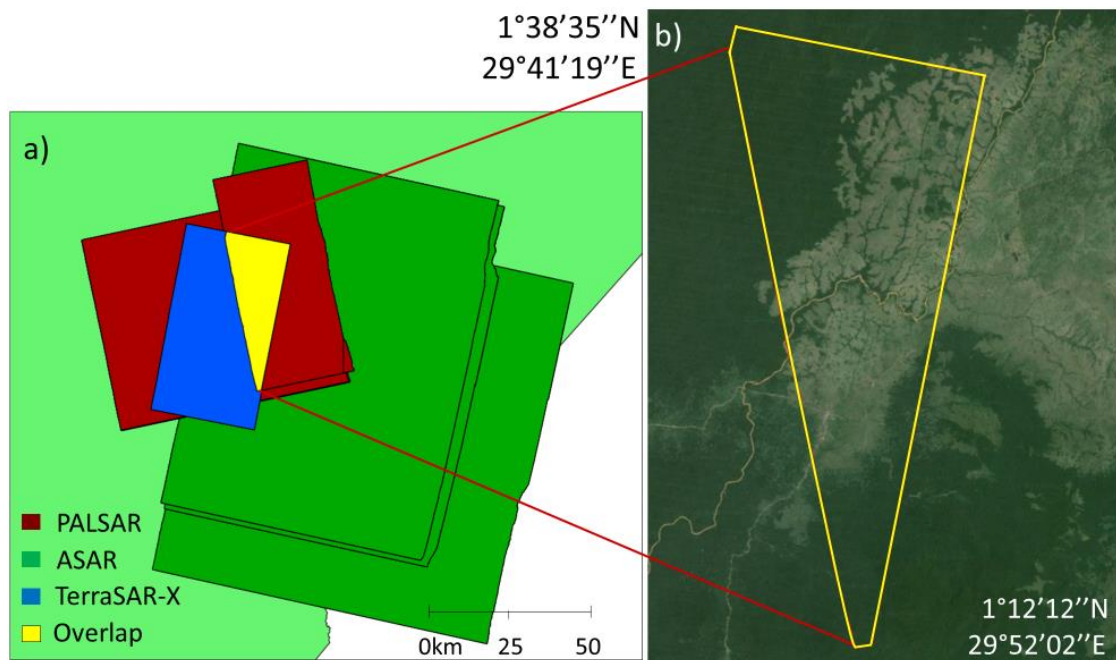


Figure 3.6 a) Overlap area from the different SAR sensors for the DRC site. b) Google Earth view of the overlapping area and wider surroundings for the DRC study site. © 2015 Google, Image © 2015 DigitalGlobe, Image Landsat.

This eastern part of the DRC is part of an undifferentiated Afromontane forest, which is less tall than the Afromontane forest. Tree species types include a mix of *Apodytes dimidiata*, *Halleria lucida*, *Ilex mitis*, *Kiggelaria africana*, *Nuxia congesta*, *Nuxia floribunda*, *Ocotea bullata* (including *Ocotea kenyensis*), *Podocarpus falcatus* (including *gracilior*), *Podocarpus latifolius*, *Prunus africana*, *Rapanea melanophloeos* and *Xymalos monospora* (White, 1983a). Many of the forested areas are converted

into agricultural areas. Other deforestation reasons are logging and firewood collection (WWF, 2016a).

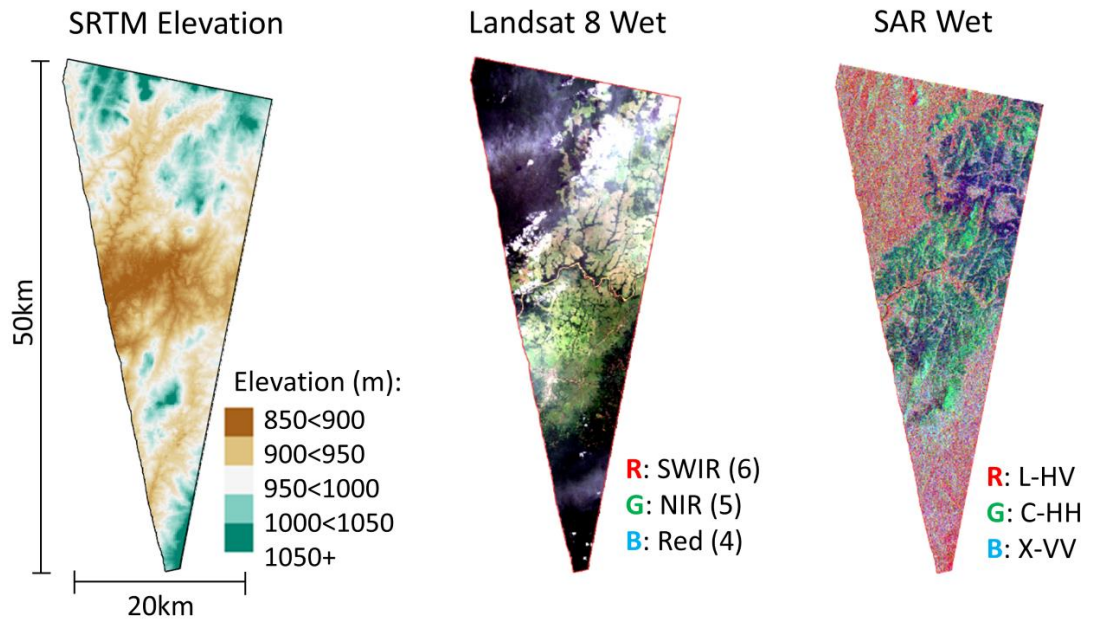


Figure 3.7 SRTM elevation, Landsat 8 and multi-frequency SAR images of the DRC study site.

### 3.1.3 Tanzania

The Tanzania study site is the main study site for the semi-arid region, used for model development. The dates of the available SAR images for the study site in Tanzania are shown in Figure 3.8. The selected images for the wet season are shown in dark blue and for the dry season in red. The rainfall pattern is distinctly different from the forested sites, with less rain and a longer and distinct dry season. The Landsat 8 images were available for both the wet and dry season but from a different year, 2013. One of the DigitalGlobe images, as available on Google Earth, overlaps closely in time with the dry season SAR images, with a second image from 2012 also available for the dry season. The optical images are used for land cover identification and validation, on which the land cover classification model is developed.

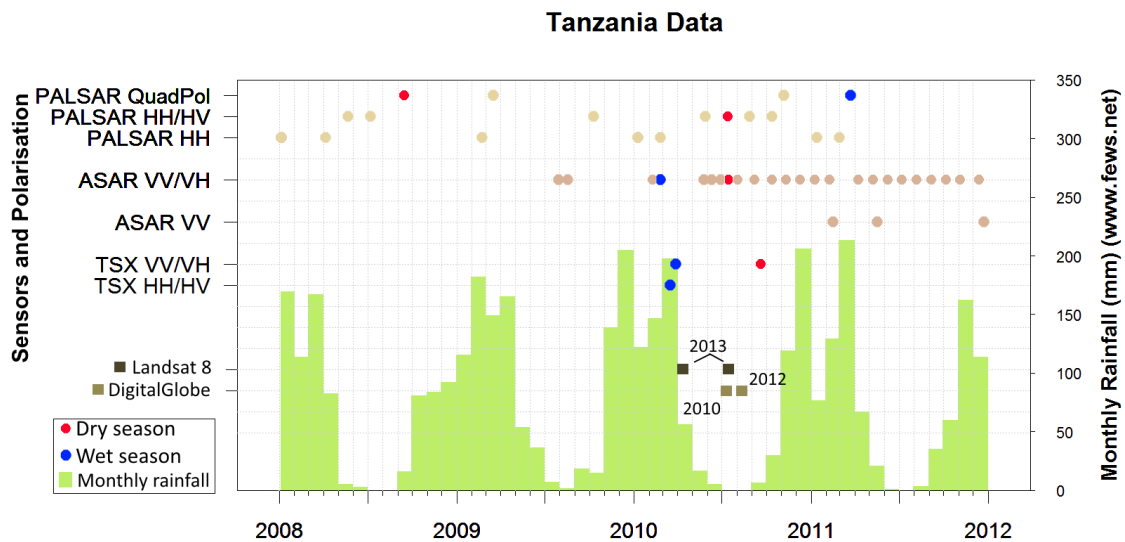


Figure 3.8 Available data from the archives for the Tanzania site, with selected images in red for the dry season and in blue for the rainy wet season. © 2015 Google, Image © 2015 DigitalGlobe.

The area of overlap for all the SAR images for the Tanzania site is shown in Figure 3.9a. It is approximately 10km x 45km and falls within the Kigosi Game Reserve, which is on the United Nations list of protected areas (IUCN, 2015). The Kigosi Game Reserve is an International Union for Conservation of Nature (IUCN) Category IV protected area, which means it is protected for its habitat and species management. Although the whole site falls, in theory, within the protected area, it is assumed that the actual boundary of the game reserve falls on the clear contrast of forest / non-forest as can be seen in the western side of Figure 3.9b. This is broadly within the Shinyanga region of Tanzania. The species of the woodlands of Shinyanga are *Brachystegia*, *Julbernardia* and *Isoberlinia*. The natural vegetation of Shinyanga is shrubs of 4-6m high, and thorny deciduous trees of 10-15m high. Agricultural food crops are predominately maize, cassava, sorghum, beans, rice, chickpeas, groundnuts and sweet potatoes. A decline in soil fertility and soil erosion are concerns in the area (Kamwenda, 2002).

The SRTM elevation for the Tanzania study site, along with the Landsat 8 and multi-frequency SAR images for both the wet and dry season are shown in Figure 3.10. This site is located higher than the forested sites, at between 1100m and 1300m above sea level.



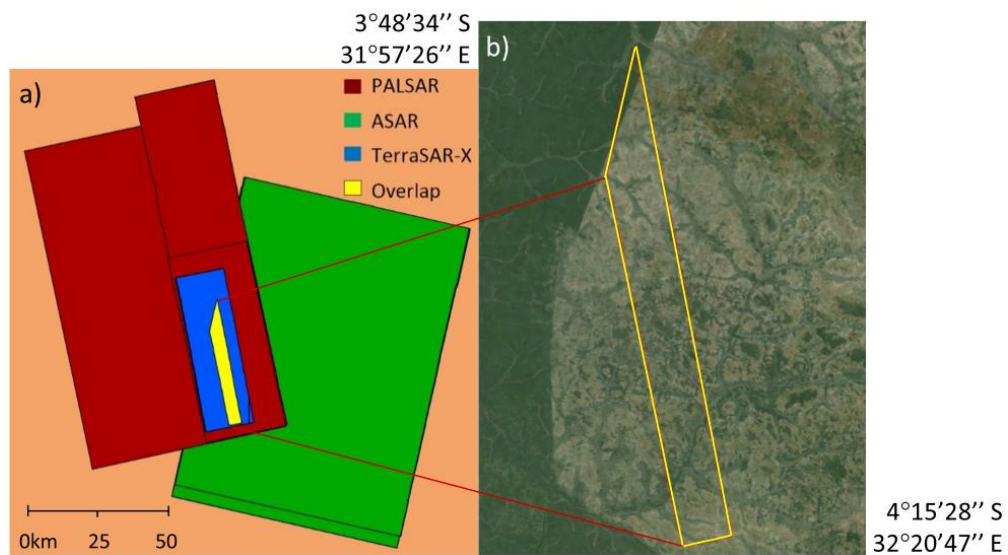


Figure 3.9 a) Overlap area from the different SAR sensors for the Tanzania study site. b) Google Earth view of the overlapping area and wider surroundings for the Tanzania semi-arid study site. © 2015 Google, Image ©2015 DigitalGlobe, ©2015 CNES/Spot Image, Image © 2015 CNES / Astrium.

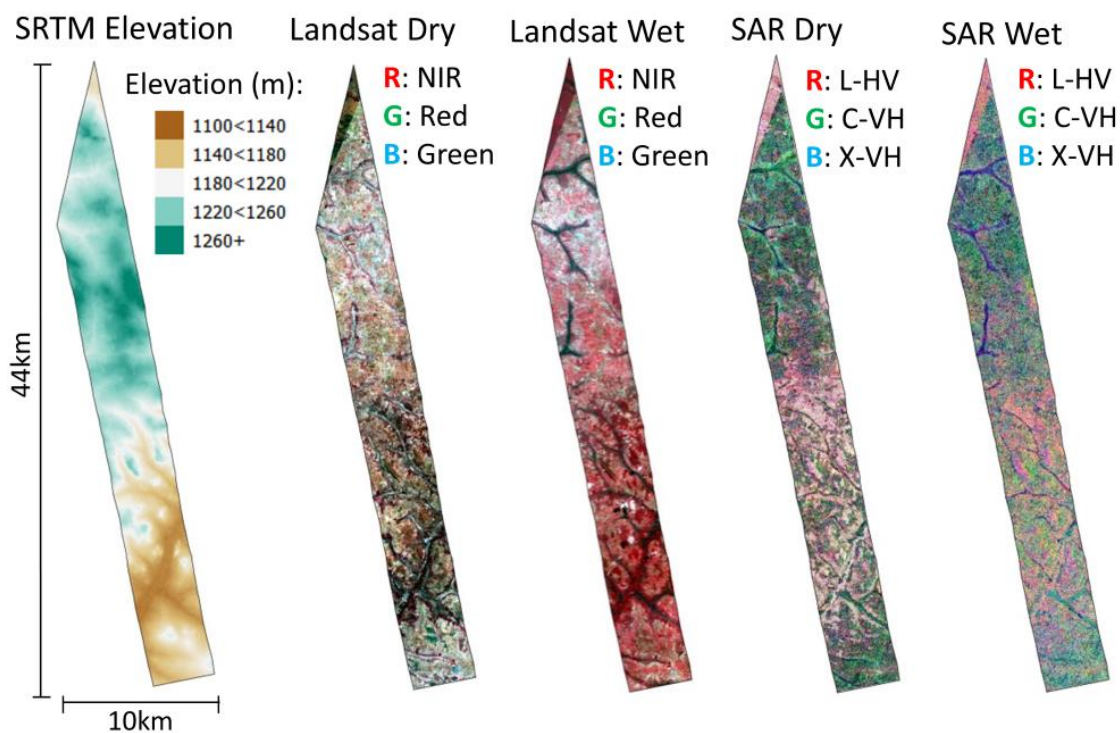


Figure 3.10 SRTM elevation, Landsat (dry and wet season images) and multi-frequency SAR (dry and wet season images) for the Tanzania site.

### 3.1.4 Chad

The Chad study site is the site that is used to test the transferability of the model for the semi-arid areas, as developed on the Tanzania study site. Figure 3.11 shows the available data for the Chad study site with the selected images for the dry season shown in red. Only images for the dry season are chosen, as SAR images from L-, C- and X-band were available for the dry season compared to the wet season. The rainfall pattern for the Chad study site is also shown in Figure 3.11, and the site is seen to be drier than the Tanzania study site with longer periods of no rain. The very high resolution DigitalGlobe image available is only from 2004, which overlap with the same season as the SAR images from 2008 to 2009. A Landsat image from 2013 is also available. The DigitalGlobe and Landsat 8 images are used to identify and validate land cover sample areas. This is the best optical imagery available, and assumptions are made regarding the stability of land cover types for the study area for the years between the optical and SAR images.

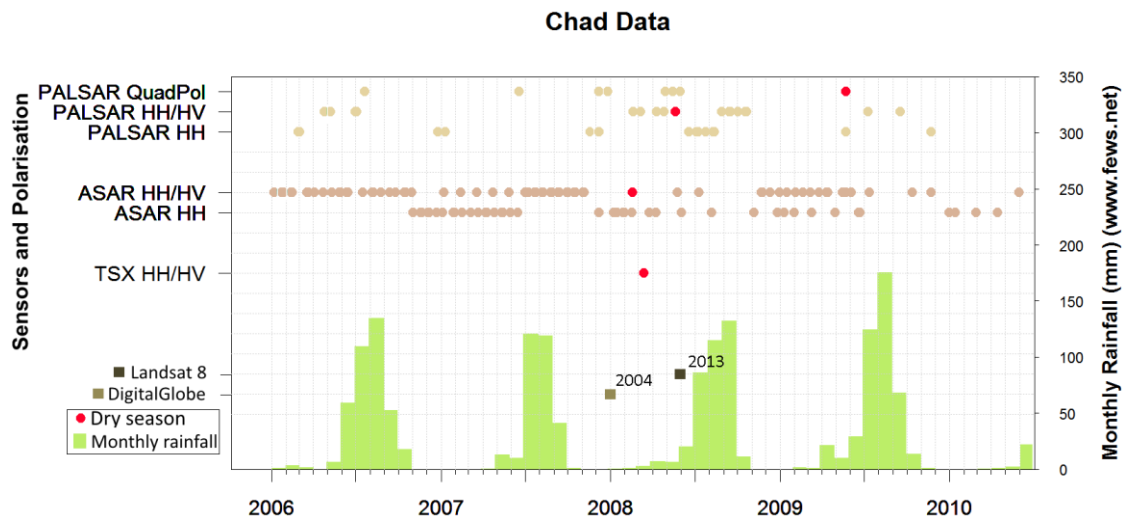


Figure 3.11 Available data from the archives for the Chad site, with selected images in red for the dry images. © 2015 Google, Image © 2015 DigitalGlobe.

The overlap area between the ALOS PALSAR, ENVISAT ASAR and TerraSAR-X images is shown in Figure 3.12a. The view of the images on Google Earth is shown in Figure 3.12b, with a dry river bed visible through the centre of the study site. This river is a border between Chad on the western side and Sudan on the eastern side. The site is 19km x 17km in size, as shown in Figure 3.13. The topography for the Chad study site is relatively flat as shown in the SRTM elevation. Additionally, the Landsat scene from 27 May 2013 is shown along with a multi-frequency SAR image of the Chad study site.

This site falls within the central Sahelian region, which is the semi-arid transition zone between the woody Sudanian savanna in the south and the desert in the north. It consists mostly of semi-arid grasslands and thorny shrublands. Grass species are *Cenchrus biflorus*, *Schoenefeldia gracilis*, and *Aristida stipoides*. The dominant tree species are acacia trees: *Acacia tortilis*, *Acacia senegal* and *Acacia laeta*. During the long dry season, most trees lose their leaves and most of the annual grasses die, resulting in a dry landscape (WWF, 2016b).

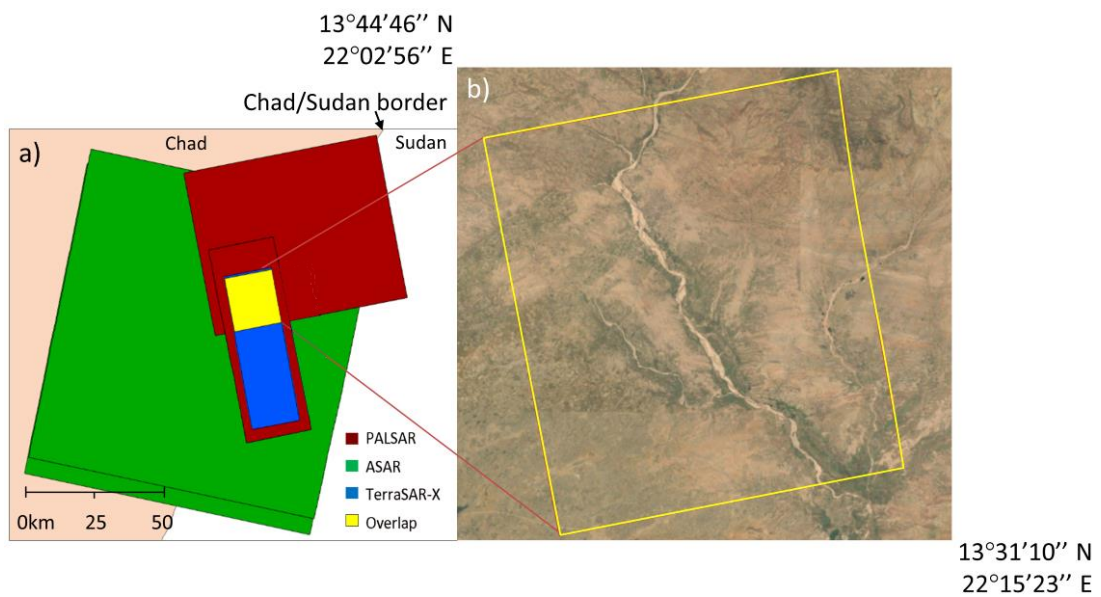


Figure 3.12 a) Overlap area from the different SAR sensors for the Chad site. b) Google Earth view of the overlapping area and wider surroundings for the Chad semi-arid study site. © 2015 Google, © 2015 CNES/Spot Image, Image © 2015 DigitalGlobe, Image © 2015 CNES / Astrium.

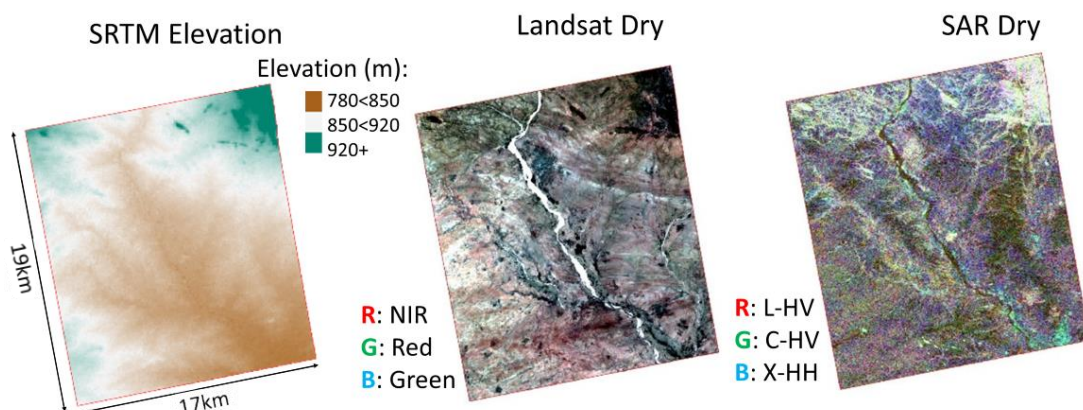


Figure 3.13 SRTM elevation, Landsat and multi-frequency SAR for the Chad study site.

A list of all the optical images from DigitalGlobe and Landsat 8, along with the dates and seasons for the images, is given in Table 3.2.



Table 3.2 The available optical images for the four study sites, with respective dates and seasons for each site.

Site	Sensor	Date	Season
Cameroon	DigitalGlobe (Google Earth)	2008-04-13	Dry
	Landsat 8	2013-04-12	Dry
		2013-09-19	Wet
Chad	DigitalGlobe (Google Earth)	2004-12-31	Dry
	Landsat 8	2013-05-27	Dry
		2013-08-31	Wet
DRC	DigitalGlobe (Google Earth)	2010-02-04	Wet
		2011-04-19	Wet
		2012-02-06	Wet
	Landsat 8	2013-04-24	Wet
		2013-07-13	Wet
Tanzania	DigitalGlobe (Google Earth)	2010-06-09	Dry
		2012-08-02	Dry
	Landsat 8	2013-06-15	Dry
		2013-04-26	Wet
Chad	DigitalGlobe (Google Earth)	2004-12-31	Dry
	Landsat 8	2013-05-27	Dry
		2013-08-31	Wet

### 3.2 Characteristics of available images for each site

A summary of the available data for the four study sites, namely Cameroon and DRC forested sites and Tanzania and Chad semi-arid sites, is given in Tables 3.3 and 3.4. The inter-seasonal comparison of the images is shown in Table 3.3 with all sensor-polarisation combinations of the same incidence angle, apart from the L-band HH images for the Cameroon study site. The L-band HH images are of 25°-27° and 41° mean incidence angles. This shows the combination of images available for comparison between the wet and dry seasons for the Cameroon and Tanzania sites. Table 3.4 shows the frequency and polarisation of images available for comparison between the main development site and the additional site for both the forested and semi-arid areas. The combination of these frequency-polarisation images can be used to test the transferability of the land cover classification model to the additional sites. The TerraSAR-X HV and VH images for the Tanzania wet season site had unsatisfactory orthorectification results, as is discussed in Section 4.1, which as a result is not used in the model development and further analysis.

Table 3.3 The available images for each study site, with colours showing images to be used for an inter-seasonal comparison between the semi-arid and forested study sites.

Sensor	Polarization	Mean Incidence Angle		Forested			Semi-Arid		
				Cameroon		DRC	Tanzania		Chad
				Dry	Wet	Wet	Wet	Dry	Dry
PALSAR	HH	25°-27°			✓	✓		✓	✓
		41°	✓			✓		✓	✓
	HV	25°-27°			✓	✓	✓	✓	✓
		41°						✓	✓
	VH	25°-27°			✓	✓	✓	✓	✓
	VV	25°-27°			✓	✓		✓	✓
ASAR	HH	26°-31°				✓			✓
	HV	26°-31°				✓			✓
	VH	36°	✓	✓			✓	✓	
	VV	26°-31°				✓			
		36°	✓	✓			✓	✓	
TSX	HH	28°-30°	✓	✓					✓
		44°-45°			✓				
	HV	28°-30°	✓	✓			x		✓
	VH	28°-30°	✓	✓			x	✓	
	VV	28°-30°	✓	✓					
		44°-45°			✓				

	Forested	Semi-Arid
Same Incidence Angle		
Different Incidence Angle		N/A

✓- available and used in further analysis

x- available, but did not orthorectify correctly. Therefore not used in further analysis.

Table 3.4 The available images for each study site, with colours showing the available images used for testing the transferability of the classification model to an additional study site for both the forested and semi-arid areas.

Sensor	Polarization	Mean Incidence Angle	Forested			Semi-Arid		
			Cameroon		DRC	Tanzania		Chad
			Dry	Wet	Wet	Wet	Dry	Dry
PALSAR	HH	25°-27°		✓	✓		✓	✓
		41°	✓		✓		✓	✓
	HV	25°-27°		✓	✓	✓	✓	✓
		41°					✓	✓
	VH	25°-27°		✓	✓	✓	✓	✓
	VV	25°-27°		✓	✓		✓	✓
ASAR	HH	26°-31°			✓			✓
	HV	26°-31°			✓			✓
	VH	36°	✓	✓		✓	✓	
	VV	26°-31°			✓			
		36°	✓	✓		✓	✓	
TSX	HH	28°-30°	✓	✓				✓
		44°-45°			✓			
	HV	28°-30°	✓	✓		✗		✓
	VH	28°-30°	✓	✓		✗	✓	
	VV	28°-30°	✓	✓				
		44°-45°			✓			

	Forested	Semi-Arid
Same Season, Same Incidence Angle		
Same Season, Different Incidence Angle		N/A
Different Season, Same incidence Angle		N/A

✓ - available and used in further analysis

✗ - available, but did not orthorectify correctly. Therefore not used in further analysis.

All images were ordered as single-look complex (SLC) images, to allow for an automated orthorectification of the images. However, for a select number of TerraSAR-X images, only detected images were granted due to the sensitivity of the geographic locations. This is for the sites in the DRC and Chad, due to the civil conflict present in these areas during this timeframe. The TerraSAR-X images for these sites were therefore manually corrected by selecting ground-control points, to match the ENVISAT ASAR and ALOS PALSAR images.

### 3.3 General methodology for the three experiments

The general methodology for the three experiments is underpinned by first getting the data in a useable format. This is covered in Chapter Four, and consists of the orthorectification of the images, creating an image overlap, calculating the derived layers, combining the images into a layer stack and finally to selecting samples of land cover classes as identified from available optical images. The derived layers that are calculated are interchannel ratios and texture measures from the backscatter coefficient data and elevation, slope and aspect layers from the SRTM data for the same sites. As part of the sample selection, the land cover polygons are randomly split between training and test sample areas. Samples from the test areas are then kept separate on which to evaluate the models.

To assess whether multi-frequency SAR backscatter coefficient images, along with interchannel ratios and texture measures, can be used to successfully distinguish between land cover classes in the semi-arid and forested areas of Africa, random forest models are built for multiple scenarios. Once the data are in the correct format, the dataset is grouped into combinations of L-, C-, and X-band for the backscatter coefficients, interchannel ratios, texture measures and elevation data. Random forest models are run for each of these scenarios and the overall accuracies as well as class-specific accuracies compared in Chapter Five. To assess which channels and / or derived variables separate the land cover classes the most, for both the wet and dry seasons, the variable importance measures from the random forest model are used. As part of the random forest model output, two measures of variable importance, namely mean decrease accuracy (MDA) and mean decrease Gini (MDG) are given and the classification accuracies of both measures are compared. The random forest models using the top 30 variables of both MDA and MDG variable importance measures are compared for each of the sites and seasons. The same top 30 variables for each site/season are then used to train an SVM model and the overall classification accuracies are compared with those from the random forest models. The final chosen model is run on the whole image for the Cameroon wet season site, to see the results not only for the samples, but applied to the whole area where the SAR images overlap.

In Chapter Six, the random forest models developed in Chapter Five are applied to the different season of the same site, for both the forested and semi-arid sites, to assess the transferability of these models to a different season. Two types of comparisons were made, namely inter-season and inter-site comparisons, for the forested and semi-arid ecoregions (Figure 3.14). For the forested sites, the model developed on the Cameroon wet season samples are applied to the Cameroon dry season samples, and model developed on the Cameroon wet season samples is applied on the DRC wet season samples. For the semi-arid sites, the model developed on the Tanzania dry season samples is applied to the Tanzania wet season samples, and model developed on the Tanzania dry season samples is applied on the Chad dry season samples.

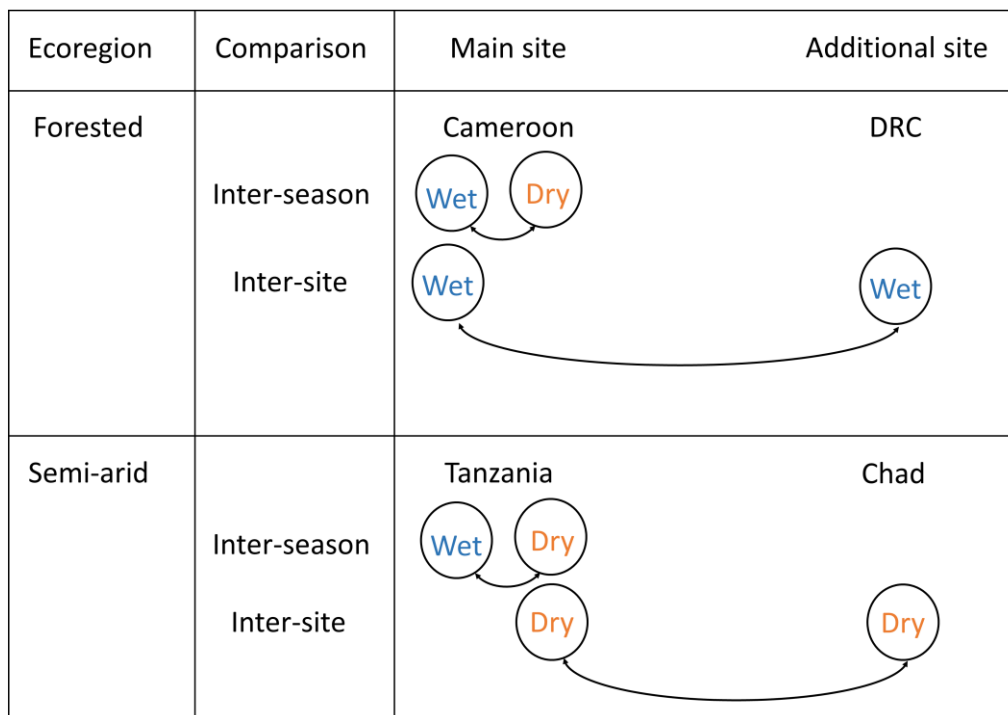


Figure 3.14 The inter-season and inter-site comparison between the main and additional sites for the forested and semi-arid ecoregions.

Since there are fewer images available for the additional site than for the main development site for both the forested and semi-arid ecoregions, the models had to be re-trained using only the available images and calculated feature layers from the additional sites (see Table 3.3 for available frequencies and polarisations for the inter-season comparison and in Table 3.4 for the inter-site comparison).

In terms of the land cover class comparison, two approaches are followed here. Firstly to apply the model from the main development site directly to the additional season and additional sites using the exactly the same land cover classes. These are *bare soil*, *dense trees*, *sparse vegetation*, *settlement* and *water* for the forested sites; and *bare soil*, *dense trees*, *sparse vegetation*, *settlement* and *agriculture* for the semi-arid sites. The *open woodland* class was left out in these models, since it was not present in the additional sites for both the forested and semi-arid ecoregions. This direct application of the model would assess the transferability of the model directly for both the inter-season and additional site comparison for the forested and semi-arid ecoregions.

Secondly, a model is built, but keeping the classes as those that are site-specific. This implies that for the forested site the model is trained on the training datasets of both Cameroon wet and Cameroon dry season and applied on the test datasets of both the Cameroon wet and Cameroon dry season for the inter-season comparison. For the additional site comparison for the forested ecoregion, the model is trained on training datasets of both the Cameroon wet and DRC wet season sites; and applied on the test datasets of both the Cameroon wet and DRC wet season sites. The resulting land cover classes are *CD-bare soil*, *CD-dense trees*, *CD-sparse vegetation*, *CD-settlement*, *CD-water*, *CW-bare soil*, *CW-dense trees*, *CW-sparse vegetation*, *CW-settlement* and *CW-water* for the forested inter-season comparison; and *CW-bare soil*, *CW-dense trees*, *CW-sparse vegetation*, *CW-settlement*, *CW-water*, *DW-bare soil*, *DW-dense trees*, *DW-sparse vegetation*, *DW-settlement* and *DW-water* for the forested additional site comparison.

Similarly for the semi-arid ecoregion, the model is trained on both the Tanzania dry season and Tanzania wet season training datasets and applied on the test datasets of both the Tanzania dry season and Tanzania wet season for the inter-season comparison. For the additional site comparison, the model is trained on training datasets of both the Tanzania dry and Chad dry season; and applied on the test datasets of both the Tanzania dry and Chad dry season. The resulting land cover classes are the *TD-bare soil*, *TD-dense trees*, *TD-sparse vegetation*, *TD-settlement*, *TD-agriculture*, *TW-bare soil*, *TW-dense*

*trees*, *TW-sparse vegetation*, *TW-settlement* and *TW-agriculture* for the semi-arid inter-season comparison; and *TW-bare soil*, *TW-dense trees*, *TW-sparse vegetation*, *TW-settlement*, *TW-agriculture*, *ChD-bare soil*, *ChD-dense trees*, *ChD-sparse vegetation*, *ChD-settlement* and *ChD-agriculture* for the semi-arid additional site comparison. Again, the *open woodland* class was left out in the site-specific land cover class models, since it was not present in the additional sites for both the forested and semi-arid ecoregions. The reason for the site-specific land cover classes, is to assess if the random forest model can distinguish the site-specific land cover classes as separable from each other. This can then be used to motivate a refinement of more land cover classes that the model can classify.

To test whether the statistical distributions of the land cover classes of the main model site and the additional site are similar enough for the model to be transferable, the distributions are compared both visually and with a statistical measure in Chapter Seven. The statistical measure used to compare the distributions of the land cover classes is the Kullback-Leibler divergence (KLD). The visual comparisons are made for the backscatter coefficients, interchannel ratios, texture measures and elevation data. This was undertaken for the Cameroon wet season training vs test samples as a baseline measure. The visual comparison was also undertaken on the Cameroon wet vs dry season samples and the Cameroon wet season vs the DRC wet season samples. The KLD measures were computed and assessed for the backscatter coefficients and the top eight texture measures for the Cameroon wet season training vs test samples, Cameroon wet season vs dry season samples and the Cameroon wet vs DRC wet season samples. This shows how different the land cover SAR signal distributions are between the training and test samples, wet and dry seasons, and between the main development study site and the additional study site for the forested site in Cameroon. A similar comparison was performed for the semi-arid site in Tanzania, comparing the Tanzania dry season training vs test samples, Tanzania dry vs wet season samples and the Tanzania dry season vs Chad dry season samples.



To round off Chapter Seven, KLD class difference measures are calculated from the KLD measures, and related to the user's accuracy of the relevant random forest models for both the forested and semi-arid sites.

### 3.4 Sample selection of land cover types

The land cover types for each site were identified by viewing very high resolution images from DigitalGlobe available on Google Earth. Apart from the DigitalGlobe images, the land cover types were also identified using available Landsat 8 images, which were processed to top-of-atmosphere reflectance for each site. The land cover classes identified for the Cameroon and Tanzania study sites were *bare soil*, *dense trees*, *open woodland*, *sparse vegetation*, *settlements*, *water* (Cameroon only) and *agriculture* (Tanzania only). An explanation of the land cover types with regards to woodland cover estimation and a description of each land cover type along with the relation of the land cover classes to LCCS (Di Grigorio, 2005) are shown in Table 3.5. The *dense trees* are *closed woody vegetation* in LCCS, the *open woodland* are *open woody vegetation* in LCCS. The *sparse vegetation* class is a combination of *sparse woody vegetation* and *herbaceous closed to open vegetation* in LCCS. *Settlements* are built-up areas, and the water land cover class is a combination of both the *natural waterbodies* and *artificial waterbodies* LCCS land cover classes. The *agriculture* land cover class is matched with the *herbaceous crops* LCCS land cover class.

One of the challenges with regards to land cover classification across these African landscapes becomes apparent here. This is the lack of clearly defined boundaries between different land cover classes, with a gradual shift from one class to the next, ranging from areas of woodland with close to complete canopy cover (here named *dense trees*), to areas with a combination of shrubs, herbaceous cover and open woody vegetation with 15-65% canopy cover (named *open woodland*), areas with mixed vegetation of herbaceous cover, shrubs and woody canopy cover between 1-15% (named *sparse vegetation*) and areas with little to no vegetation (named *bare soil*). The *settlement*, *water* and *agricultural* areas are more distinct, although the *settlement*

areas often have mixed vegetation, and the *agricultural* areas could vary between bare soil and fully vegetated land, depending on the time of year. To enable a better understanding of the landscape, some examples of the land cover types identified are shown in Figures 3.15 to 3.17. Figure 3.15a shows the location of the land cover sample areas, which are shown more closely in Figure 3.16 for the Cameroon study site. Figure 3.15b shows the location of the land cover sample areas, which is zoomed into in Figure 3.17 for the Tanzania study site.

The land cover classification using Google Earth was reliable to identify and distinguish the different land cover classes from each other. The main uncertainty from using Google Earth, is that the height of the vegetation is unknown, which might affect a finer land classification such as specific tree types. However, Google earth is being used increasingly as an alternative approach to ground truthing in the remote sensing research community (Thenkabail et al., 2009; Tehrany et al., 2013). Google earth provides easy access to very high resolution imagery across the globe. The main challenge observed in this study, is the date and season differences between the imagery used in the research, compared to the acquisition date of the google earth imagery. An assumption is therefore made that the land cover classes remained the same between the two sets of imagery, to make reliable conclusions.

By comparing the selected screenshots from Figure 3.16 and Figure 3.17, the difference between the landscape of the forested Cameroon site and more semi-arid Tanzania site can be seen. All the selected sample areas for the Cameroon and Tanzania sites are shown in Figure 3.18. The sample polygon areas are randomly grouped into training and test areas, with a 50:50 split for training:test data. From these training polygon areas, 3000 pixels per land cover class are randomly selected as training samples. As test samples, another 3000 pixels per land cover class are randomly selected from the test polygon areas, which are different polygons to the training data areas.

Table 3.5 The land cover types identified in the Cameroon and Tanzania study sites, with their woodland cover estimation and additional description.

Land cover name	Woodland cover estimate	Description	LCCS dichotomous phase	LCCS description	LCCS code	LCCS classifiers used
<b>Dense trees</b>	65-100% canopy cover	Dense trees / forest	A12. Natural and semi-natural terrestrial vegetation	Closed woody vegetation	A12A1A10	A-Life form: woody A-Cover: closed > 65%
<b>Open woodland</b>	15-65% canopy cover	Mixed vegetation, trees, grassland, shrubs	A12. Natural and semi-natural terrestrial vegetation	Open woody vegetation	A12A1A11	A-Life form: woody A-Cover: open 65-15%
<b>Sparse vegetation</b>	1-15% canopy cover	Mixed vegetation, trees, grassland, shrubs	A12. Natural and semi-natural terrestrial vegetation	Sparse woody vegetation / Herbaceous closed to open vegetation	A12A1A14 / A12A2A20	A-Life form: woody A-Cover: open 15-1% / A-Life form: herbaceous A-Cover: open 100-15%
<b>Bare soil</b>		Bare soil , with possibly some vegetation	A12. Natural and semi-natural terrestrial vegetation / B16. Bare areas	Herbaceous closed to open vegetation / Bare soil and/or other unconsolidated material(s)	A12A2A20 / B16A5	A-Life form: herbaceous A-Cover: open 100-15% A-Life form: woody A-Cover: sparse 15-1% / A-Surface aspect: bare soil and other unconsolidated materials
<b>Settlements</b>		Settlements	B15. Artificial surfaces and associated areas	Built up area(s)	B15A1	A-Surface aspect: built up
<b>Water (Cameroon)</b>		Water	B27. Artificial waterbodies, snow and ice / B28. Natural waterbodies, snow and ice	Artificial waterbodies / Natural waterbodies	B27A1 / B28A1	A- Physical status: water / A- Physical status: water
<b>Agriculture (Tanzania)</b>		Agricultural fields	A11. Cultivated and managed terrestrial areas	Herbaceous crop(s)	A11A3	A-Life form: Main crop-herbaceous

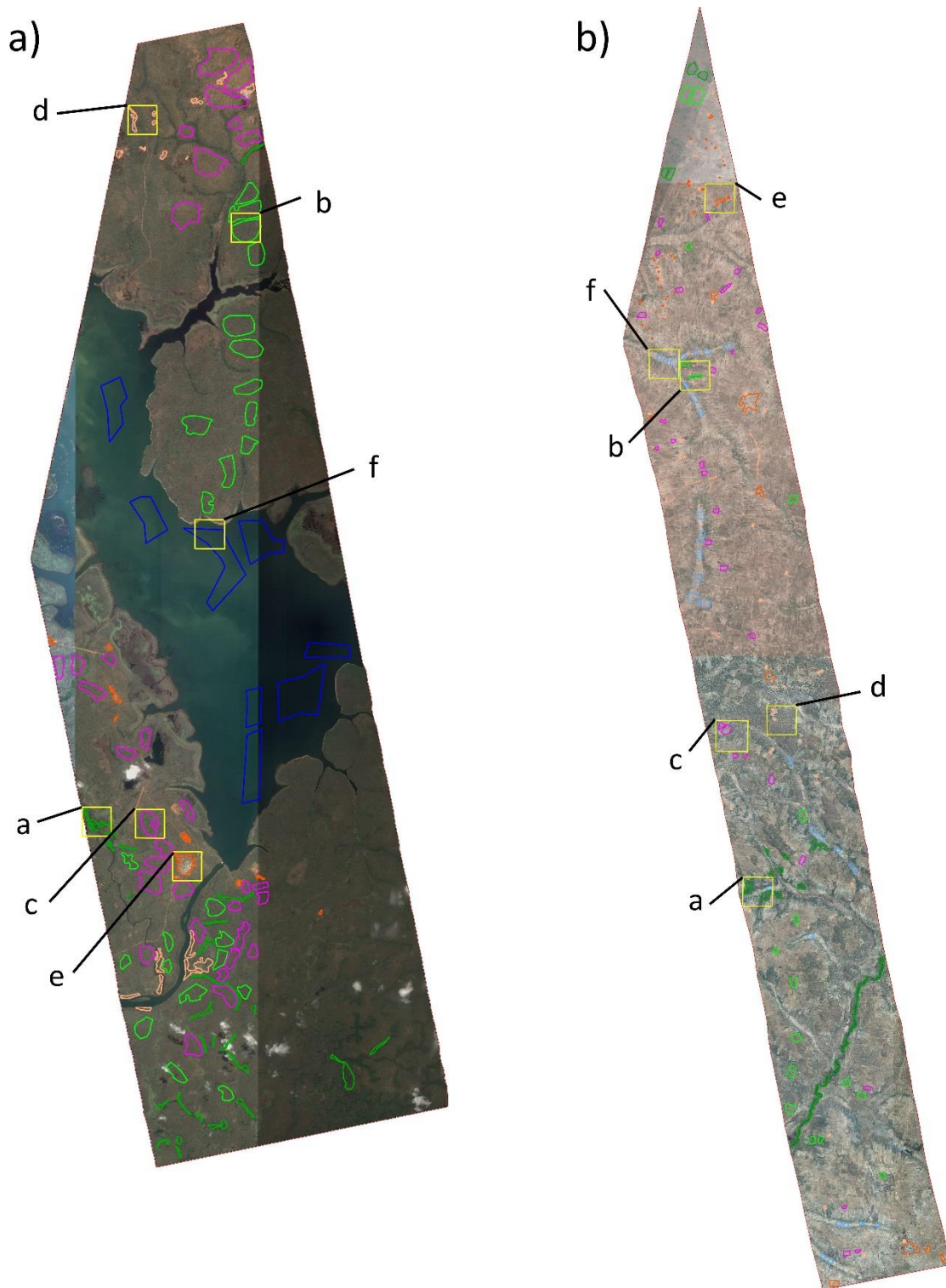


Figure 3.15 a) Cameroon study site with letters a-f referring to the land cover classes show in Figure 3.16. b) Tanzania study site with letters a-f referring to the land cover classes in Figure 3.17. © 2015 Google, Image © 2015 DigitalGlobe.





Figure 3.16 Screenshots from Google Earth of selected sample areas of identified land cover classes for the Cameroon study site. Each image is 1km x 1km in size. © 2015 Google, Image © 2015 DigitalGlobe.



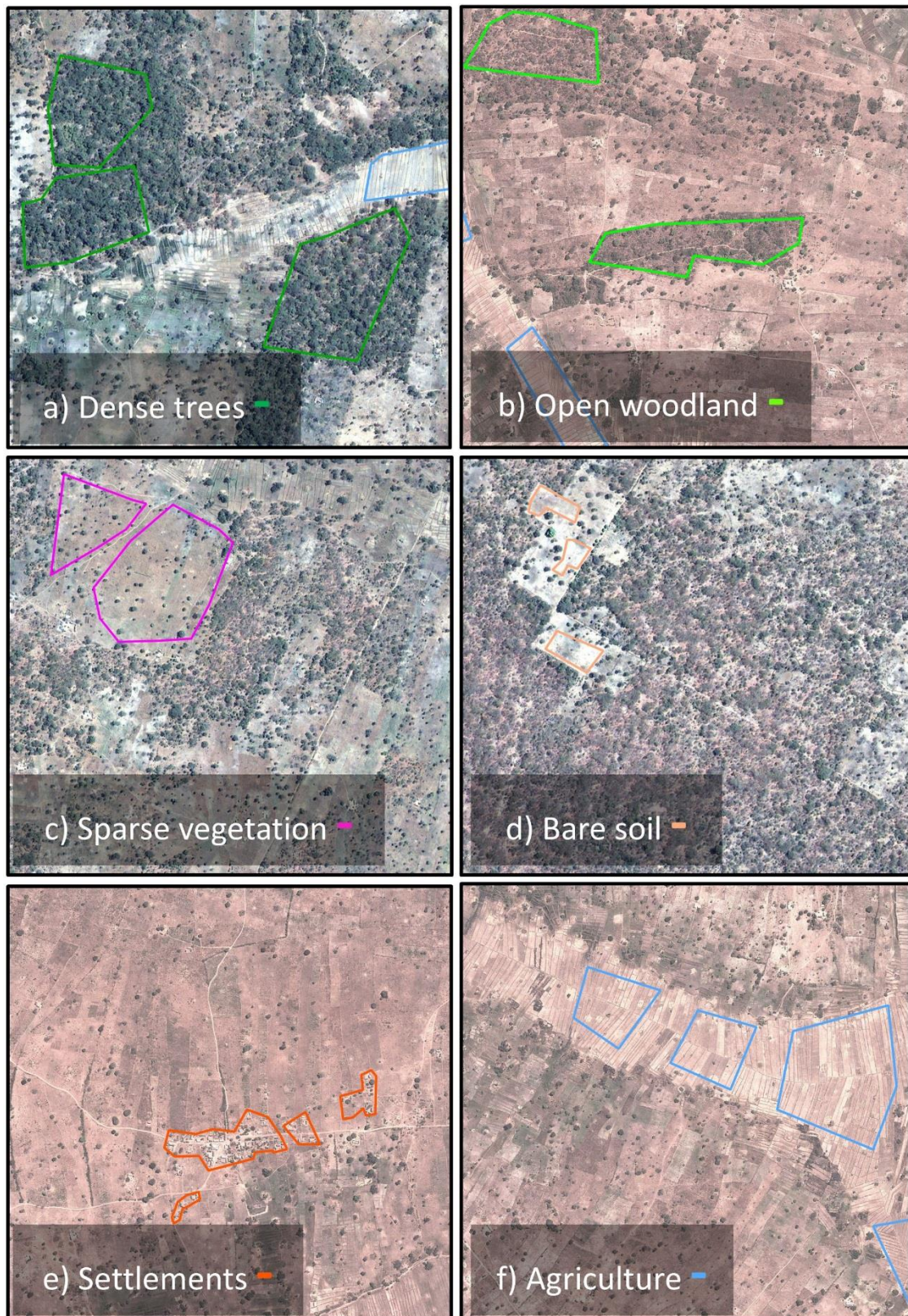
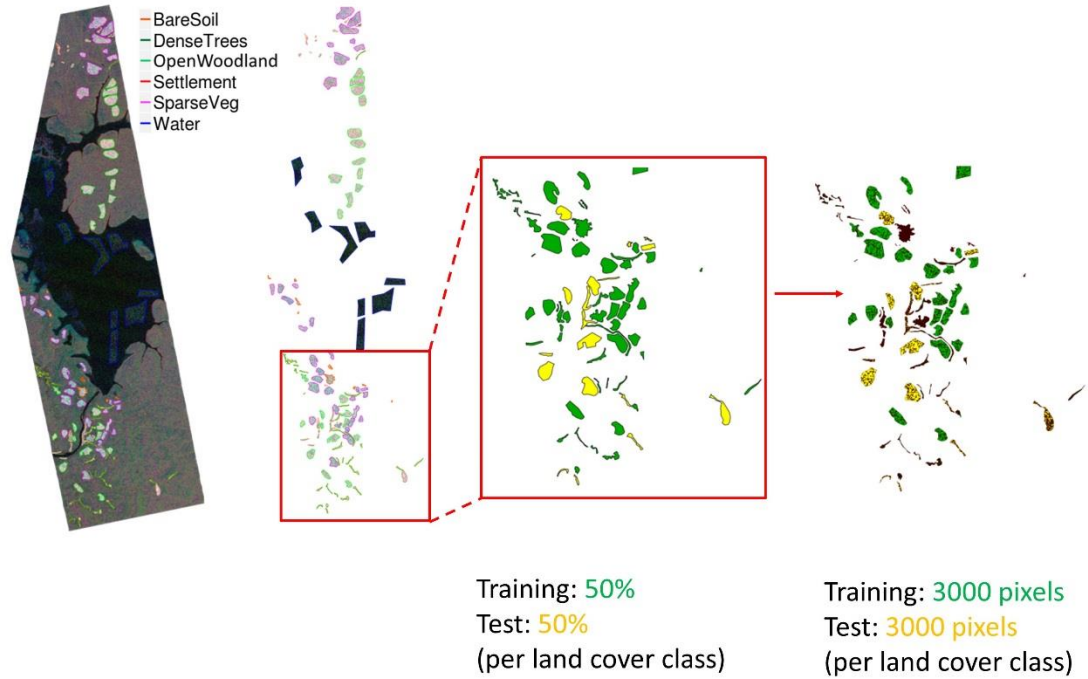


Figure 3.17 Screenshots from Google Earth of selected sample areas of identified land cover classes for the Tanzania study site. Each image is 1km x 1km in size. © 2015 Google, Image © 2015 DigitalGlobe.



## Cameroon



## Tanzania

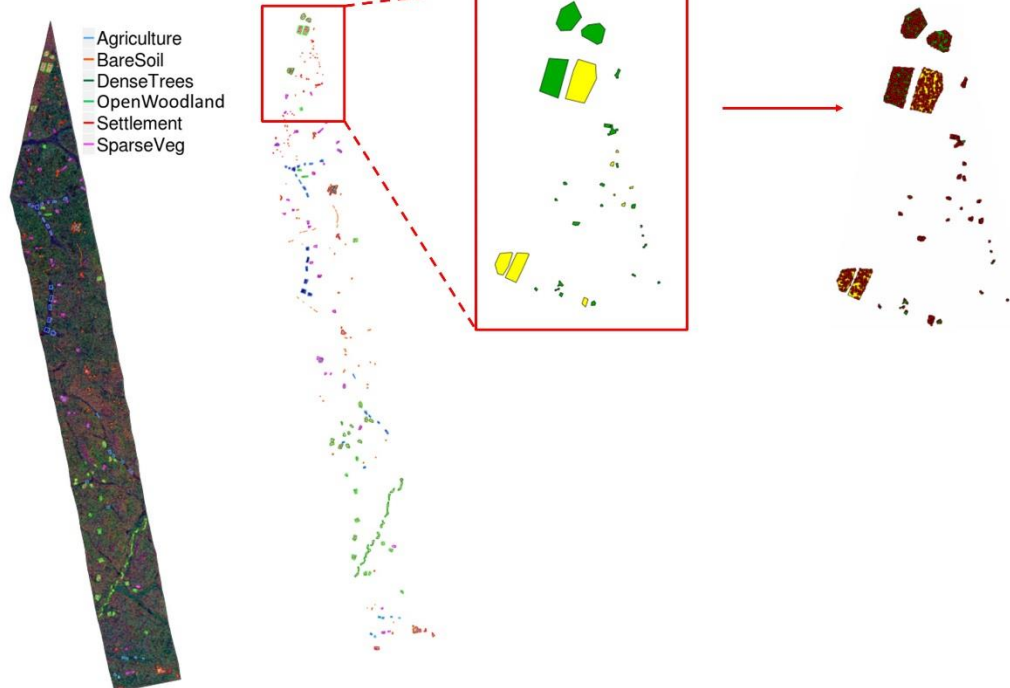


Figure 3.18 Sample selection process for the Cameroon and Tanzania sites. © 2015 Google, Image © 2015 DigitalGlobe.

The number of training and test samples were chosen as 3000 per land cover class, since larger samples ran into computational and memory constraints during the sample extraction step. Sample sizes of 1000, 3000 and 5000 were attempted, with 3000 giving satisfactory results, and computationally efficient. The number of pixels from the training and test samples selected for the Cameroon study site are shown in Table 3.6 and for the Tanzania site in Table 3.7. The split between the training and test polygons was 50:50, but due to different numbers of pixels in each of the polygons, the % of pixels in the training and test polygons varies closer to a 40:60 split. For both the Cameroon and Tanzania test sites, 3000 pixels are selected for training and 3000 pixels for test for each land cover class. It can be noted that the Tanzania samples are much smaller areas, and as a result the 3000 pixels are a larger proportion of the total number of available sample pixels.



Table 3.6 Number of training and test samples available, training and test sample percentage split, and 3000 samples as a percentage of the available land cover samples, for the Cameroon study site.

Cameroon										
Land cover	Nr of training polygons	Nr of test polygons	Total nr of polygons	Available training samples	Available test samples	Total available samples	% pixels in available training samples	% pixels in available test samples	Chosen 3000 samples as % of available pixels	
									for training	for test
Bare soil	10	10	20	18408	19423	37831	49	51	16	15
Dense trees	17	16	33	21568	33924	55492	39	61	14	9
Open woodland	12	12	24	112954	170696	283650	40	60	3	2
Sparse vegetation	16	16	32	154016	202353	356369	43	57	2	1
Water	4	4	8	232548	163835	396383	59	41	1	2
Settlement	10	8	18	9288	10269	19557	47	53	32	29

Table 3.7 Number of training and test samples available, training and test sample percentage split, and 3000 samples as a percentage of the available land cover samples, for the Tanzania study site.

Tanzania										
Land cover	Nr of training polygons	Nr of test polygons	Total nr of polygons	Available training samples	Available test samples	Total available samples	% pixels in available training samples	% pixels in available test samples	Chosen 3000 samples as % of available pixels	
									for training	for test
Agriculture	15	15	30	25608	16193	41801	61	39	12	19
Bare soil	44	46	90	5442	5509	10951	50	50	55	54
Dense trees	15	15	30	27788	26770	54558	51	49	11	11
Open woodland	25	25	50	28108	17935	46043	61	39	11	17
Sparse vegetation	15	15	30	13762	17645	31407	44	56	22	17
Settlement	9	9	18	15789	19332	35121	45	55	19	16

## **Chapter Four     Data preparation**

For the SAR data to be combined and compared between different sensors, all images need to be processed to the desired level. These pre-processing steps include orthorectification, calibration and mosaicking, calculating new information layers of interchannel ratios and texture calculations; and adding elevation, slope and aspect data from a SRTM DEM; and then combining it all in a layer stack (Figure 4.1). The data preparation and pre-processing steps follow the site selection and data selection from the archives as discussed in Chapter Three. Data preparation was necessary to translate the data in the correct format to be able to develop a classification model with the data. The data preparation consists of several pre-processing steps of: orthorectification of the images (Section 4.1), creating an image overlap (Section 4.2), calculating derived layers (Section 4.3) and combining the data all into a layer stack (Section 4.4). After the pre-processing steps, the data was extracted based on the land cover samples selected in Section 3.4.

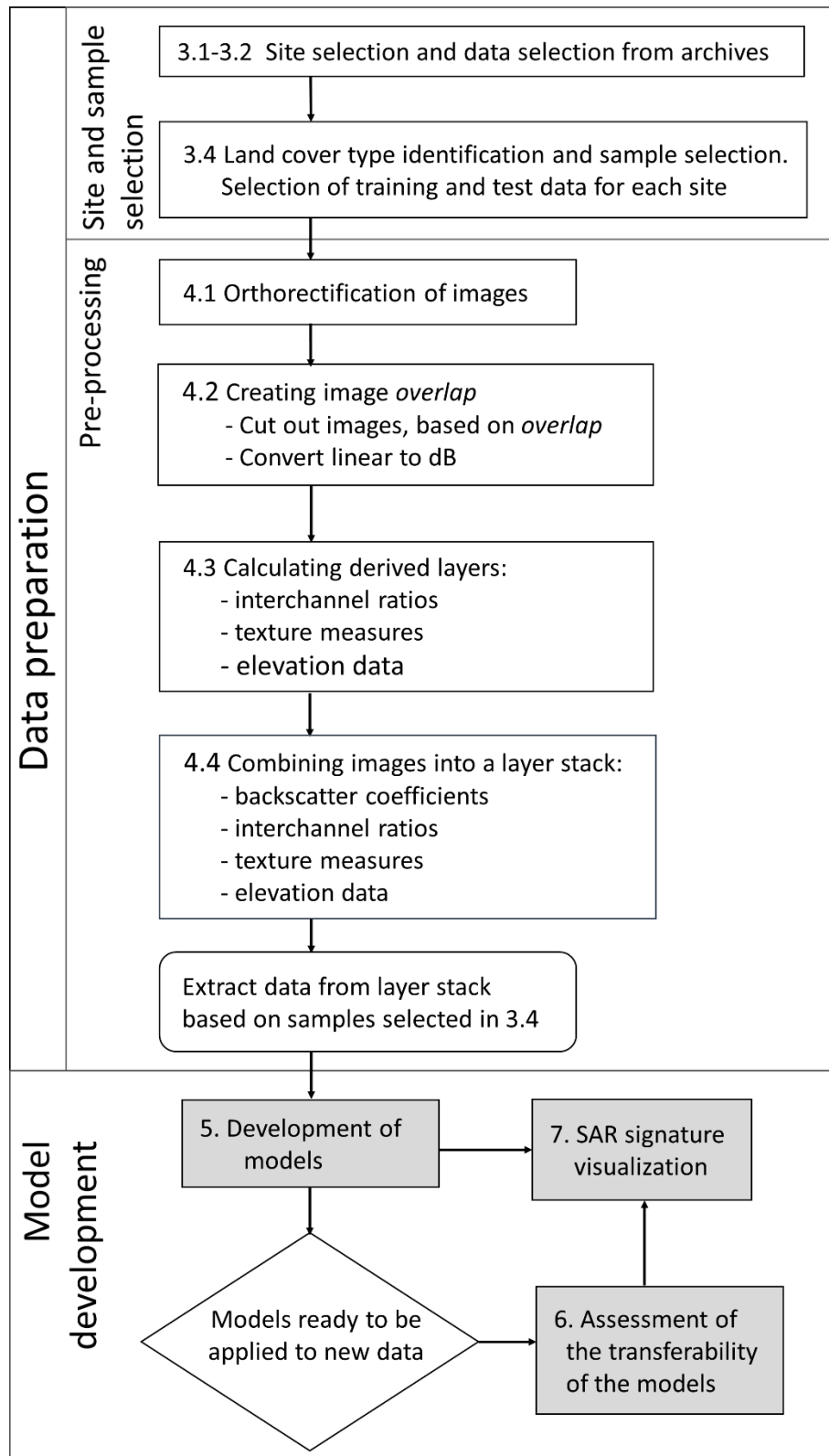


Figure 4.1 Overview of the pre-processing steps, leading up to the model development. Numbers correspond to chapter and section numbers.

The land cover types were selected by identifying several land cover types, using the available very high resolution DigitalGlobe imagery. The land cover polygons were drawn and then randomly split into training and test areas from which samples were randomly selected for model development.

These five data preparation steps listed above, are performed on the data for the main model development and additional study sites, i.e. for the Cameroon and Tanzania study sites, for main model development, for the forested and semi-arid areas respectively, and for the additional sites in the DRC and Chad. Once the data preparation was complete (pre-processing and sample selection), the data were in a format ready for model development and for comparison between several classification models (Chapter Five). After the models have been developed, they are ready to be applied to new data. The transferability of the models was tested, by applying the models on data from a different season and an additional site for the forested and semi-arid ecoregions (Chapter Six). Finally, the SAR signatures were visualized and compared for all the selected land cover types for the forested and semi-arid study sites (Chapter Seven).

#### 4.1 Orthorectification of images

The main aim of the orthorectification or geo-terrain correction step is to align the images from the different SAR sensors with each other and to enable the placement of images on to a map, thus linking each pixel with a specific location on the earth's surface. This process can usually be done relatively easily using ESA's SAR software, Next ESA SAR Toolbox (NEST), but following this route did not allow all the images to overlap properly. The TerraSAR-X images and ENVISAT ASAR images overlapped well, but the ALOS PALSAR images did not match, due to incorrect orbital state vector information in the metadata. This is a known issue, and after applying the *ALOS deskewing application* within NEST, it still did not lead to satisfactory results.

To orthorectify the images, the GAMMA SAR and Interferometric software (Werner et al., 2000) was used. To automate the orthorectification using GAMMA, a Linux shell

script, originally written by Waldram (2014), was adapted and customized for this research. Waldram (2014) used the script to orthorectify multi-temporal ALOS PALSAR images to classify AGB for a region in Central Kalimantan in Indonesia. The script was modified to orthorectify ENVISAT ASAR and TerraSAR-X images in addition to the ALOS PALSAR images, and was run on all the available single-look complex (SLC) images. As part of the GAMMA script, the images need a reference image to be aligned to. As a reference image, panchromatic Landsat images were mosaicked together, for each study site. The reference image needs to cover a larger geographic area than the SAR image that is being orthorectified. Other options are to use an SRTM DEM or a combination of a Landsat image and SRTM DEM (Lucas et al., 2007) or an already orthorectified SAR image. The use of panchromatic Landsat images was chosen as it has a higher spatial resolution of 15x15m compared to the SRTM DEM with 90x90m. This is closer in line to the SAR images with a spatial resolution between 5-15m, as is discussed in more detail later in this section. The use of panchromatic Landsat images produced good results for most of the images.

As part of the GAMMA orthorectification steps, the final model fit of the orthorectification procedure is reported in the metadata. The maximum final model fit standard deviation in the range and azimuth direction for each site/season/sensor combination are presented in Table 4.1. All the images used in this research produced sub-pixel orthorectification accuracy, with the exception of the DRC wet season L-band images, which had approximately 1.5 pixels standard deviation in both the range and azimuth directions. The orthorectification of the Tanzania wet season TerraSAR-X images resulted in large standard deviation errors of 22 pixels in both the range and azimuth directions. As a result, it was therefore decided not to use the Tanzania wet season TerraSAR-X images for the remainder of the research. The enhanced ellipsoid corrected (EEC) TerraSAR-X images were already orthorectified by the DLR. A visual comparison was undertaken to establish whether these images are aligned to the ALOS PALSAR and ENVISAT ASAR images.

Table 4.1 The images used in this research with the final model fit standard deviation in the range and azimuth directions.

Site	Season	Sensor	SLC/EEC	Number of images	Polarisations	Maximum final model fit standard deviation (samples)	
						Range	Azimuth
Cameroon	Wet	L	SLC	4	Quad	0.45	0.63
		C	SLC	2	Dual VV/VH	0.97	0.52
		X	SLC	4	Dual HH/HV, Dual VV/VH	0.65	0.80
Cameroon	Dry	L	SLC	1	HH	0.32	0.67
		C	SLC	2	Dual VV/VH	0.97	0.47
		X	SLC	4	Dual HH/HV, Dual VV/VH	0.65	0.77
DRC	Wet	L	SLC	5	Quad HH	1.49	1.59
		C	SLC	3	VV Dual HH/HV	0.37	0.58
		X	EEC	2	HH, VV	<sub>-3</sub>	<sub>-3</sub>
Tanzania	Dry	L	SLC	6	Quad Dual HH/HV	0.54	0.82
		C	SLC	2	Dual VV/VH	0.38	0.43
		X	SLC	1	VH <sup>4</sup>	0.59	0.61
Tanzania	Wet	L	SLC	2	HV, VH <sup>4</sup>	0.32	0.48
		C	SLC	2	Dual VV/VH	<sub>-1</sub>	<sub>-1</sub>
		X	SLC	2	HV, VH <sup>4</sup>	22.05 <sup>2</sup>	22.56 <sup>2</sup>
Chad	Dry	L	SLC	6	Quad Dual HH/HV	0.40	0.34
		C	SLC	2	Dual HH/HV	0.31	0.02
		X	EEC	2	Dual HH/HV	<sub>-3</sub>	<sub>-3</sub>

<sup>1</sup> The geocoding information is stored on the ALICE server for three months, and this information was unfortunately lost as a result. Upon visual inspection it was decided to still use these images.

<sup>2</sup> The high range and azimuth values for the Tanzania wet season TerraSAR-X images. It was decided not to use these images in the analysis, due to the sub-optimal orthorectification result.

<sup>3</sup> The EEC images were ordered orthorectified, and these were used as is. A visual comparison was done to assess the alignment with the ALOS PALSAR, ENVISAT ASAR images.

<sup>4</sup> Not all SLC polarisation images resulted in orthorectified images with the GAMMA script. Only the SLC images that successfully orthorectified by the GAMMA script are listed here.

Figure 4.2 shows the GAMMA script steps, which include multi-looking, radiometric calibration, geocoding/orthorectification of the image and creation of an ENVI header file (.hdr), to enable the orthorectified image to be opened in any standard remote sensing or GIS software.

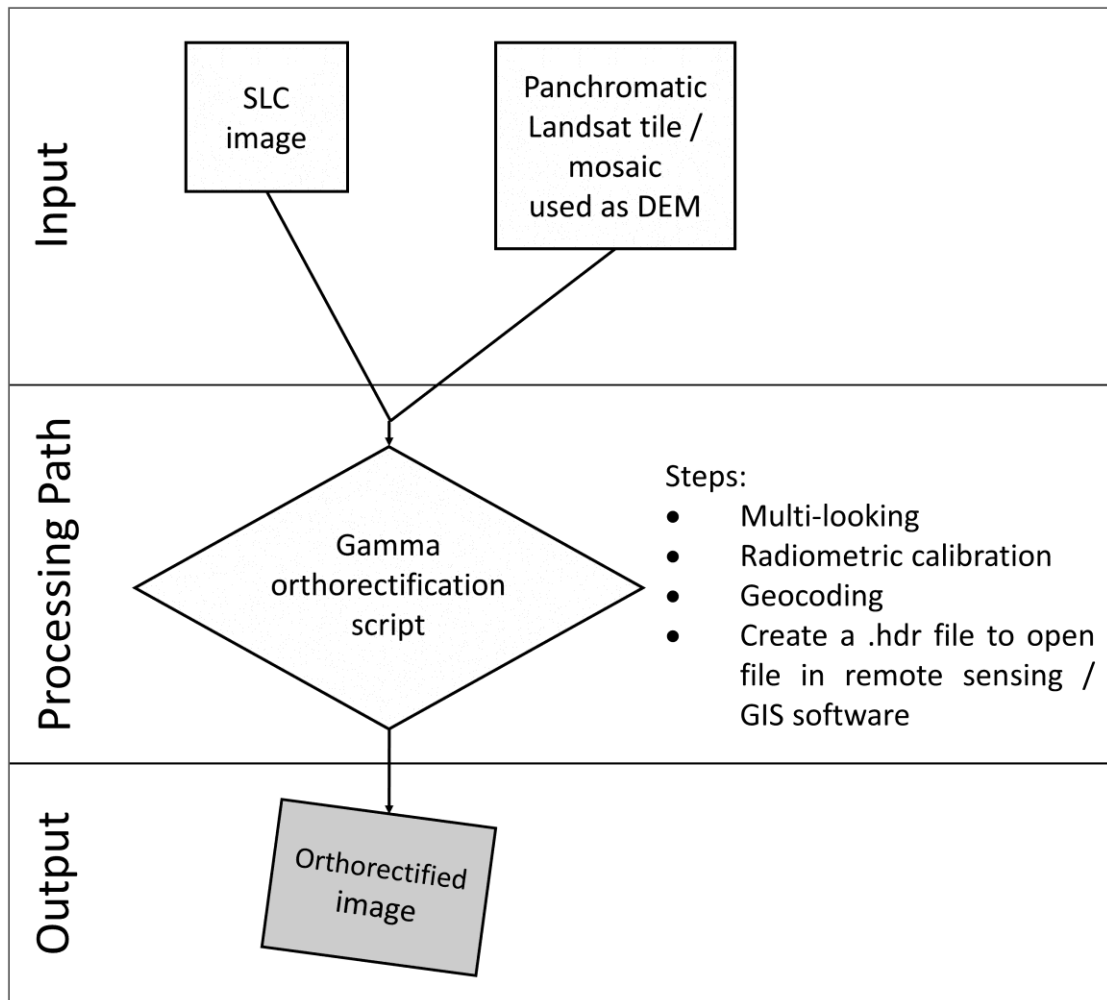


Figure 4.2 First step of the pre-processing workflow, to create orthorectified images.

Figure 4.3 shows a comparison of the spatial resolution of different sensors after the multi-looking step. The images are multi-looked and resampled to the nearest 5m interval, as set out in Table 4.2, to allow for relatively easy layer comparison and stacking of the images. The TerraSAR-X images were first multi-looked and then resampled to 5m, ALOS PALSAR to 10m, and ENVISAT ASAR to 15m spatial resolution. After the



GAMMA orthorectification script, the images were still in a linear format, and were converted to a dB scale using a log conversion in the following steps.

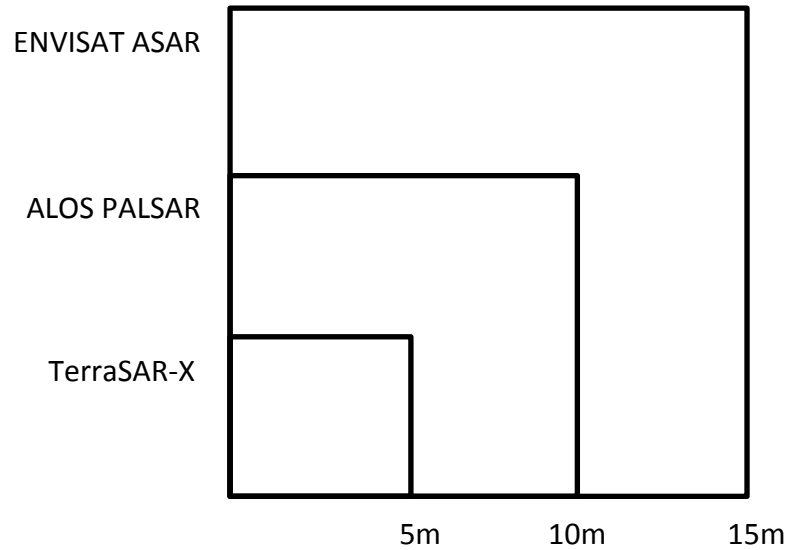


Figure 4.3 Spatial resolution comparison for the multi-looked and spatially resampled images used in this study, from ENVISAT ASAR, ALOS PALSAR and TerraSAR-X.

The GAMMA script was run on the ALICE High Performance Computing Facility at the University of Leicester, due to the high volume and large size and of the files. Additionally, ALICE was the only access to an implementation of the GAMMA software at the University of Leicester.

Table 4.2 Multi-looking applied in the range and azimuth direction for each sensor to achieve the listed spatial resolutions.

Sensor	Original Spatial resolution Range (m)	Original Spatial resolution Azimuth (m)	Multi-looking Range	Multi-looking Azimuth	Spatial resolution Range (m)	Spatial resolution Azimuth (m)	Resampled Spatial resolution, Range and Azimuth (m)
<b>ALOS PALSAR</b>	9.369	3.534	1	3	9.369	10.603	10m
<b>ENVISAT ASAR</b>	7.804	3.954	2	4	15.608	15.818	15m
<b>TerraSAR-X</b>	0.909	2.410	5	2	4.547	4.821	5m

## 4.2 Creation of image overlap

As was shown when the study sites were introduced in Chapter Three, only a relatively small area coincides between the images from the three sensors for each site. The steps of how to create this overlap-area are shown in Figure 4.4. Starting with the orthorectified images from the GAMMA script, the images were copied from ALICE to a local laptop. Some file manipulation steps were necessary, such as moving the files all to a single folder, converting the images from ENVI format to .tif files and mosaicking the images that were acquired as part of the same orbit.

A shapefile was created of each image, which is then used to obtain the area of overlap between all the shapefiles. The ENVI to .tif conversion, mosaicking, shapefile creation and overlap calculation were all performed using the Geospatial Data Abstraction Library (GDAL) (GDAL, 2013).

After an overlap for each study site was obtained, the overlap shapefile was used to extract that portion of the data from each of the images. After the relevant portions of the images were obtained, the intensity images were converted from a linear scale to sigma-naught ( $\sigma^0$ ), which is in a decibel (dB) scale, with Equation 4-1:

$$\sigma_{PQ}^0 = 10 \times \log_{10}(S_{PQ}) \quad \text{Equation 4-1}$$

where  $S_{PQ}$  is the linear scale intensity in PQ polarization, with P either horizontally or vertically transmitted, and Q either horizontally or vertically received. The  $\sigma^0$ -images are now in the familiar -0.40 to -0.05 range (approximately), and are used in this form for the classification model development.

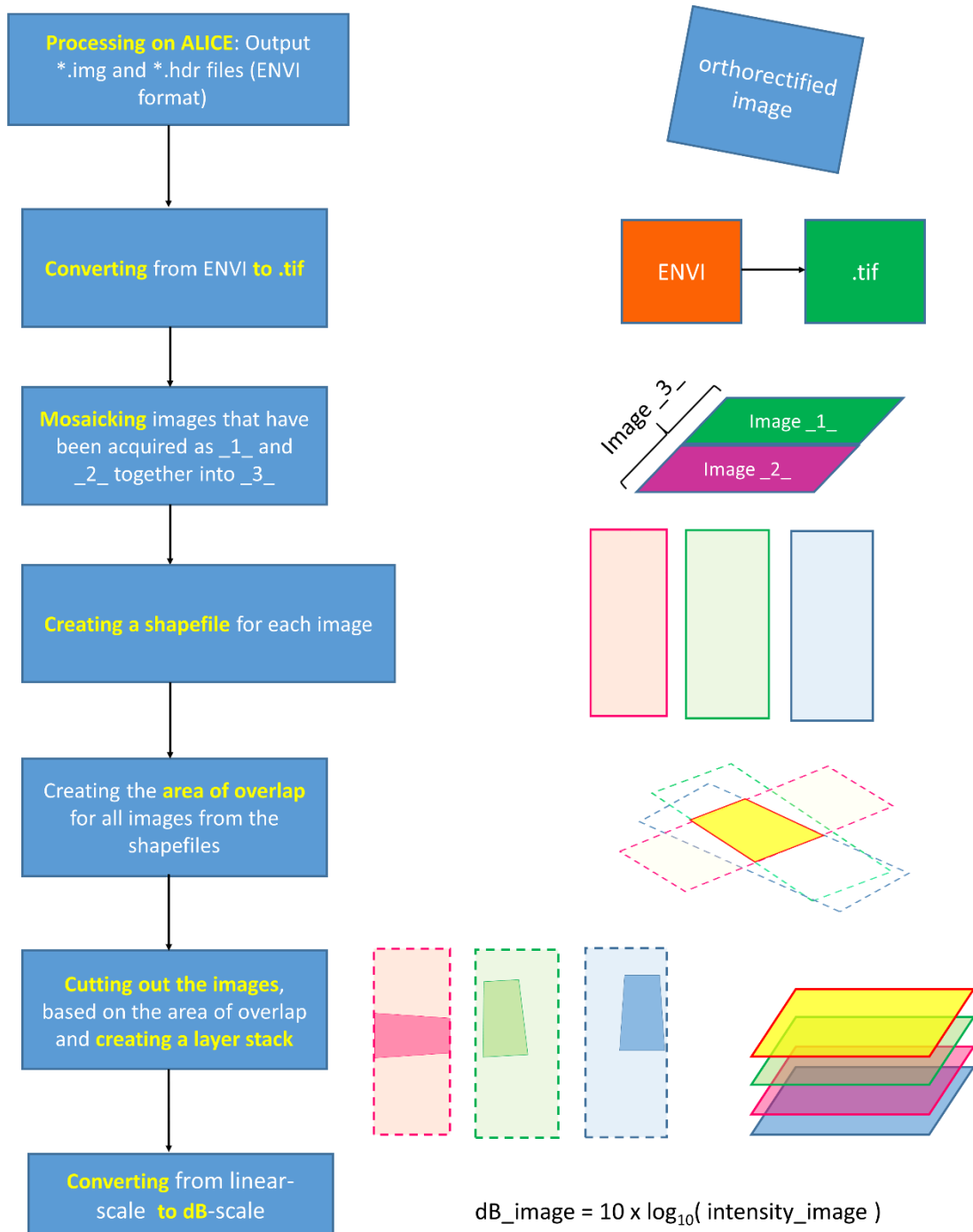


Figure 4.4 Second step of the pre-processing workflow, creating overlap areas from the SAR images for each site, creating a layer stack and converting from linear to dB scale.

### 4.3 Calculation of derived layers

Three sets of derived layers were calculated, namely interchannel ratios, texture measures and DEM-derived layers. This increases the data dimensionality from just the backscatter coefficients to a multitude of information layers that can be used for classification model development.

#### 4.3.1 Interchannel ratios

In terms of interchannel ratios, there are several image manipulations that can be calculated to enhance the information content between the various polarisation channels (e.g. ratios, sums and differences). The phase-related calculations, such as the phase difference and fully polarimetric decompositions, are excluded from this study. This was necessary because this study aimed to derive a classification algorithm from the channels that can be used from dual-polarized and single polarized SAR images.

The interchannel ratios that were calculated are listed in Table 4.3. Most of the interchannel ratios were performed on the linear scale of the SAR images. The backscatter coefficients  $\sigma_{HH}^0$ ,  $\sigma_{VV}^0$  and  $\sigma_{HV}^0$  along with the co-polarisation ratio and depolarisation ratio have been used in sea-ice classification from SAR images (Dierking et al., 2003). The cross-polar backscatter coefficient  $\sigma_{HV}^0$  have been used for agricultural crop classification whilst the ratios in dB have been used for bare soil and vegetation classification (Buckley, 2002; Quegan et al., 2003). The linear ratios and linear sum calculations have been used for land cover and terrain characterization (Simental et al., 2005). Several interchannel ratios have been added in this study, that were not mentioned in Simental et al. (2005). These are:  $S_{HH} + S_{HV}$ ,  $S_{VV} + S_{VH}$ ,  $S_{VV} - S_{VH}$ ,  $\frac{S_{HH}-S_{HV}}{S_{HH}+S_{HV}}$  and  $\frac{S_{VV}-S_{VH}}{S_{VV}+S_{VH}}$ . This is to complete the possible feature space of calculations. The normalised difference interchannel index calculations are similar in information content to the linear difference calculations, but normalise the images in a similar way in which the normalised difference vegetation index (NDVI) is calculated from optical imagery.

Table 4.3 The interchannel ratios, sums and differences used in this study, extended from Lönnqvist et al. (2010).

Name of calculation	Equation	Eq.	Previous application	Reference
Backscatter coefficients	$\sigma_{HH}^0 = 10 \cdot \log_{10}(S_{HH})$	4-2	Sea ice	(Dierking et al., 2003; Skriver et al., 2003)
	$\sigma_{VV}^0 = 10 \cdot \log_{10}(S_{VV})$	4-3		
	$\sigma_{HV}^0 = 10 \cdot \log_{10}(S_{HV})$	4-4	Agriculture (crops), sea ice	(Quegan et al., 2003; Skriver et al., 2003)
Ratios in dB	$rdB_{VH/VV} = 10 \cdot \log\left(\frac{ S_{VH} ^2}{ S_{VV} ^2}\right)$	4-5	Bare soil / vegetation	(Quegan et al., 2003)
	$rdB_{VV/HH} = 10 \cdot \log\left(\frac{ S_{VV} ^2}{ S_{HH} ^2}\right)$	4-6	Vegetation types	(Buckley, 2002)
	$rdB_{HV/HH} = 10 \cdot \log\left(\frac{ S_{HV} ^2}{ S_{HH} ^2}\right)$	4-7		
Co-polarisation ratio	$\gamma = \frac{\sigma_{VV}^0}{\sigma_{HH}^0}$	4-8	Sea ice	(Dierking et al., 2003)
Depolarisation ratio	$\delta = \frac{\sigma_{VV}^0}{\sigma_{HH}^0 + \sigma_{VV}^0}$	4-9		(Dierking et al., 2003; Skriver et al., 2003)
Linear ratios	$rlinear_{HH/VV} = \frac{S_{HH}}{S_{VV}}$	4-10	Terrain characterization, land cover	(Simental et al., 2005) ( <sup>1</sup> ) Variation of (Simental et al., 2005)
	$rlinear_{VH/VV} = \frac{S_{VH}}{S_{VV}}$	4-11		
	$rlinear_{HV/HH} = \frac{S_{HV}}{S_{HH}}$	4-12		
Linear Sums	$S_{HH} + S_{VV}$	4-13		
	$S_{HH} + S_{HV}^{(1)}$	4-14		
	$S_{VV} + S_{VH}^{(1)}$	4-15		
Linear Differences	$S_{HH} - S_{VV}$	4-16		
	$S_{HH} - S_{HV}$	4-17		
	$S_{VV} - S_{VH}^{(1)}$	4-18		
Normalised difference interchannel indices	$\frac{S_{HH} - S_{VV}}{S_{HH} + S_{VV}}$	4-19		
	$\frac{S_{HH} - S_{HV}^{(1)}}{S_{HH} + S_{HV}^{(1)}}$	4-20		
	$\frac{S_{VV} - S_{VH}^{(1)}}{S_{VV} + S_{VH}^{(1)}}$	4-21		

#### 4.3.2 Texture measures of SAR images

To increase data dimensionality further, several texture measures are calculated for each of the SAR images. The Grey-Level Co-occurrence Matrix (GLCM) texture measurements are the most widely used texture measures for images (Haralick et al., 1973). Haralick et al. assessed the use of several texture measures for different types of image data, including multispectral satellite imagery from Landsat 1 with seven land-use categories. A piecewise linear discriminant function classification algorithm was used for the multispectral satellite image, giving 83% classification accuracy on a test data set.

Fourteen different texture measures were introduced by Haralick et al. (1973). They are Angular Second Moment (ASM), Contrast (CON), Correlation (COR), Sum of Squares Variance (VAR), Inverse Difference Moment (IDM), Sum Average (SA), Sum Variance (SV), Sum Entropy (SENT), Entropy (ENT), Difference Variance (DVAR), Difference Entropy (DENT), two measures of Information Measures of Correlation (IMCOR1, IMCOR2) and Maximal Correlation Coefficient (MAXCOR). These texture measures use the matrix known as the Grey-Level Co-occurrence Matrix (GLCM) to derive the texture measures. The GLCM is a matrix that defines how often each grey-level occurs next to every other grey-level in a certain distance and direction. This is calculated for an image using a moving window at a specific square window size (e.g. 3x3, 5x5, 7x7 etc). The GLCM is given by

$$GLCM = \begin{bmatrix} P(1,1) & P(1,2) & \cdots & P(1,N_g) \\ P(2,1) & P(2,2) & \cdots & P(2,N_g) \\ \vdots & \vdots & \ddots & \vdots \\ P(N_g,1) & P(N_g,2) & \cdots & P(N_g,N_g) \end{bmatrix} \div \sum_{i,j=1}^{N_g} P_{ij}$$

$$= \begin{bmatrix} p(1,1) & p(1,2) & \cdots & p(1,N_g) \\ p(2,1) & p(2,2) & \cdots & p(2,N_g) \\ \vdots & \vdots & \ddots & \vdots \\ p(N_g,1) & p(N_g,2) & \cdots & p(N_g,N_g) \end{bmatrix}$$

Equation  
4-22

where  $N_g$  is the number of distinct grey levels in the image, and element  $P(i,j)$  is the number of times a pixel with value  $i$  is next to (or at a specified distance away from) a

pixel with value  $j$ . The matrix is then normalized by dividing each element by the total number of occurrences, resulting in a probability (Haralick et al., 1973; Boland, 1999). The GLCM probability measure, which is dependent on the inter-pixel distance ( $\delta$ ) and orientation ( $\theta$ ), can therefore be defined by:

$$\Pr(x) = \{ p(i, j) \mid (\delta, \theta) \} \quad \text{Equation 4-23}$$

where the co-occurrence probability between grey levels  $i$  and  $j$  is defined as:

$$p(i, j) = \frac{P_{ij}}{\sum_{i,j=1}^{N_g} P_{ij}} \quad \text{Equation 4-24}$$

where  $P_{ij}$  is the number of occurrences of grey levels  $i$  and  $j$  within a given window, given the inter-pixel distance  $\delta$ , orientation  $\theta$  and the number of distinct grey levels  $N_g$  (Haralick et al., 1973; Clausi, 2002). To enable the calculation of the GLCM matrices, all the SAR images are rescaled between 1 and 64 in the linear domain. This is called the quantization level. The quantization level can affect the classification results and contrary to expectation, a higher quantization level (e.g. 128 or 256) can have lower classification accuracy than a lower quantization level. Contrast and Entropy are the two recommended texture measures by Clausi (2002), and a quantization of greater than 24 grey levels is also recommended by Clausi (2002).

The GLCM can be calculated in four orientation directions, namely  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  and  $135^\circ$ . What is often done to derive orientation invariant texture measures, is to calculate the average of all four orientation angles, as is done in this study. A summary of GLCM texture measures used for SAR image classification in other publications is listed in Table 4.4. As first order statistics, which use the pixel values directly, the neighbourhood mean (MEAN) for a given window size is calculated. As second order statistics, which use the co-occurrence of grey-levels in a given window size, four GLCM texture measures are calculated in this study, namely contrast, correlation, entropy and variance.

Table 4.4 Summary of texture measures used in other studies for the classification of SAR images.

Publication	Application	Sensor	Spatial resolution	Texture measure	Window size
Kuplich et al., 2005	Biomass estimation	JERS-1 (L-band)	25m	CON	3x3,5x5
Peng et al., 2005	Land cover, mountains	Radarsat	12.5m	CON,ENT,COR, ASM,MEAN	17x17
Herold et al., 2004	Water, agriculture, bare, urban	SIR-B (L-band)	12.5m	Variance	21x21
	Urban	Radarsat	25m	Variance	13x13
Clausi, 2002	Sea ice	Aerial SAR	100m	CON, ENT, COR	8x8
		Radarsat	150m		16x16
Anys, 1995	Agricultural crops	Convair-580 (C-band)	12.5m	MEAN, VAR (1 <sup>st</sup> order), CON, VAR (2 <sup>nd</sup> order)	7x7

The neighbourhood mean is calculated by:

$$f_{mean} = \frac{\sum_{k=1}^N x_k}{N} \quad \text{Equation 4-25}$$

where  $N$  is the number of pixels in the window, and  $x_k$  is the grey tone value of pixel  $k$ .

Contrast is given by:

$$f_{contrast} = \sum_{n=0}^{N_g-1} n^2 \left\{ \sum_{\substack{i=1 \\ |i-j|=n}}^{N_g} \sum_{j=1}^{N_g} p(i,j) \right\} \quad \text{Equation 4-26}$$



Correlation is calculated by:

$$f_{correlation} = \frac{\sum_i \sum_j (ij)p(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y} \quad \text{Equation 4-27}$$

where  $\mu_x$ ,  $\mu_y$ ,  $\sigma_x$  and  $\sigma_y$  are the means and standard deviations of  $p_x$  and  $p_y$ .

Entropy is calculated as per Equation 4-28:

$$f_{entropy} = - \sum_i \sum_j p(i,j) \log(p(i,j)) \quad \text{Equation 4-28}$$

with  $p(i,j)$  as defined by Equation 4-22 and Equation 4-23.

Variance is calculated as per Equation 4-29:

$$f_{variance} = \sum_i \sum_j (i - \mu)^2 p(i,j) \quad \text{Equation 4-29}$$

with  $\mu$  the mean of  $p(i,j)$  for the moving window and  $p(i,j)$  defined by Equation 4-24. The variance is also referred to as the Sum of Squares (Haralick et al., 1973).


As mentioned, the texture calculations are applied at a specific window size. To relate the texture measures to features on the ground, the spatial resolution of the image has to be taken into account. The interpretation of the window size is therefore dependent on the spatial resolution of the sensor. Table 4.5 show how the window size and sensor spatial resolution relates to the footprint area on the surface on the ground. The highlighted sections show the window sizes at which the texture measures are calculated for this study.

Figure 4.5 shows a graphical representation of the footprint areas of the texture measures for the different sensors on a relative scale. It can be seen that a 9x9 window size on 5m TerraSAR-X imagery have an equivalent footprint of a 3x3 15m ASAR image,

and is comparable to a 5x5 10m PALSAR image. Similarly a 15x15 window size 5m TerraSAR-X image has the same footprint as a 5x5 15m ASAR image and is similar to a 7x7 10m PALSAR image. This comparison shows that the high spatial resolution of TerraSAR-X has a significant impact on the effectiveness and granularity of the texture calculations compared to the coarser resolutions of ALOS PALSAR and ENVISAT ASAR. The motivation for calculating the contrast, correlation, entropy and variance texture measures along with the neighbourhood mean at the selected window sizes, is to provide many additional information layers to enhance the classification of the chosen land cover types.

*Table 4.5 The footprint area covered in meters by the texture calculation, for each sensor, at different window sizes.*

Window size	TSX (5m)	PALSAR (10m)	ASAR (15m)
3	15	30	45
5	25	50	75
7	35	70	105
9	45	90	135
11	55	110	165
13	65	130	195
15	75	150	225
17	85	170	255
19	95	190	285

 Texture window sizes calculated

The mean, entropy and contrast texture measures of L-HH image, were found useful to distinguish dense trees and bare soil land cover classes (Figure 4.6). Similarly, the mean, entropy and contrast texture measures of X-VV give a similar separation between dense trees and bare soil (Figure 4.7).

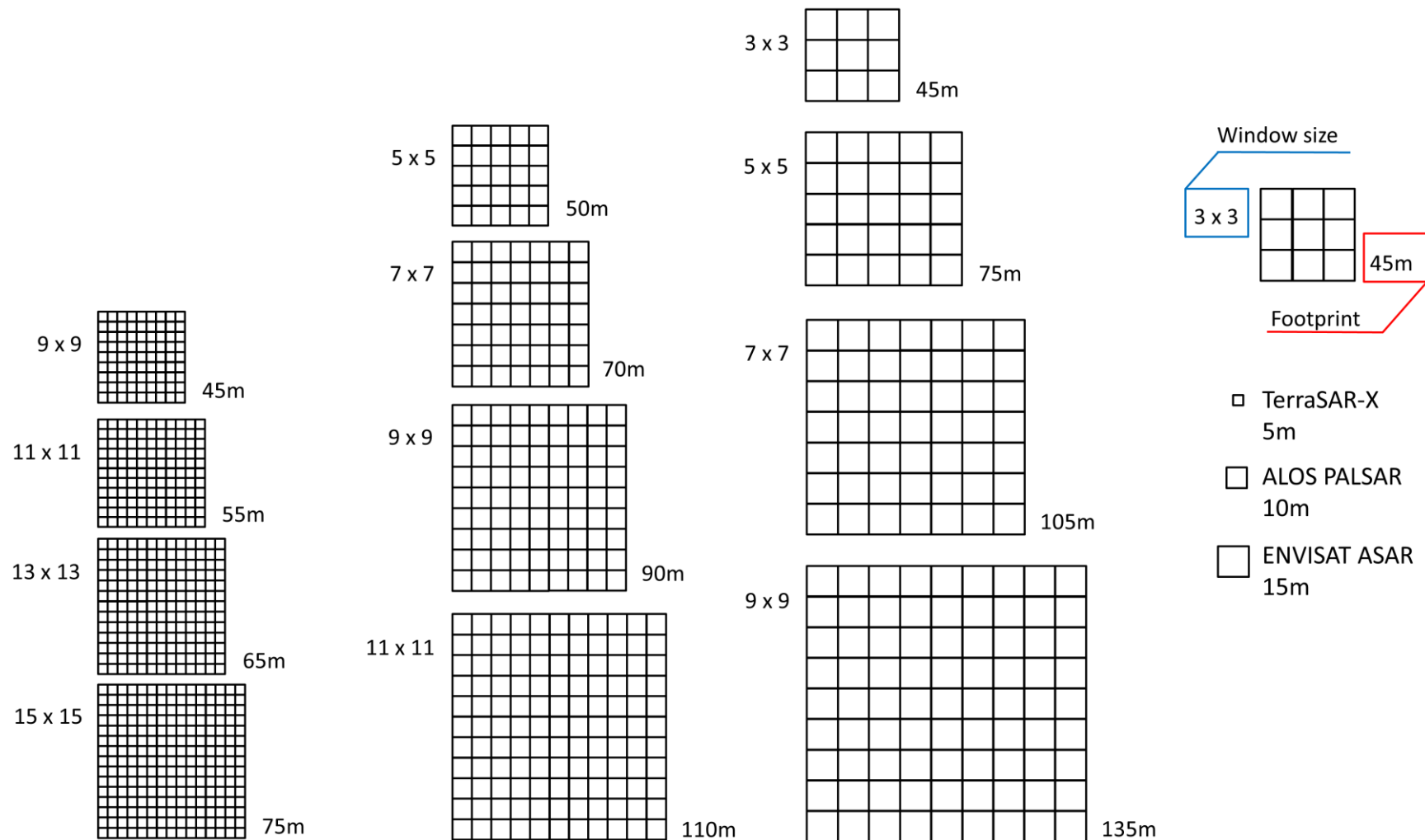


Figure 4.5 Comparison of the footprints of the different window-sizes used for the texture calculations for TerraSAR-X, ALOS PALSAR and ENVISAR ASAR.

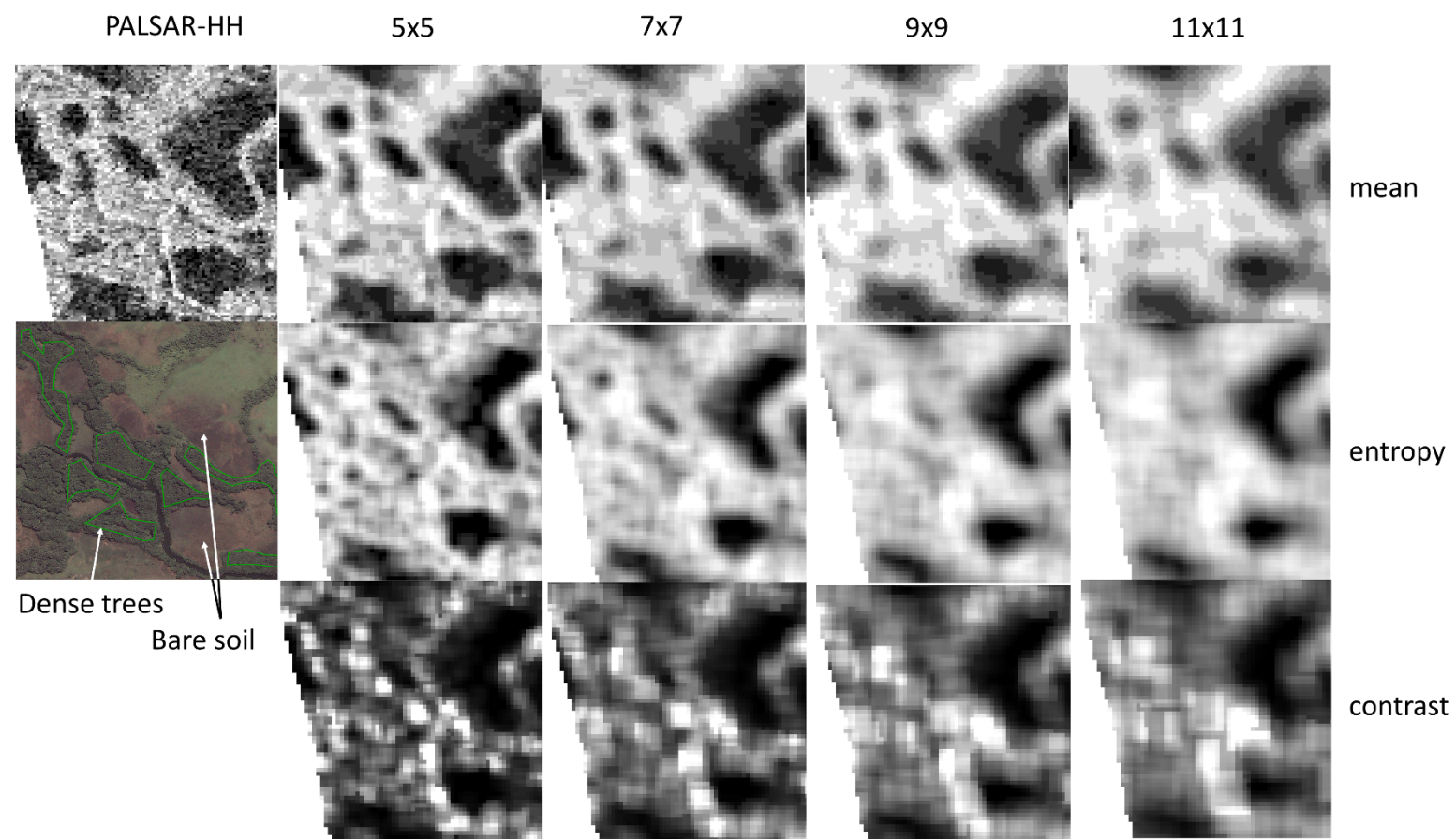


Figure 4.6 Texture measures at various window sizes for ALOS PALSAR (L-HH), highlighting dense trees and bare soil.

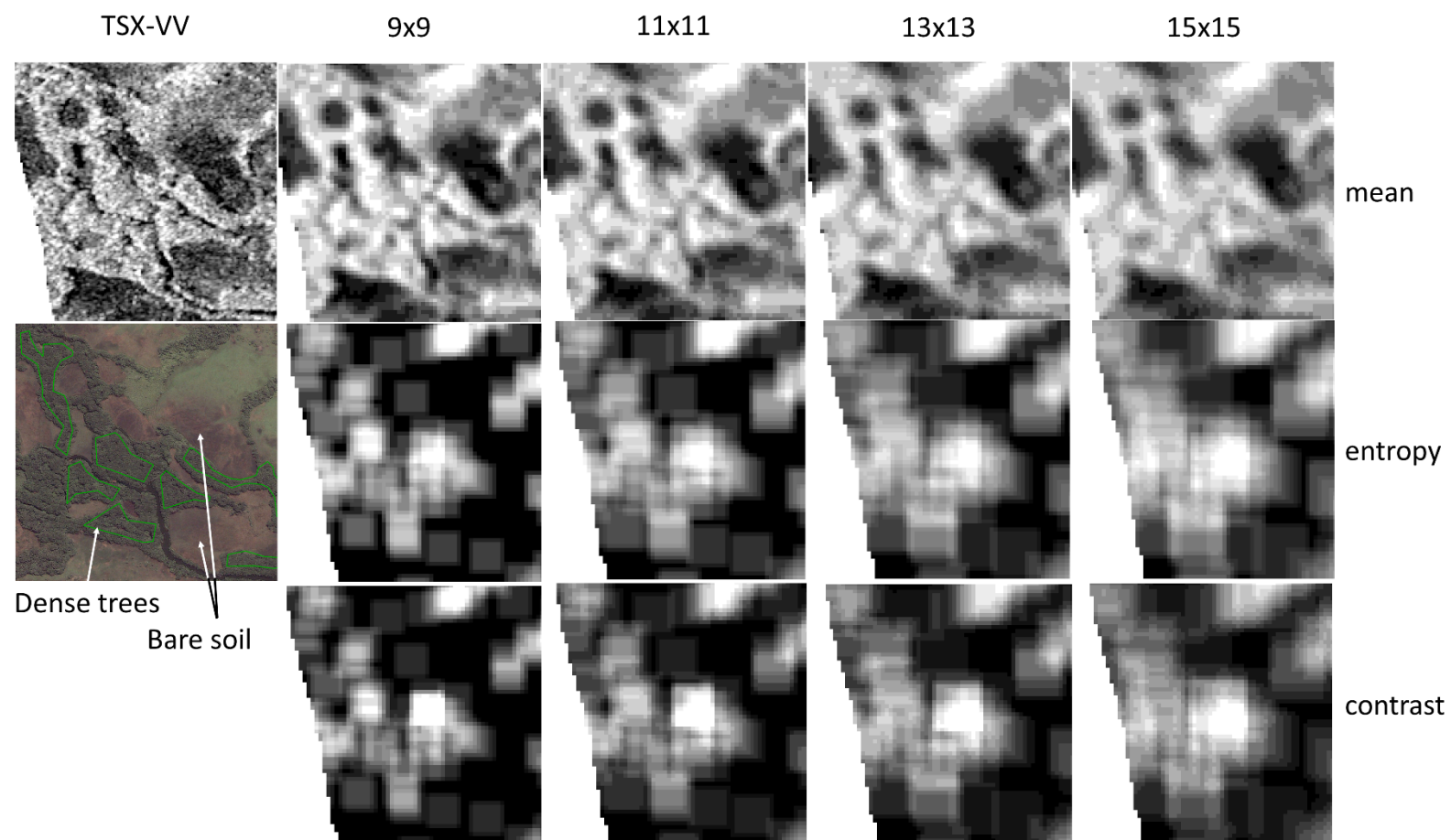


Figure 4.7 Texture measures at various window sizes for TerraSAR-X (X-VV), highlighting dense trees and bare soil.

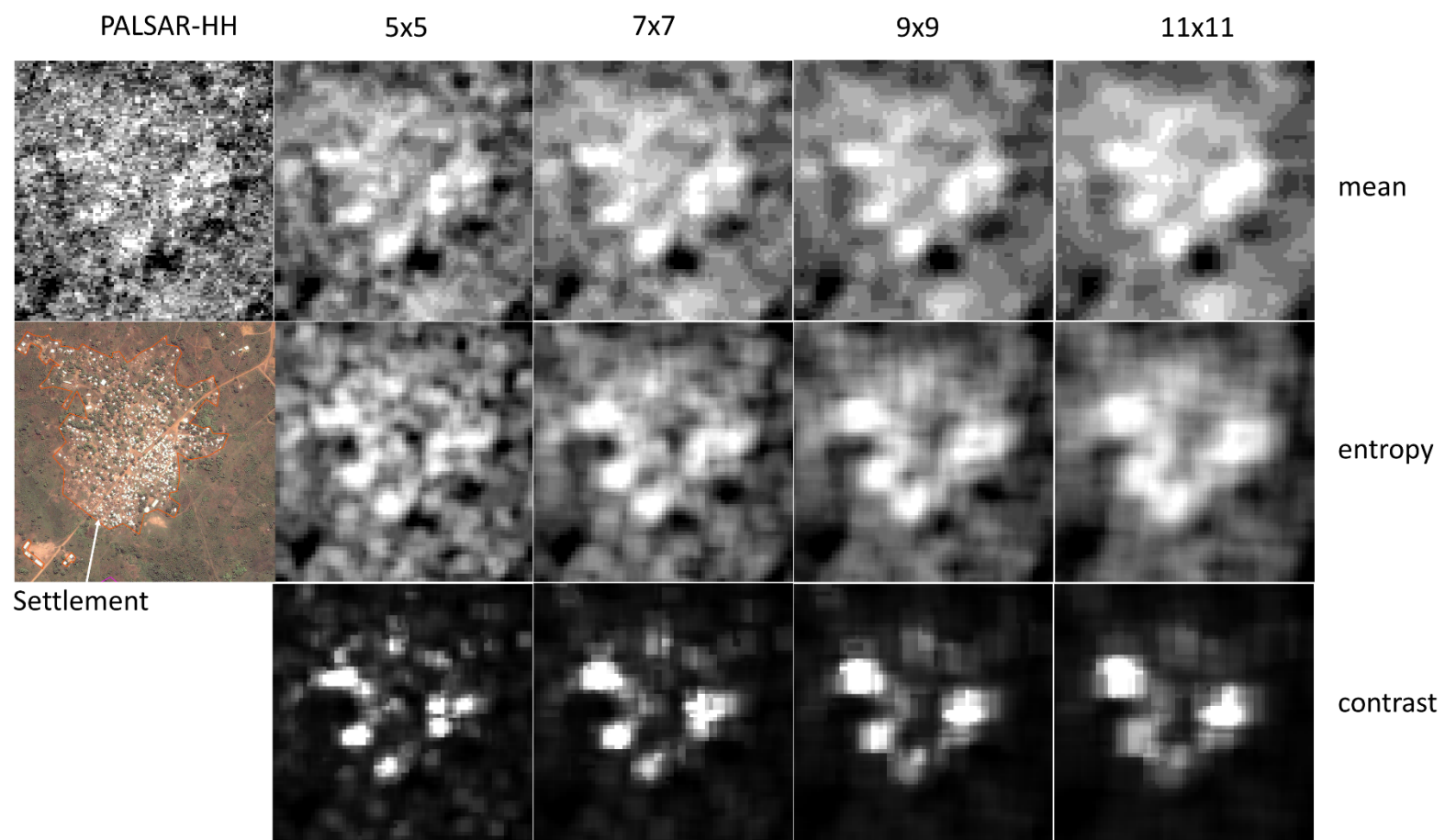


Figure 4.8 Texture measures at various window sizes for ALOS PALSAR (L-HH), highlighting settlements.

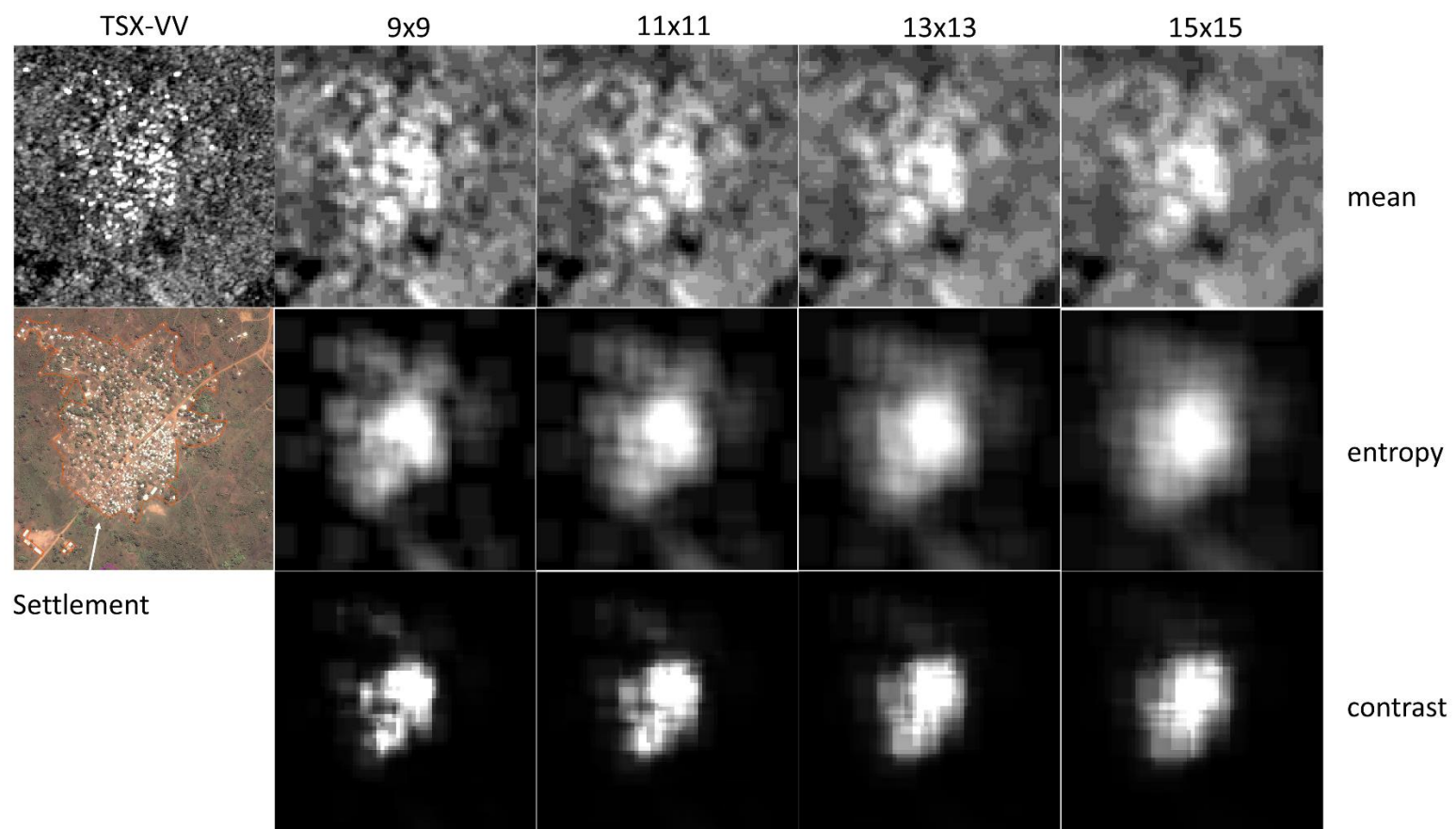


Figure 4.9 Texture measures at various window sizes for TerraSAR-X (X-VV), highlighting settlements.

The entropy and contrast texture measure of X-VV works well to identify settlement areas (Figure 4.8), whereas the contrast texture measure of L-HH emphasised the settlement areas as well (Figure 4.9). The mean texture measure does not highlight the settlement areas as it did for the dense trees and bare soil areas. By comparing the contrast texture measure of L-HH with X-VV, it is noted that different areas of the settlement are highlighted by the two frequencies. The variance texture measure looked very similar to the contrast images, whereas the correlation texture measure did not provide visually meaningful results for the dense trees, bare soil and settlement land cover classes.

#### 4.3.3 Elevation data from DEM

The SRTM elevation data are used as part of the orthorectification data preparation step in Section 4.1. Since the SRTM data are readily available, elevation data along with the derived slope and aspect information are used as additional information layers. Furthermore, the inclusion of elevation, aspect and slope has been shown to increase the accuracy of land cover mapping (Peng et al., 2005).

All interchannel ratios, texture measures and the SRTM-derived elevation layers were processed using the open source software GRASS GIS (Neteler et al., 2012).

#### 4.4 Combining images into a layer stack

To enable utilisation of all the available backscatter coefficient images for each site, along with the derived layers of interchannel ratios, texture measures and DEM-derived layers, all the images were combined into a layer stack. In order to do this the images were all resampled to 5m spatial resolution, using nearest neighbour resampling.

To recap the preceding steps:

- The images is first orthorectified (Section 4.1),
- then, an area of overlap is calculated, which is used to extract the overlapping area from the orthorectified images (Section 4.2).



- Several additional information layers are calculated, namely interchannel ratios, texture measures and DEM-derived layers (Section 4.3).
- All of these layers are then combined into a layer stack (this section, Section 4.4).

Figure 4.10 shows the layer stack creation for the Cameroon and Tanzania study sites. First, the stack of multi-frequency SAR images is shown on the left hand side of the figure. Secondly, the separation of images into dry season and wet season sets is shown in the centre of the figure. Thirdly, on the right-hand side of the figure, the expanded stacks are shown which include the multi-frequency SAR images, the interchannel ratio layers, the selected texture layers at various window sizes and the DEM-derived information layers of the elevation, slope and aspect. Combining all the layers together came to 165 layers for the Cameroon dry site, 245 layers for the Cameroon wet season site, 219 layers for the Tanzania dry site and 134 layers for the Tanzania wet season site. The number of layers were not too much for classification, since it is straightforward to be dealt with by machine learning algorithms such as random forests.

Where possible, a virtual stack (.vrt) file format is preferred to a single image containing all the layers, for the sake of more efficient disk space allocation, with each set of site-season stack of images in the order of 10-30 Gb of data. Once all the data preparation steps had been completed, the layer stacks were then ready for sample selection (Section 3.4) in preparation for the land cover classification model development (Chapter Five).

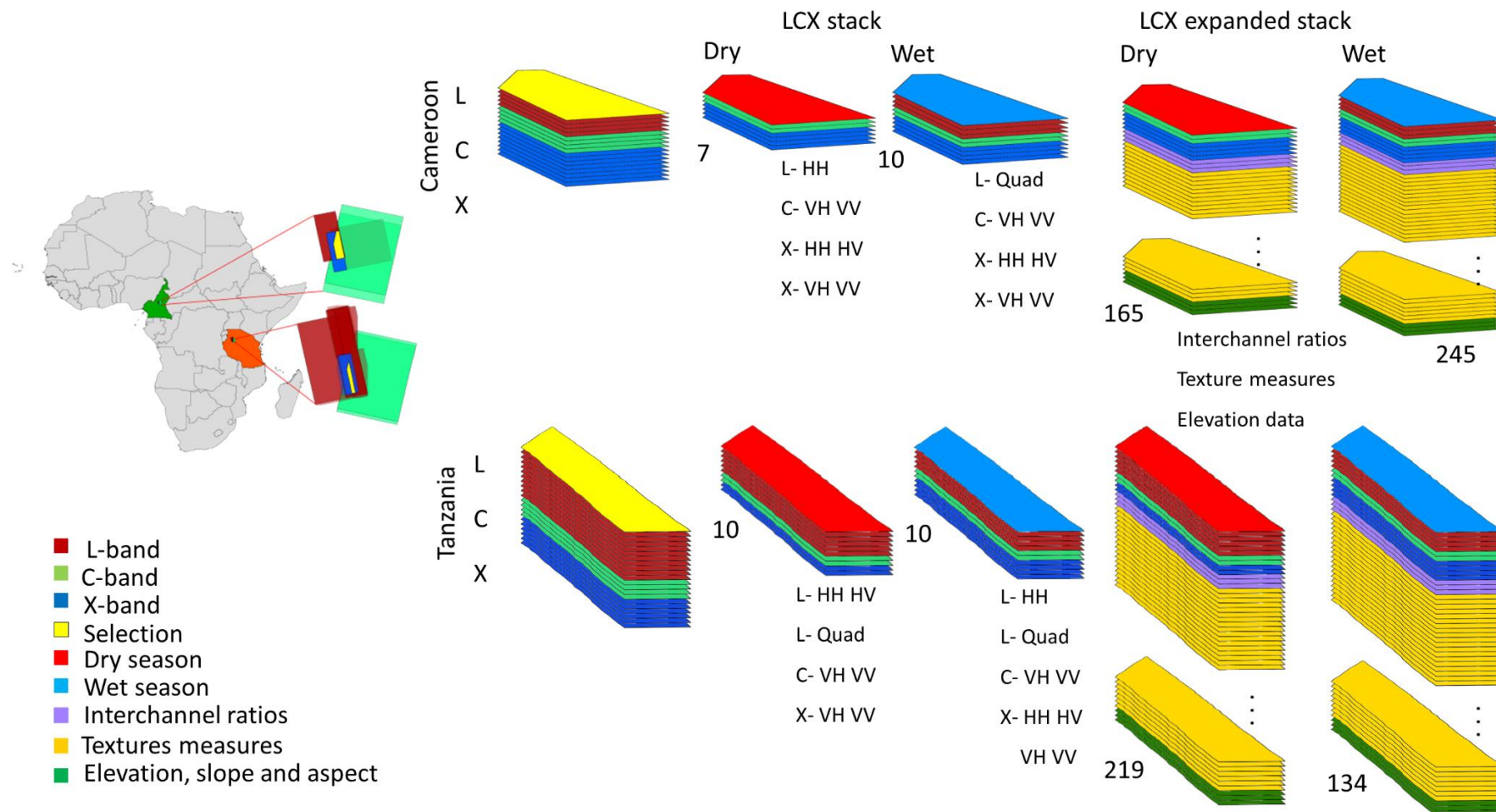


Figure 4.10 Layer stack creation and additional layer calculation of the Cameroon and Tanzania SAR images.

## **Chapter Five      Random forest and SVM classification of multi-frequency SAR over Cameroon and Tanzania study sites**

In this chapter, the classification accuracies of random forests and SVMs, using different frequency combinations and input-variable scenarios, are compared. This is undertaken for the Cameroon (forested) and Tanzania (semi-arid) sites in both the wet and dry seasons. As part of the random forest model, measures of variable importance are given, namely the MDA and MDG. These variable importance measures enable the selection of those that are most important for building a model, with a sufficient number of variables to give a high level of accuracy but without an excessive number of variables. This therefore reduces unnecessary complexity of the model.

Random forests is an ensemble classifier that has proved to be a highly efficient non-parametric classification technique (Breiman, 2001; Waske and Braun, 2009). SVMs are not an ensemble classifier, but still provide superior classification accuracies to traditional classifiers in most scenarios (Cortes and Vapnik, 1995; Waske and Benediktsson, 2007) (see Section 2.6).

As a results, the main algorithm investigated was first random forests and secondly SVMs. First, the different combinations of frequencies for different input scenarios were compared using random forests. In this process, all the available variables are compared, ranked and discussed. The available variables include: the backscatter coefficients for all frequencies and polarisations in  $\sigma^0$ -values (which are measured in decibels (dB) such as L-HH, C-VH etc.), the interchannel ratios and a selection of texture measures and elevation data (see Section 4.3). From all the available input variables, the 30 most important were chosen to build a random forest model for land cover classification for both the semi-arid and forested areas. The top 30 variables were chosen to train the random forest models with only a minor decline in overall accuracy compared to a random forest model using all available variables. As a second comparison, the top 30 variables were in used to train a SVM model and classification accuracies were

compared to that of the random forest model. The transferability of these models to an additional study site was tested in Chapter Six.

## 5.1 Introduction

The objective of this study was to compare the classification accuracies of land cover classification using single frequency L-, C-, X-band images, with dual-frequency combinations of LC, LX, CX-bands and three-frequency LCX-band images. This study addresses the question of: what is the added benefit of using two or three frequencies, compared to only one frequency? In order to achieve this objective a comparison was conducted on random forest classification models of the different band combinations (L-, C-, X-band, LC-, LX-, CX- and LCX-bands) for each of the available study site / season scenarios (i.e. for Cameroon dry season, Cameroon wet season, Tanzania dry season and Tanzania wet season). Apart from the standard radar backscatter coefficients, several additional input layers (calculated as set out in Section 4.3) are compared for each frequency combination. The list of options that are investigated for each frequency combination is:

- all layers (backscatter coefficients, interchannel ratios, texture measures and elevation data)
- backscatter coefficients only
- interchannel ratios only
- texture measures only
- elevation data only
- backscatter coefficients and interchannel ratios
- backscatter coefficients and texture measures
- backscatter coefficients and elevation data
- backscatter coefficients, interchannel ratios and texture measures

The methodology of comparing the different combinations is discussed further in Section 5.2.

## 5.2 Specific methodology for developing the classification algorithms

The methodology followed in this study was a standard supervised classification involving sample selection, training / test data split, model development on the training data and testing the model on the test data.

### 5.2.1 Input variable combination scenarios

The input layer stack of data was split into different combinations of input variables as set out in Table 5.1. Each of the frequency combinations of L-, C-, X-band, LC, LX, CX and LCX-bands were combined with a combination of the available input layers (backscatter coefficients, interchannel ratios, texture measures and elevation data). For each of these input variable scenarios, a random forest model was trained. The number of trees of each forest was kept to 500, which is the default for the random forest implementation. The default value of 500 was often used in other studies (Waske and Braun, 2009; Naidoo et al., 2012). The random forest model development was undertaken using the R statistical programming language (R-Core-Team, 2014). The standard implementation of random forests (Breiman and Cutler, 2001) is the *randomForest* package in R (Liaw and Wiener, 2002). The *randomForest* package was used to train the random forest models for each of the input variable scenarios. As part of the output from the *randomForest* package, the model, variable importance measures and class-specific and overall accuracies for each of the input variable scenarios are given. The different input variable scenarios are shown in Table 5.1, with the classification accuracies based on the test data for each scenario plotted in Section 5.3.1.

Table 5.1 List of scenarios investigated for model input comparison.

Name of scenario	Frequency	Included layers in scenario
L - dBite	L	$\sigma^0$ , i, t, e
C - dBite	C	$\sigma^0$ , i, t, e
X - dBite	X	$\sigma^0$ , i, t, e
LC - dBite	LC	$\sigma^0$ , i, t, e
LX - dBite	LX	$\sigma^0$ , i, t, e
CX - dBite	CX	$\sigma^0$ , i, t, e
LCX - dBite	LCX	$\sigma^0$ , i, t, e
L - dB	L	$\sigma^0$
C - dB	C	$\sigma^0$
X - dB	X	$\sigma^0$
LC - dB	LC	$\sigma^0$
LX - dB	LX	$\sigma^0$
CX - dB	CX	$\sigma^0$
LCX - dB	LCX	$\sigma^0$
L - i	L	i
C - i	C	i
X - i	X	i
LC - i	LC	i
LX - i	LX	i
CX - i	CX	i
LCX - i	LCX	i
L - t	L	t
C - t	C	t
X - t	X	t
LC - t	LC	t
LX - t	LX	t
CX - t	CX	t
LCX - t	LCX	t
LCX - e	none	e

Name of scenario	Frequency	Included layers in scenario
L - dBi	L	$\sigma^0$ , i
C - dBi	C	$\sigma^0$ , i
X - dBi	X	$\sigma^0$ , i
LC - dBi	LC	$\sigma^0$ , i
LX - dBi	LX	$\sigma^0$ , i
CX - dBi	CX	$\sigma^0$ , i
LCX - dBi	LCX	$\sigma^0$ , i
L - dBt	L	$\sigma^0$ , t
C - dBt	C	$\sigma^0$ , t
X - dBt	X	$\sigma^0$ , t
LC - dBt	LC	$\sigma^0$ , t
LX - dBt	LX	$\sigma^0$ , t
CX - dBt	CX	$\sigma^0$ , t
LCX - dBt	LCX	$\sigma^0$ , t
L - dBe	L	$\sigma^0$ , e
C - dBe	C	$\sigma^0$ , e
X - dBe	X	$\sigma^0$ , e
LC - dBe	LC	$\sigma^0$ , e
LX - dBe	LX	$\sigma^0$ , e
CX - dBe	CX	$\sigma^0$ , e
LCX - dBe	LCX	$\sigma^0$ , e
L - dBit	L	$\sigma^0$ , i, t
C - dBit	C	$\sigma^0$ , i, t
X - dBit	X	$\sigma^0$ , i, t
LC - dBit	LC	$\sigma^0$ , i, t
LX - dBit	LX	$\sigma^0$ , i, t
CX - dBit	CX	$\sigma^0$ , i, t
LCX - dBit	LCX	$\sigma^0$ , i, t

where  $\sigma^0$ : Backscatter coefficients  
i: Interchannel ratios  
t: Texture measures  
e: Elevation data

### 5.2.2 Variable importance measures from random forest modelling

As part of the standard implementation of the random forests, two measures of variable importance are given, namely the MDA and MDG. Random forests is introduced in Section 2.6.1. As each individual tree of the random forest is built, a third of the samples are left out, and the tree is only built on two thirds of the samples. Cross-validation, or a separate test set to get an unbiased estimate of the test set error, is not necessary in random forests, as the OOB error estimate can be used instead (Breiman, 2001; Breiman and Cutler, 2001). However, an additional test dataset was still used in this research to assess the classification accuracy of applying the model to data from the same image, but different polygon areas. The random forest MDA and MDG variable importance techniques are often used as feature selection techniques (Díaz-Uriarte and Alvarez de Andrés, 2006; Rodriguez-Galiano et al., 2012b; Gromski et al., 2014).

Feature selection are broadly grouped into three groups for filter methods, wrapper methods and embedded techniques (Saeys et al., 2007). Filter methods ranks the variables by assessing each variable individually (although some multi-variate filter methods have been developed), and include information gain and random k-nearest neighbour feature selection (Li et al., 2011). Wrapper methods wraps a search algorithm within a classification model, to evaluates specific subset of features, such as SVM Recursive Feature Elimination (Pal and Foody, 2010). Embedded feature selection methods are built into the classifier, such as the random forest feature selection techniques. Variants of the standard random forest variable importance measures are conditional variable importance for random forests (Strobl et al., 2008), area under the curve (AUC) based permutation variable importance measures (Janitza et al., 2013), and variable importance measure for random forests with missing values (Hapfelmeier et al., 2014). However, in this research the standard random forest variable importance measures MDA and MDG were used.

### 5.2.2.1 Mean decrease accuracy (MDA)

The MDA is based on the permutation importance of each variable. The permutation importance was given in Strobl et al. (2008), which is re-stated here. The MDA works as follows: when the values of predictor variable  $X_j$  is randomly permuted (re-ordered), and predictor variable  $X_j$  is important, then the prediction accuracy of the response variable in conjunction with the non-permuted variables will decrease as a result, as applied on the OOB sample. When values of  $X_j$  are randomly permuted and the prediction accuracy is not important, then the prediction accuracy on the OOB sample will not be affected. The MDA is then the difference between the prediction accuracy before permuting  $X_j$  and after permuting  $X_j$ , averaged over all the trees in the RF. The mathematical notation of this is given by:

Let  $\overline{OOB}^{(t)}$  be the OOB sample for a tree  $t$ , with  $t \in (1, \dots, ntree)$  and  $ntree$  the number of trees in the random forest model. The variable importance ( $VarImp$ ) of variable  $X_j$  in tree  $t$  is then

$$VarImp^{(t)}(X_j) = \frac{\sum_{i \in \overline{OOB}^{(t)}} I(y_i = \hat{y}_i^{(t)})}{|\overline{OOB}^{(t)}|} - \frac{\sum_{i \in \overline{OOB}^{(t)}} I(y_i = \hat{y}_{i, \kappa_j}^{(t)})}{|\overline{OOB}^{(t)}|} \quad \text{Equation 5-1}$$

where  $\hat{y}_i^{(t)} = f^{(t)}(x_i)$  is the predicted class for observation  $i$  before permutation, and  $\hat{y}_{i, \kappa_j}^{(t)} = f^{(t)}(x_{i, \kappa_j})$  is the predicted class for observation  $i$  after permutation of variable  $X_j$ . The raw variable importance for each variable is then calculated by averaging the variable importance over all the trees:

$$VarImp_{unscaled}(X_j) = \frac{\sum_{t=1}^{ntree} VarImp^{(t)}(x_j)}{ntree} \quad \text{Equation 5-2}$$



The raw variable importance is scaled by dividing it by its standard error (also called the z-score) in the standard implementation of RF. This was given by Strobl and Zeileis (2008) as:

$$VarImp_{scaled} = z = \frac{VarImp_{unscaled}(X_j)}{\hat{\sigma} / \sqrt{ntree}} \quad \text{Equation 5-3}$$

where  $\sigma$  is the standard deviation for each individual  $VarImp^{(t)}$  and the mean importance from all  $ntree$  trees in the random forest has standard error  $\hat{\sigma} / \sqrt{ntree}$ . However, since the  $VarImp_{unscaled}$  has been shown to have better statistical properties than  $VarImp_{scaled}$  (Strobl and Zeileis, 2008; Strobl et al., 2008), the unscaled version of the MDA variable importance measure was used instead of the scaled version in this study.

#### 5.2.2.2 Mean decrease Gini (MDG)

The MDG is the second measure of variable importance from the random forest output and is the average of the decrease in node impurity over all the trees in the RF. The decrease in node impurity is related to the Gini index, as is shown in Equation 5-7. Splitting a tree based on the Gini index is a standard way of splitting classification and regression trees (Breiman et al., 1984; Ishwaran, 2015). This can be defined mathematically as follows (Ishwaran, 2015):

Let the training data be  $T = (\mathbf{X}_i, Y_i)_{1 \leq i \leq n}$  where  $\mathbf{X}$  is the matrix of predictor variables and  $Y$  is the outcome class vector with class label  $Y \in \{1, \dots, C\}$  with  $C \geq 2$  possible classes. The Gini node impurity for node  $p$ ,  $\hat{\Gamma}(p)$ , is then defined as:

$$\hat{\Gamma}(p) = \sum_{c=1}^C \hat{\lambda}_c(p)(1 - \hat{\lambda}_c(p)) \quad \text{Equation 5-4}$$

with  $\hat{\lambda}_c(p)$  the class frequency for class  $c$  in a node  $p$ . Let  $s$  be the proposed split that splits node  $p$  into a left daughter node  $p_L \{X_i \leq s\}$  and a right daughter node,  $p_R \{X_i > s\}$ . The Gini node impurities for  $p_L$  and  $p_R$ ,  $\hat{\Gamma}(p_L)$  and  $\hat{\Gamma}(p_R)$ , are

$$\hat{\Gamma}(p_L) = \sum_{c=1}^C \hat{\lambda}_c(p_L) (1 - \hat{\lambda}_c(p_L)) \quad \text{Equation 5-5}$$

and

$$\hat{\Gamma}(p_R) = \sum_{c=1}^C \hat{\lambda}_c(p_R) (1 - \hat{\lambda}_c(p_R)) \quad \text{Equation 5-6}$$

respectively, with  $\hat{\lambda}_c(p_L)$  the class frequency for class  $c$  in  $p_L$  and with  $\hat{\lambda}_c(p_R)$  the class frequency for class  $c$  in  $p_R$ . The decrease in node impurity,  $\hat{\Gamma}(s, p)$ , for split  $s$  and node  $p$ , is then defined as

$$\begin{aligned} \hat{\Gamma}(s, p) &= \hat{\Gamma}(p) - [\hat{\phi}(p_L)\hat{\Gamma}(p_L) + \hat{\phi}(p_R)\hat{\Gamma}(p_R)] \\ &= \hat{\Gamma}(p) - \hat{G}(s, p) \end{aligned} \quad \text{Equation 5-7}$$

with  $\hat{\phi}(p_L)$  and  $\hat{\phi}(p_R)$  the proportions of observations in nodes  $p_L$  and  $p_R$  respectively.

The second part of Equation 5-7 is known as the Gini Index,  $\hat{G}(s, p)$ , with

$$\hat{G}(s, p) = \hat{\phi}(p_L)\hat{\Gamma}(p_L) + \hat{\phi}(p_R)\hat{\Gamma}(p_R) \quad \text{Equation 5-8}$$

Therefore, a good split-point will be where the split-point maximizes the decrease in node impurity, or equally minimizes the Gini Index.

Both the MDA, which is based on the permutation importance, and the MDG, which is based on the decrease in node impurity, are criteria for ranking the variables in terms of importance with relation to the outcome classes. Both of these measures were considered and compared when selecting the short-list of variables for the land cover classification model in this research.

## 5.3 Results

### 5.3.1 Random forest classification accuracy results

The classification accuracies based on the different input variable scenarios (from Table 5.1) are given in this section. Figure 5.1 shows the classification accuracies using all the available feature variables grouped by different frequency combinations (L-, C-, X-band, LC-, LX-, CX- and LCX-bands). For the Cameroon site, the dry season gives best results at 92% overall classification accuracy using either X or CX, and 91% for LCX. This compares to the Cameroon wet season with 88% overall classification accuracy using either bands LX or LCX. For the Tanzania site, the dry season gives the highest classification accuracy of 65% using LX and LCX. The Tanzania wet season has the highest classification accuracy of 52% using LC, and 51% using only L-band. Only L- and C-band data were available for the Tanzania wet season (see Sections 3.2 and 4.1).

The overall classification accuracies are much lower for all four site-season graphs using only the backscatter coefficients (Figure 5.2), compared to using the backscatter coefficients in conjunction with the interchannel ratios, texture measures and elevation data (Figure 5.1). This shows the benefit of using the additional derived feature layers, which added between 15-30% increase in overall classification accuracy for all four site-season combinations.

Using only backscatter coefficients, the Cameroon dry site has the highest overall classification accuracy of 66% (LCX), compared to the Cameroon wet site with 59% (LCX). The random forest model on the Tanzania dry images gives 49% (LCX) and the Tanzania wet images, only 32% (LC, no X-band available). In comparison, the Tanzania dry LC-band gave 48% overall accuracy. The dry season yields substantially higher overall classification accuracies to the wet season for both ecoregions. This can be explained by a higher soil moisture content and higher vegetation moisture content in the wet season images. This leads to higher SAR backscatter for all the land cover classes, and as a result higher misclassification between classes. In the dry season images, there is a greater

differentiation between dry bare soil and dense trees, with a greater backscatter spread for the vegetation classes between the two extremes.

### Random forest results using all variables ( $\sigma^0$ , i, t, e)

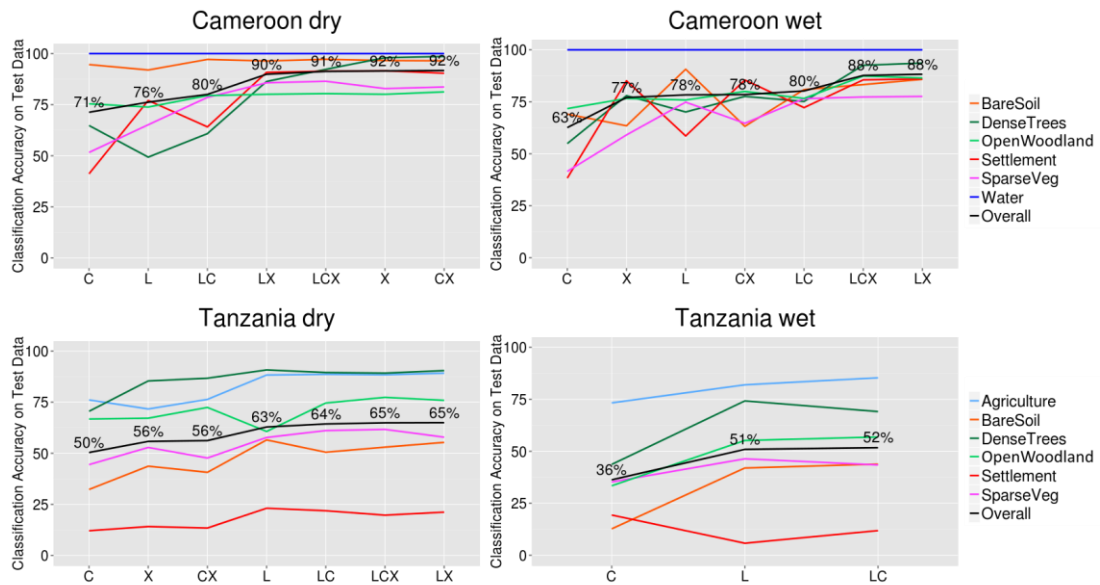


Figure 5.1 Random forest classification accuracies using all variables for different frequency combinations ( $\sigma^0$ ) for the Cameroon dry season study site (top left), the Cameroon wet season study site (top right), the Tanzania dry season study site (bottom left) and the Tanzania wet season study site (bottom right).

### Random forest results using backscatter coefficients only ( $\sigma^0$ )

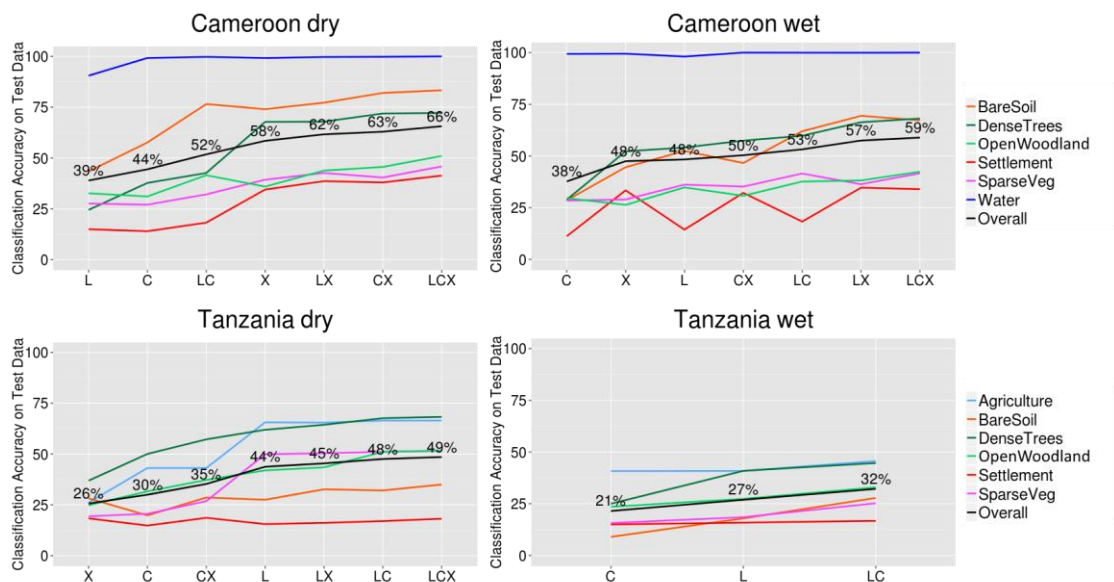


Figure 5.2 Random forest classification accuracies using backscatter coefficients only for different frequency combinations ( $\sigma^0$ ) for the Cameroon dry season study site (top left), the Cameroon wet season study site (top right), the Tanzania dry season study site (bottom left) and the Tanzania wet season study site (bottom right).

The results using only interchannel ratios (Figure 5.3) are slightly lower than when only the backscatter coefficients are used (Figure 5.2). The highest overall classification accuracies are 63%, 58%, 47% and 21% for the Cameroon dry, Cameroon wet, Tanzania dry and Tanzania wet site respectively. To enable the calculation of the interchannel ratios, either a dual-polarized or a quad-polarized image is required. Only the Cameroon wet season site had a dual-polarized or quad-polarized image for L-band, C-band and X-band. The Cameroon dry season site does not have a dual-polarized L-band and Tanzania dry season study site does not have a dual polarized X-band image. The Tanzania wet season study site only had a dual-polarized C-band image with no dual- or quad-polarized L- or X-band images available.

The highest overall classification accuracies using only texture measures are 92%, 85%, 64% and 49% for the Cameroon dry season, Cameroon wet season, Tanzania dry season and Tanzania wet season respectively (Figure 5.4). Using only the texture measures (Figure 5.9) was only marginally lower (or the same for the Cameroon dry season), than the random forest models with all the variable groups (Figure 5.1). This shows that the texture measures variables are more predictive of the land cover classes than the backscatter coefficients, interchannel ratio layers and elevation data. This entails that the texture measures was a greater differentiator of the land cover classes than the other variables considered.

Using only the elevation data results in a relatively high classification accuracy for the Cameroon study site, at 56% and 57% overall classification accuracy for the dry and wet seasons respectively. For the Tanzania site the result was substantially lower, at 28% and 27% for the dry and wet seasons respectively (Figure 5.5). The combinations of input variables that don't include texture measures give overall accuracies similar to the backscatter coefficients only (Figure 5.2). They are backscatter coefficients and interchannel ratios (Figure 5.6) and backscatter coefficients and elevation data (Figure 5.7). The combinations of input variables that include texture measures, give overall accuracies similar to the scenario that only includes texture measures (Figure 5.4). These

combinations are backscatter coefficients and texture measures (Figure 5.8) and backscatter coefficients, interchannel ratios and texture measures (Figure 5.9).

### Random forest results using interchannel ratios only

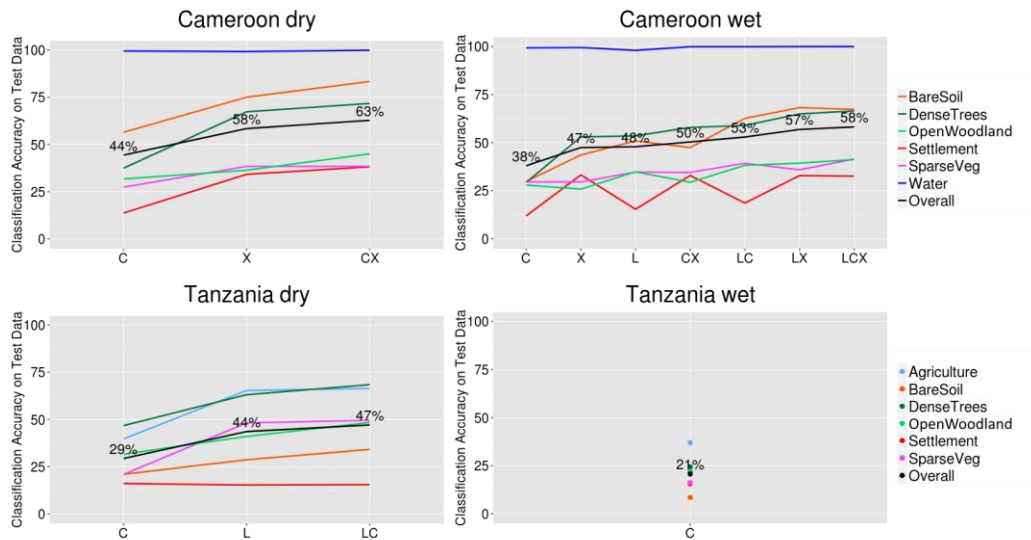


Figure 5.3 Random forest classification accuracies using only interchannel ratios for different frequency combinations ( $\sigma^0$ ) for the Cameroon dry study site (top left), the Cameroon wet study site (top right), the Tanzania dry study site (bottom left) and the Tanzania wet study site (bottom right).

### Random forest results using texture measures only

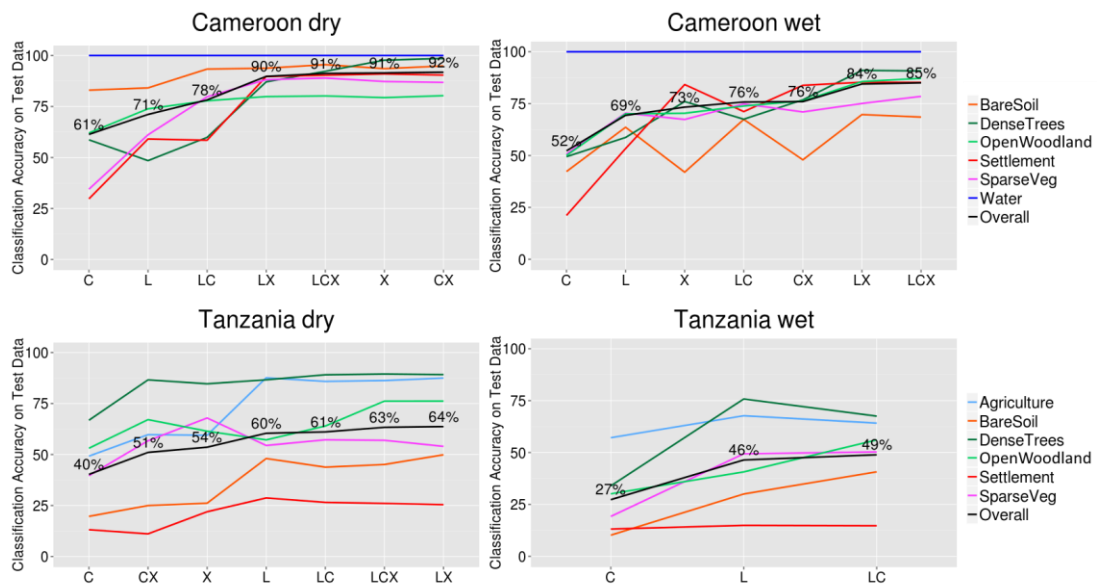


Figure 5.4 Random forest classification accuracies using texture measures only for different frequency combinations ( $\sigma^0$ ) for the Cameroon dry study site (top left), the Cameroon wet study site (top right), the Tanzania dry study site (bottom left) and the Tanzania wet study site (bottom right).

### Random forest results using elevation data only

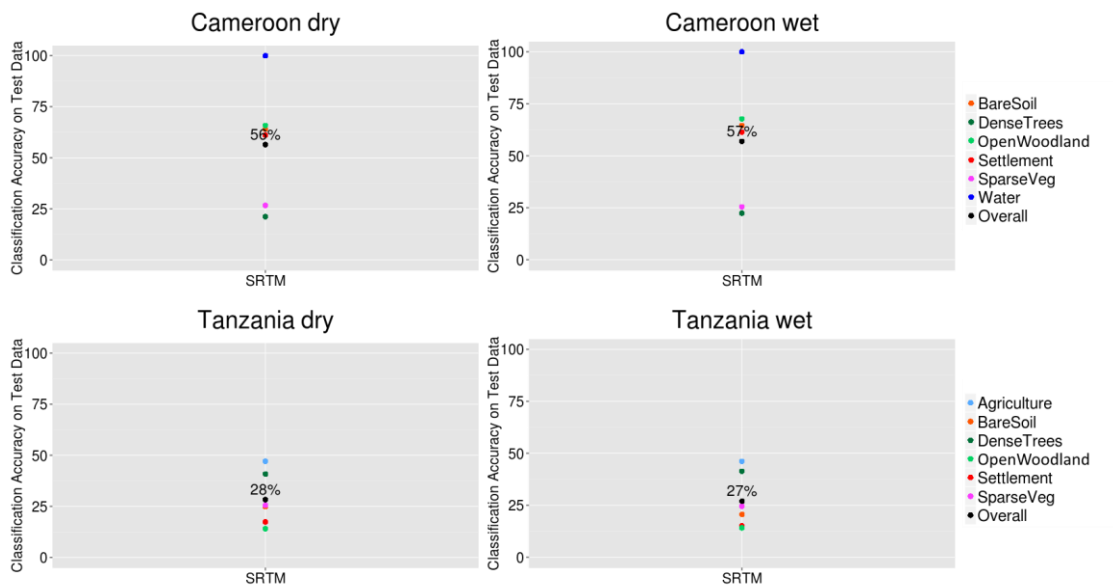


Figure 5.5 Random forest classification accuracies using only elevation data from SRTM (elevation, slope and aspect) for the Cameroon dry study site (top left), the Cameroon wet study site (top right), the Tanzania dry study site (bottom left) and the Tanzania wet study site (bottom right).

### Random forest results using backscatter coefficients and interchannel ratios

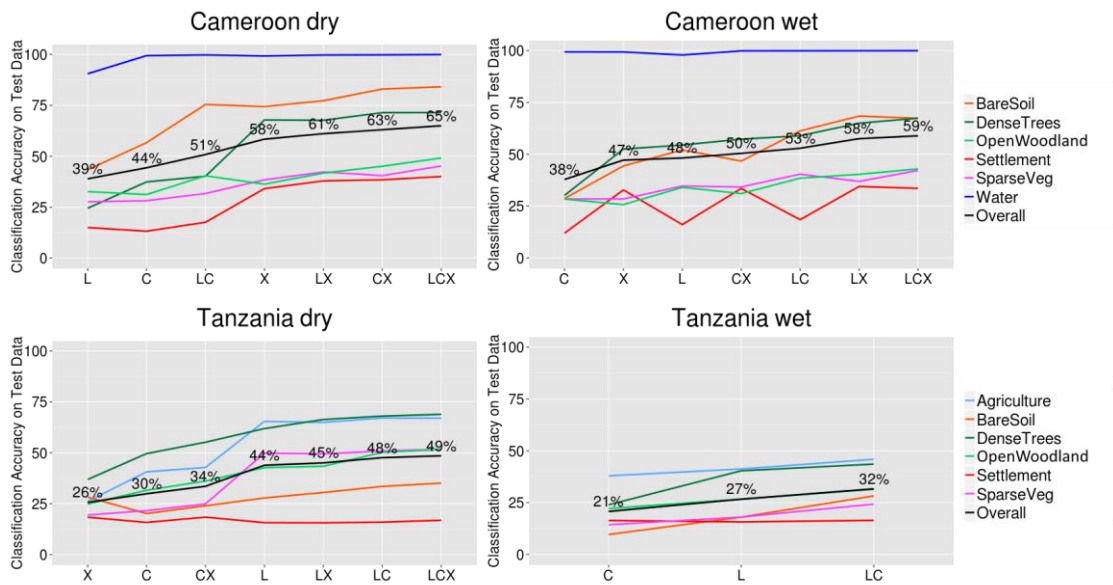


Figure 5.6 Random forest classification accuracies using backscatter coefficients and interchannel ratios for different frequency combinations ( $\sigma^0$ ) for the Cameroon dry study site (top left), the Cameroon wet study site (top right), the Tanzania dry study site (bottom left) and the Tanzania wet study site (bottom right).

## Random forest results using backscatter coefficients and elevation data

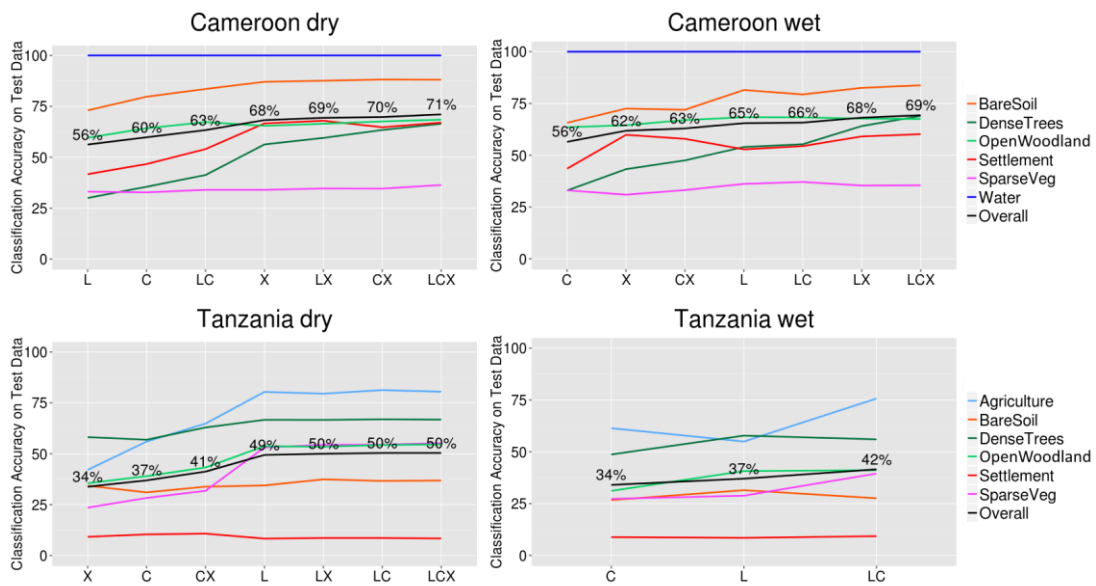


Figure 5.7 Random forest classification accuracies using backscatter coefficients and elevation data for different frequency combinations ( $\sigma^0$ ) for the Cameroon dry study site (top left), the Cameroon wet study site (top right), the Tanzania dry study site (bottom left) and the Tanzania wet study site (bottom right).

## Random forest results using backscatter coefficients and texture measures

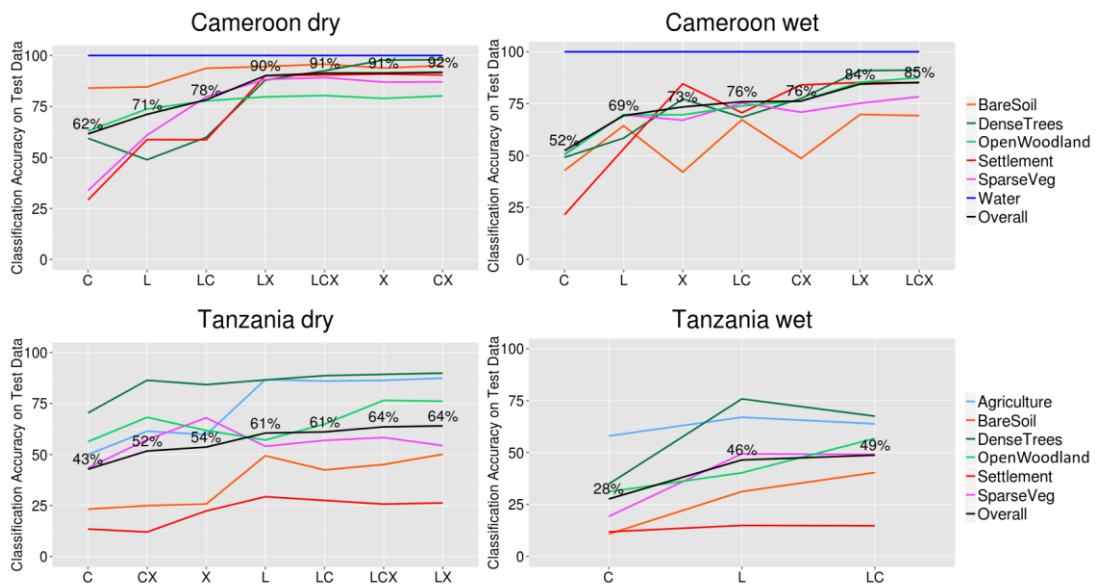


Figure 5.8 Random forest classification accuracies using backscatter coefficients and texture measures for different frequency combinations ( $\sigma^0$ ) for the Cameroon dry study site (top left), the Cameroon wet study site (top right), the Tanzania dry study site (bottom left) and the Tanzania wet study site (bottom right).



## Random forest results using backscatter coefficients, interchannel ratios and texture measures

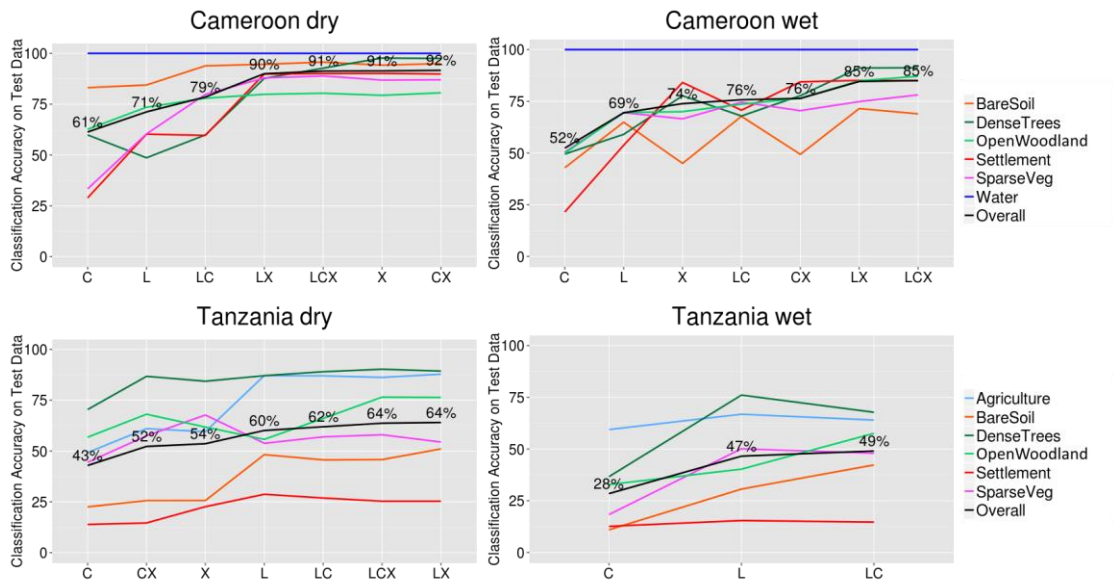


Figure 5.9 Random forest classification accuracies using backscatter coefficients, interchannel calculations and texture measures for different frequency combinations ( $\sigma^0$ ) for the Cameroon dry study site (top left), the Cameroon wet study site (top right), the Tanzania dry study site (bottom left) and the Tanzania wet study site (bottom right).

### 5.3.2 Selecting variables based on variable importance

#### 5.3.2.1 Variable importance of all available variables

As mentioned and described in Section 5.2.2, there are two measures of variable importance as part of the random forest model output, namely the MDA and MDG. Both of these measures are considered and compared against each other to select a short-list of variables for a refined model for the Cameroon dry and wet season, Tanzania dry and wet season sites. The unscaled version of the MDA was used to determine the variable importance (Strobl and Zeileis, 2008; Strobl et al., 2008).

The order of variable importance based on MDA and MDG of all the available variables for the Cameroon dry and wet seasons and the Tanzania dry and wet seasons are shown in Figure 5.10. Figure 5.10 groups the variables based on variable type, namely

backscatter coefficients, interchannel ratios, texture measures and elevation data. The backscatter coefficients and interchannel calculations are consistently grouped as the variables with a lower variable importance based on both MDA and MDG. The texture measures and elevation data stand out as being the variables with the highest variable importance based on both MDA and MDG. The variable importance of each individual group is investigated further. Looking more closely at both the Cameroon and Tanzania site, revealed a relationship between elevation and land cover type. It was noted that for the Cameroon site, the dense trees class are mostly in the valleys, whereas the open woodland land cover class are higher on the hills. Sparse vegetation and bare soil classes are located closer to the riverbed areas. For the Tanzania site, the agricultural areas seems to coincide with the valleys, with the dense trees and open woodland classes higher on the hills.

Considering the variable importance based only on the backscatter coefficients, the C-VV channel performs the best overall across all the study sites and seasons (Figure 5.11). The X-VH and X-HV channels are either the lowest, or towards the lower half, of the variable importance measures. The order of variable importance for the other channels varies depending on each site/season as well as the variable importance measure being considered, i.e. either MDA or MDG. These graphs are of interest to know which frequency and polarisation channels give the highest contribution towards the classification accuracy of the land cover classes for each site / season. It is worthwhile to mention here that different classification accuracies can be expected under different rainfall amount conditions. Higher amounts of rainfall lead to higher soil moisture and higher vegetation water content, which results in increased backscatter returns. An indication of whether the models are still suitable or not applicable at all, are addressed in Section 6.3.1.1 and 6.3.2.1, where the inter-season transferability of the Cameroon and Tanzania sites are investigated respectively.

Assessing the interchannel ratios individually, the C-VV plus C-VH, and C-VV minus C-VH interchannel ratios have the highest variable importance across all the site-season scenarios (Figure 5.12).

Analysis of the texture measures revealed that the mean texture measure are the most predictive variables, followed by entropy, contrast, variance and correlation. The texture measures for the top frequency-polarisation band combinations are X-HH, C-VV and L-HH for Cameroon dry (Figure 5.13); X-HH, C-VV and L-HV for Cameroon wet (Figure 5.14); X-VH, C-VV and L-HV for Tanzania dry (Figure 5.14); and X-VH, C-VV and L-HV for Tanzania wet (Figure 5.15). For each of the texture measures, the larger window sizes lead to a higher ranking of variable importance, which indicate higher classification accuracies.

### All variables based on MDA and MDG grouped by variable type

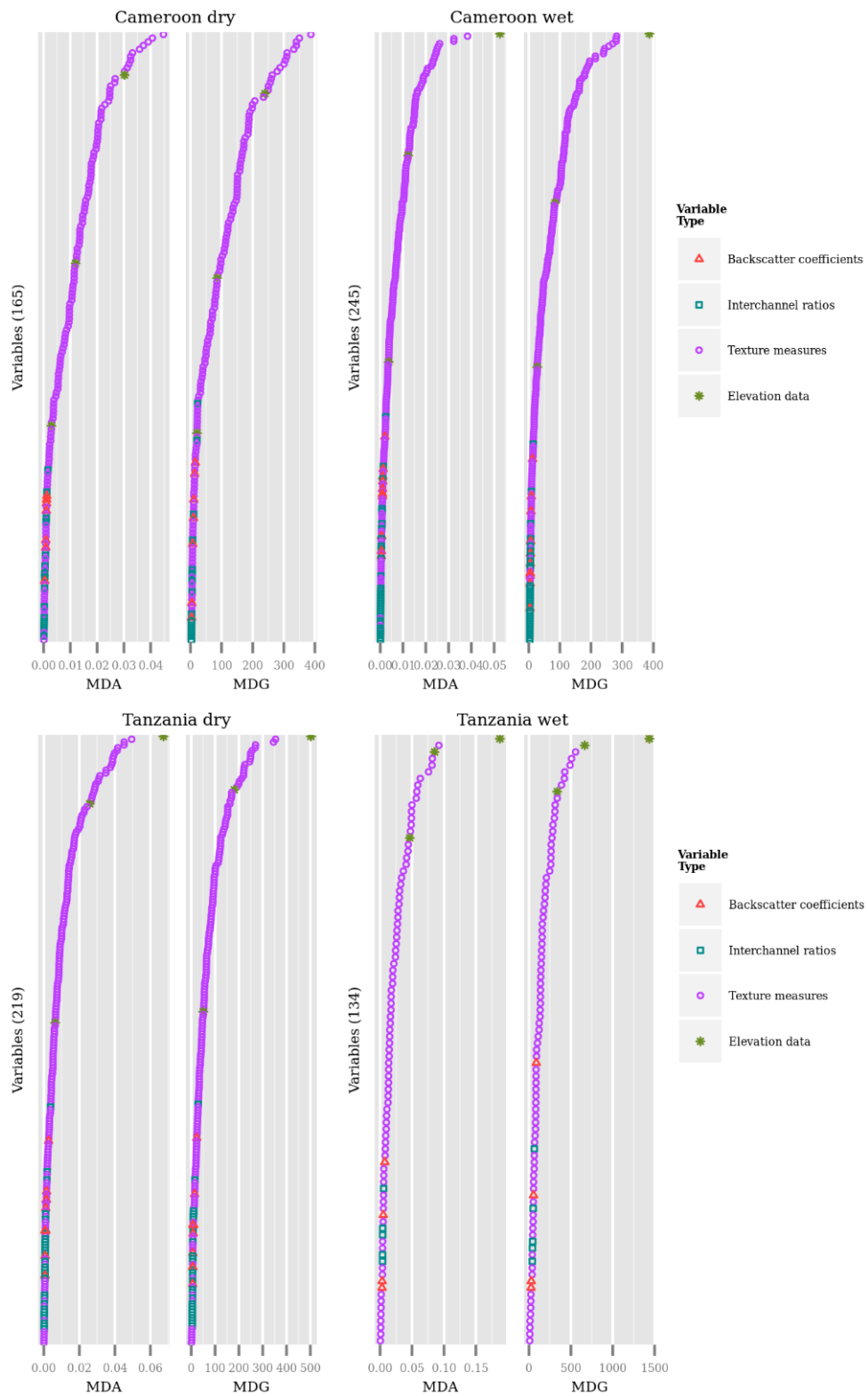


Figure 5.10 The MDA and MDG variable importance measures for Cameroon dry season (top left), Cameroon wet season (top right), Tanzania dry season (bottom left) and Tanzania wet season (bottom right), grouped by variable type (backscatter coefficients, interchannel ratios, texture measures and elevation data). The number of variables for each grouped are shown in brackets to the left of each group.

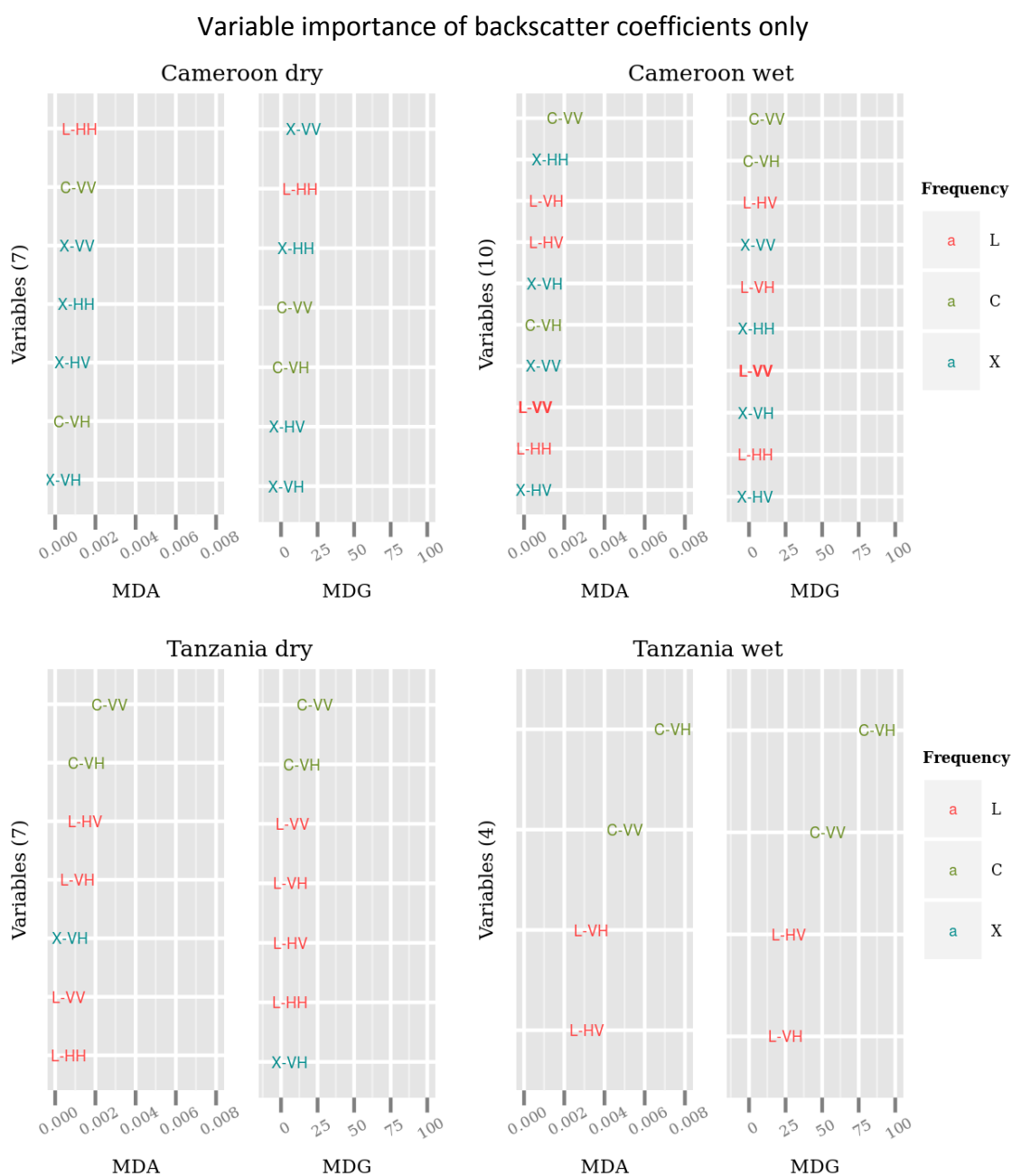


Figure 5.11 The order of variable importance based on MDA and MDG using only backscatter coefficients, for Cameroon dry season (top left), Cameroon wet season (top right), Tanzania dry season (bottom left) and Tanzania wet season study sites (bottom right). The number of variables for each grouped are shown in brackets to the left of each group.

## Variable importance of interchannel ratios only

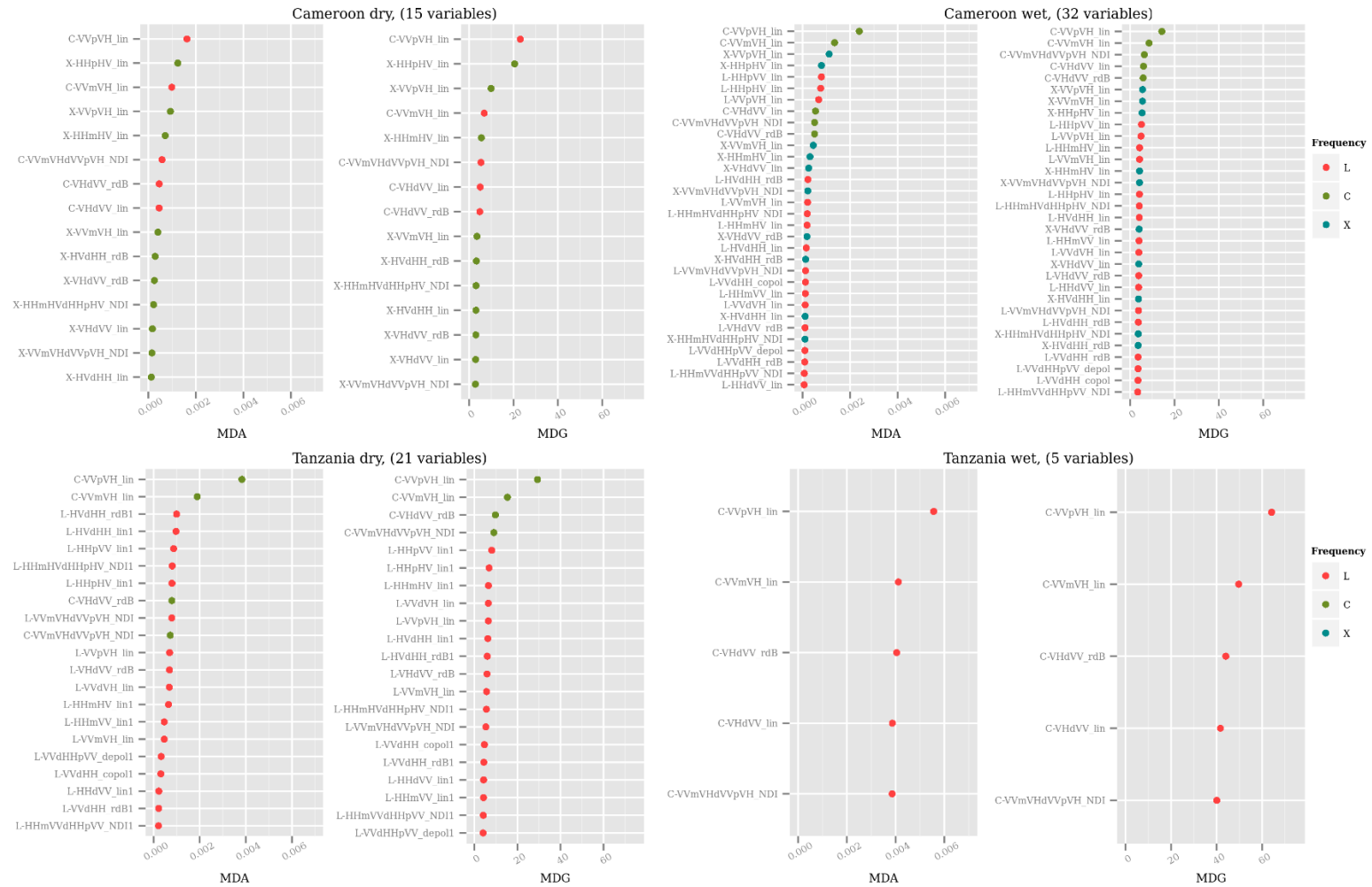


Figure 5.12 Variable importance (MDA and MDG) of only interchannel ratios, for each of the site / season scenarios.

## Variable importance of texture measures – Cameroon dry

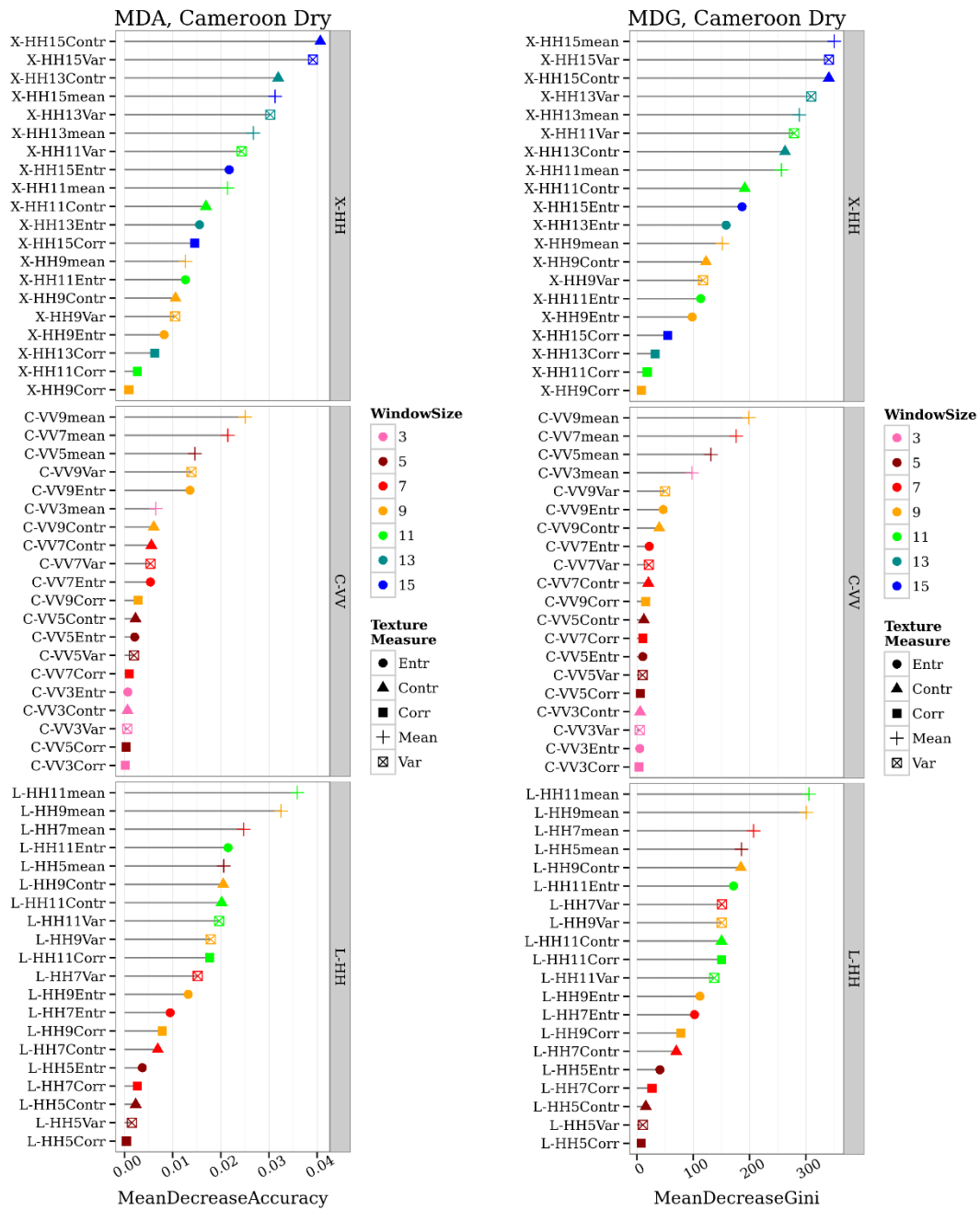


Figure 5.13 MDA and MDG variable importance of texture measures, for X-HH, C-VV and L-HH frequency-polarisation combinations for the Cameroon dry study site.

## Variable importance of texture measures – Cameroon wet

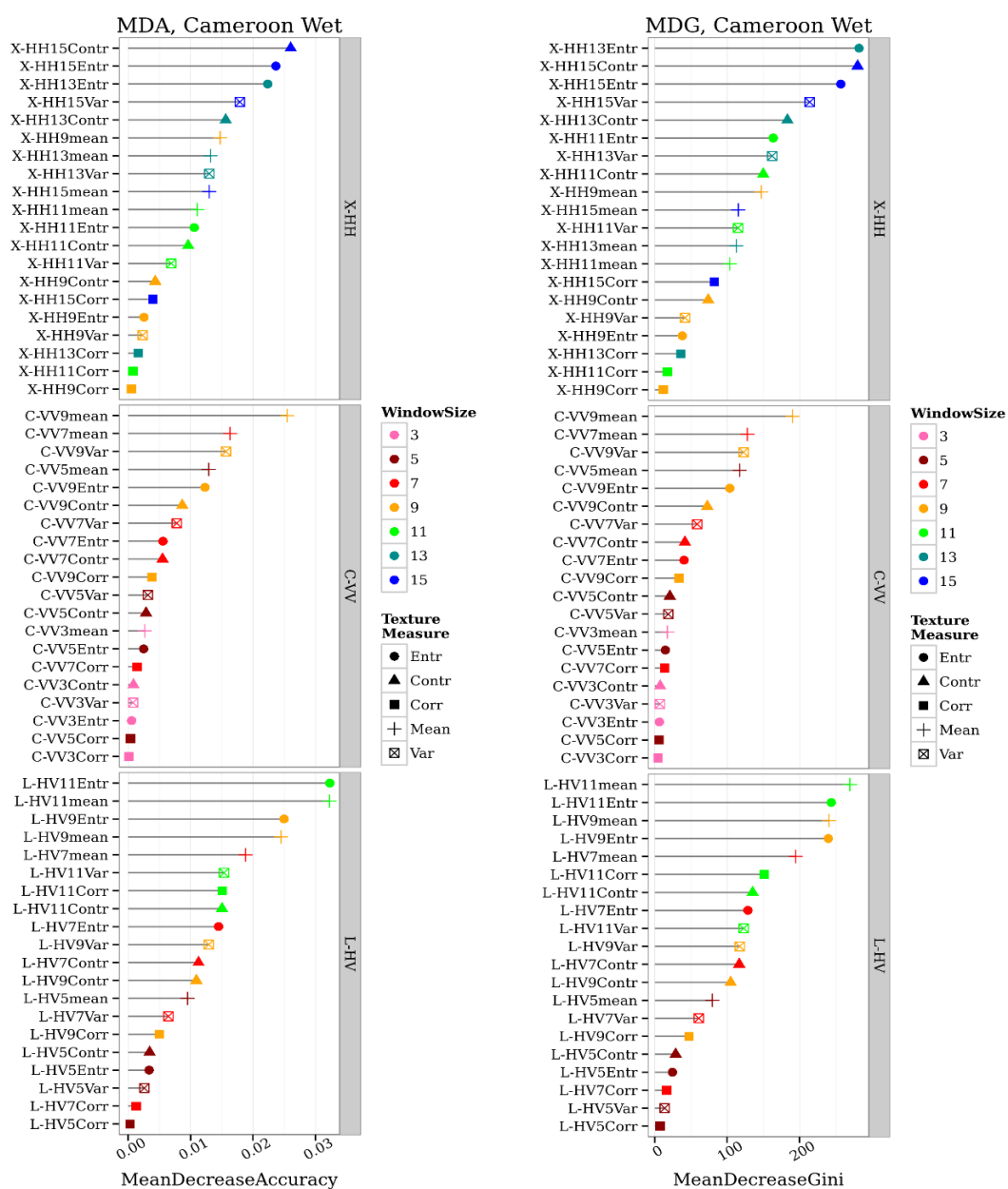


Figure 5.14 MDA and MDG variable importance of texture measures, for X-HH, C-VV and L-HV frequency-polarisation combinations for the Cameroon wet study site.



## Variable importance of texture measures – Tanzania dry

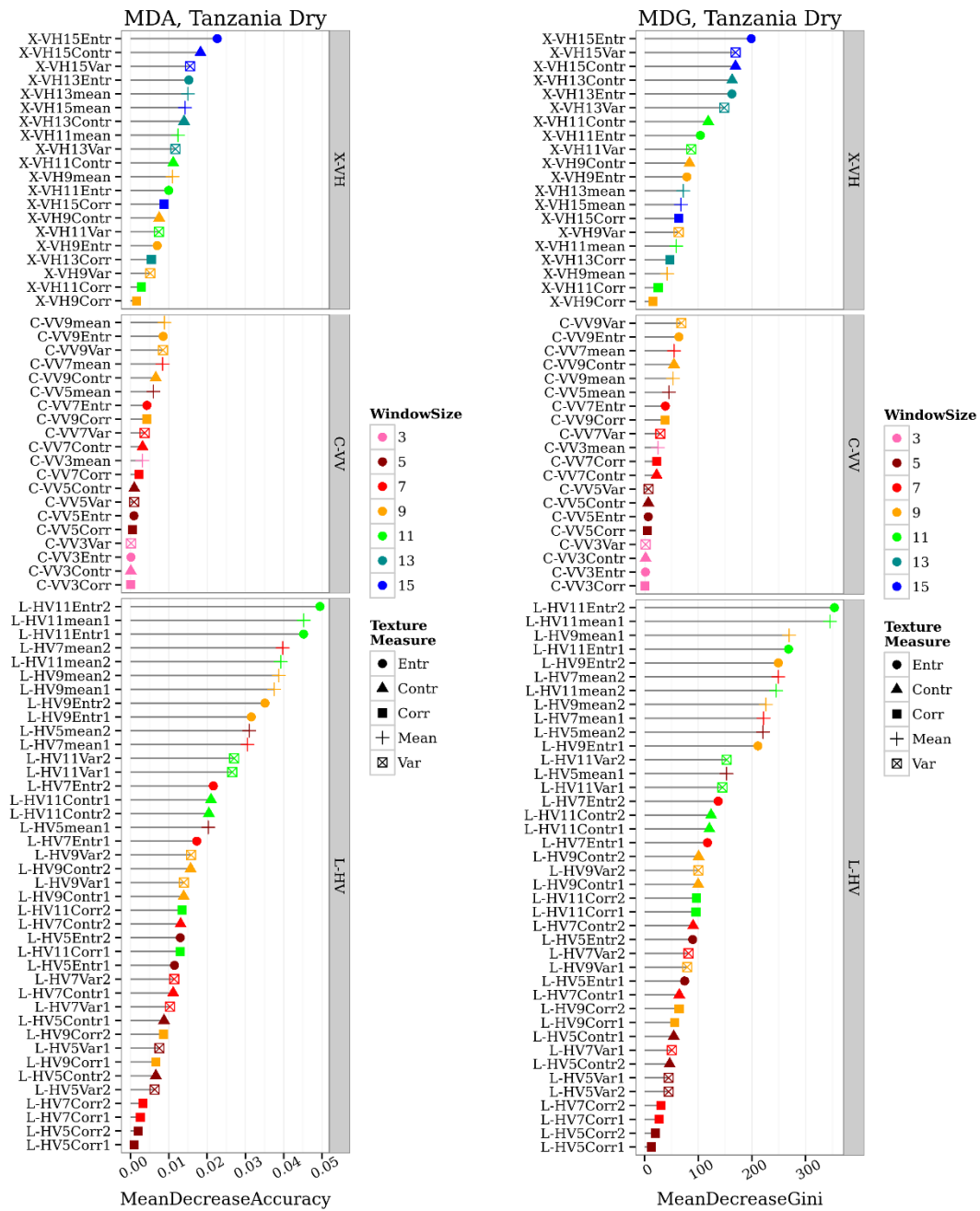


Figure 5.15 MDA and MDG variable importance of texture measures, for X-VH, C-VV and L-HV frequency-polarisation combinations for the Tanzania dry study site.

## Variable importance of texture measures – Tanzania wet

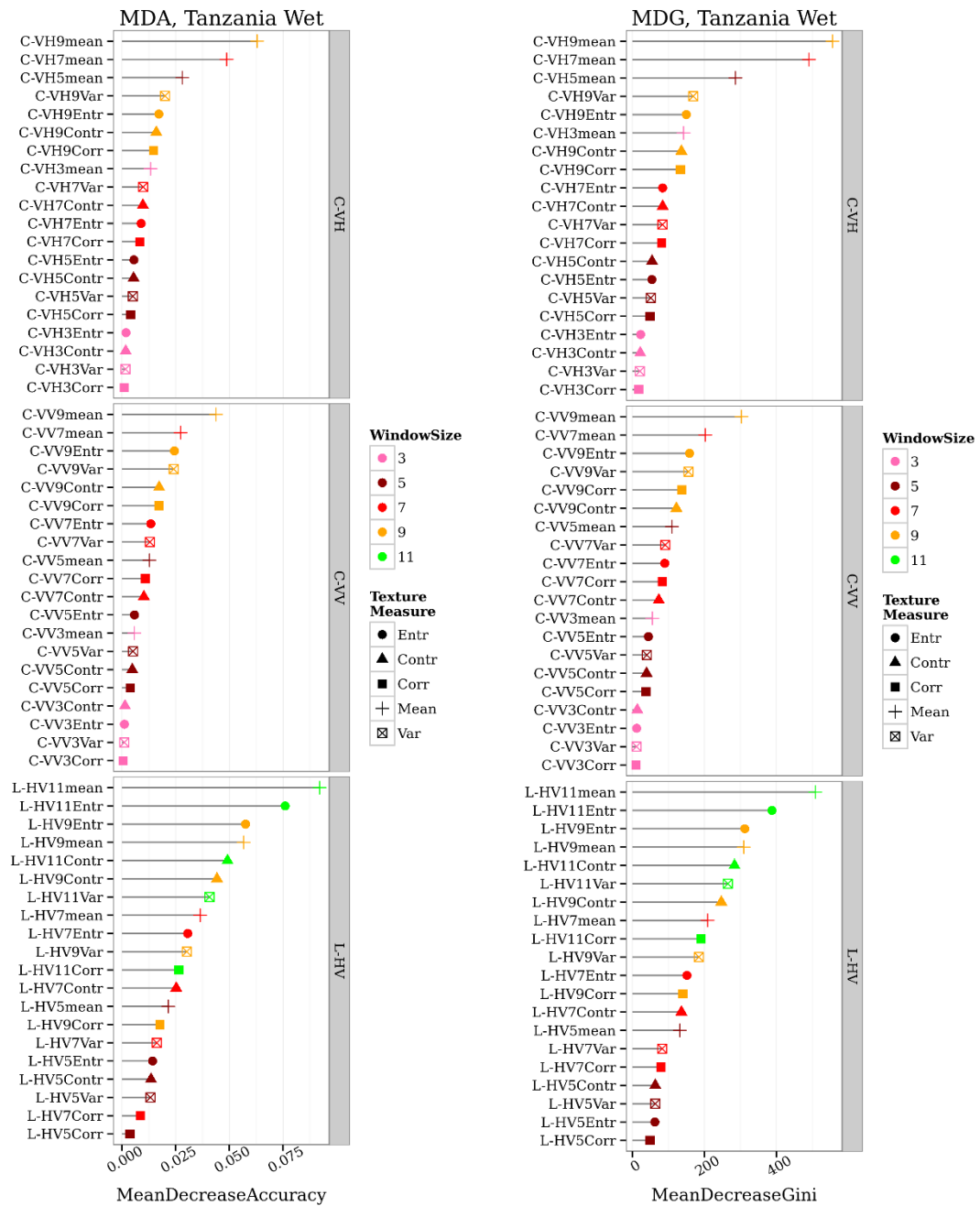


Figure 5.16 MDA and MDG variable importance of texture measures, for X-VH, C-VH and L-HV frequency-polarisation combinations for the Tanzania wet study site.

For the Cameroon dry site in Figure 5.13, the X-HV entropy (15 and 13 window sizes) are the top variables selected by the MDA measure and the X-HV entropy (15) and X-HH variance (15) are the top variables selected by the MDG measure. For the L-band and C-band, the neighbourhood mean layers (see Section 4.3.2 for calculation) give the highest classification accuracies for the L- and C- frequencies.

For the Cameroon wet study site, as shown in Figure 5.14, the X-HH band entropy, contrast and variance layers, at 13 and 15 window sizes, have the highest variable importance. For C-VV the neighbourhood mean texture measures, at window sizes 9 and 7, give the highest variable importance. The correlation measures have the lowest variable importance for the X, C and L frequencies.

For the Tanzania dry and Tanzania wet study sites (Figure 5.15 and Figure 5.16) the L-HV texture measures give the highest variable importance. For the Tanzania wet site the C-VH neighbourhood mean texture measures, at 9 and 7 window sizes, give the highest variable importance, whereas for the Tanzania dry site the C-band texture measures do not give very high variable importance.

Figure 5.17 shows the MDA and MDG variable importance values for the elevation data for each site, with the variable importance order shown for elevation, slope and aspect for each site. For all the study sites, the elevation above sea level gives very high variable importance, followed by the aspect and then the slope.

Since the polarimetric decompositions of fully polarimetric data were excluded from this research, the direct relationship between the physical scattering mechanisms of volume scattering, surface scattering and double bounce, were not directly related to the texture measures in the final model. Polarimetric decompositions such as entropy / alpha / anisotropy model is often used to give a measure of the physical scattering mechanisms (Hajnsek et al., 2003).

### Variable importance of elevation data only

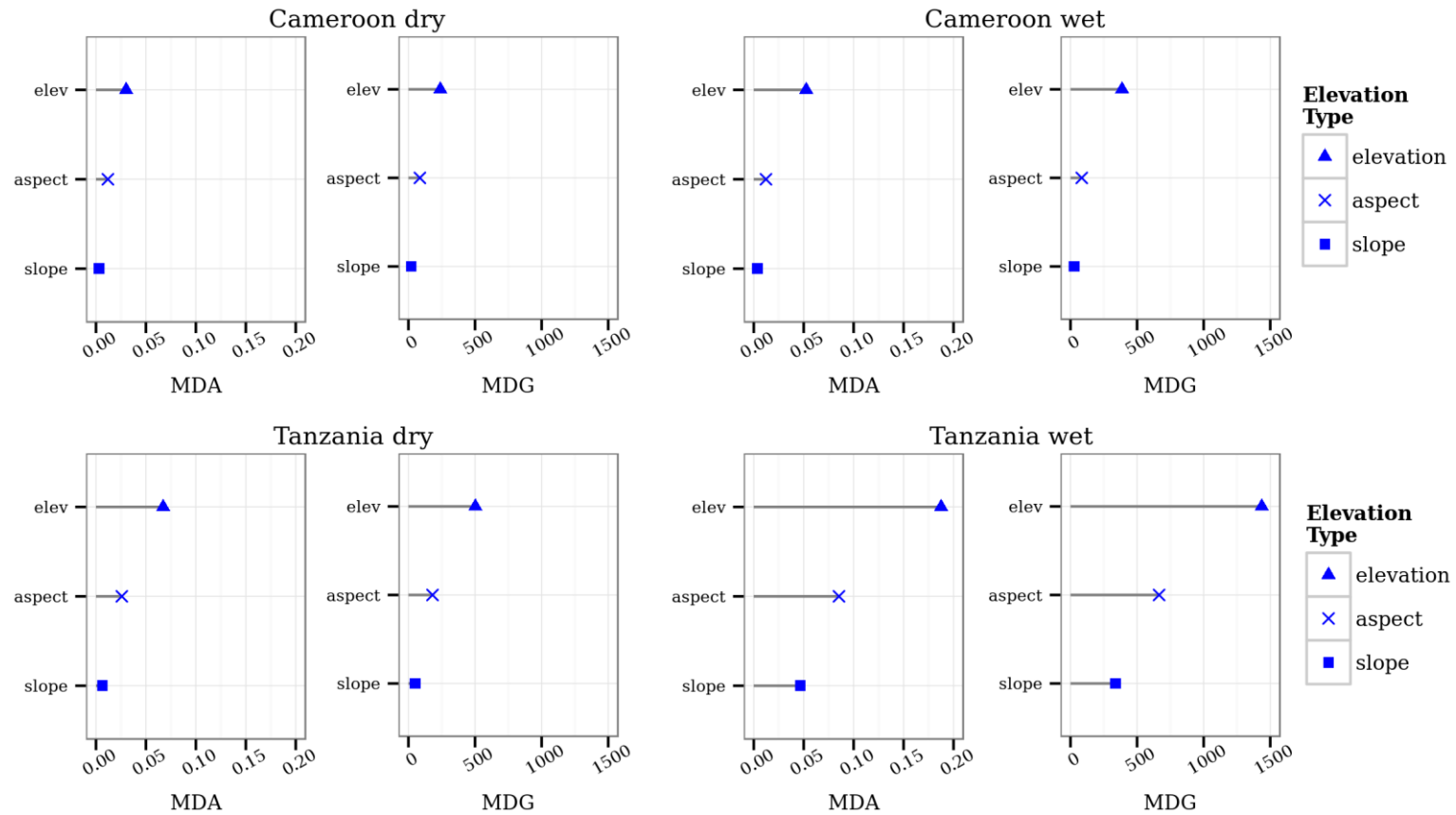


Figure 5.17 MDA and MDG variable importance for elevation, slope and aspect for the Cameroon dry (top left), Cameroon wet (top right), Tanzania dry (bottom left) and Tanzania wet (bottom right) study sites.

### 5.3.2.2 Variable importance of the top 30 variables

From all the available variables, which included the backscatter coefficients, interchannel calculations, texture measures and elevation data, the top 30 variables were chosen, based on both the MDA and MDG. The random forest models based on the top 30 variables resulted in only a small decline in overall accuracy compared to a random forest model using all available variables. The classification accuracies based on the top 30 variables is shown in Section 5.3.3, and compared can be compared to the random forest models based on all the variables in Section 5.3.1.

The top 30 variables for the Cameroon dry study site are shown in Figure 5.18. The top 30 variables based on MDA and MDG are shown in the top half of the graph. The bottom half shows the same top 30 variables, but grouped by the frequency-polarisation group. This was similarly undertaken for the Cameroon wet season study site in Figure 5.19, the Tanzania dry season study site in Figure 5.20 and for the Tanzania wet season study site in Figure 5.21.

In Figure 5.10 and Figure 5.17 it was shown that the elevation data variables have high variable importance values. However, to be able to use the models on additional study sites, which are at a slightly different elevation above sea-level, it was decided to exclude the elevation data from the top 30 variable selection. Therefore, after excluding the elevation data, the top 30 variables are all texture measures, for all four site-season combinations.

For the Cameroon dry season site, the X-HH and X-HV texture layers have the highest variable importance (Figure 5.18). For the Cameroon wet season site, the most important variables in the top 30 list are spread evenly between all the frequency combinations of X-HH, X-HV, C-VV, L-HV and L-VH (Figure 5.19). The top 30 variables for the Tanzania dry season site are a combination of X-VH and all the available L-band channels, namely L-HH, L-HV, L-VH and L-VV. For the Tanzania dry season site, only texture measure from L-band and X-band images were selected, with no C-band image

texture measures in the top 30 variables (Figure 5.20). For the Tanzania wet season site, where no X-band data were available, the top 30 variables are from the texture layers calculated from C-VH, C-VV, L-HV and L-VH (Figure 5.21).

The two combinations of top 30 variables, based on MDA and MDG, are used to train random forest models in the following section for each site (Section 5.3.3).

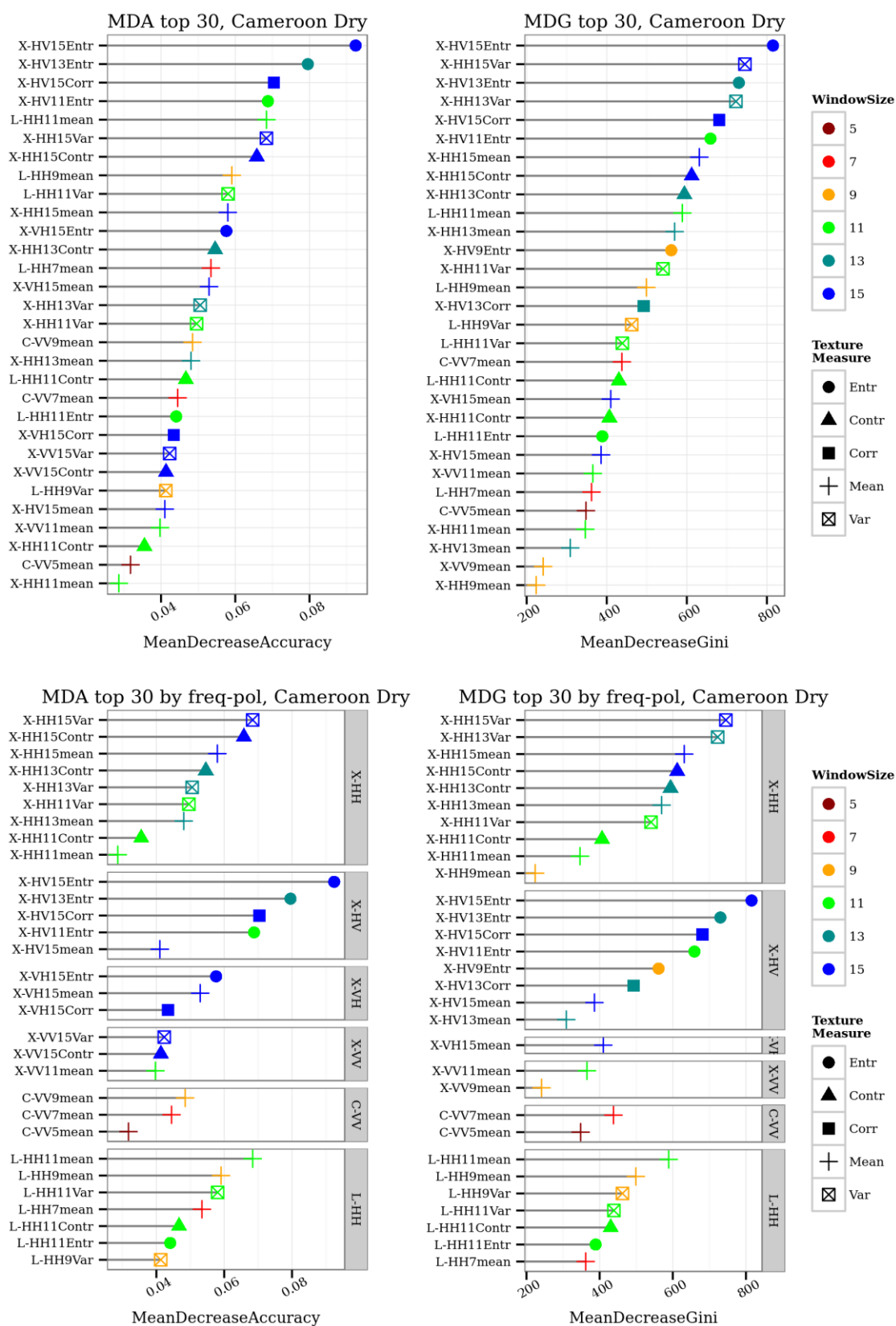


Figure 5.18 Top 30 variables based on MDA and MDG for the CD site random forest model. The top two graphs show variable 1-30 in ranking order, whereas the bottom two graphs are grouped by the original frequency-polarisation channels.

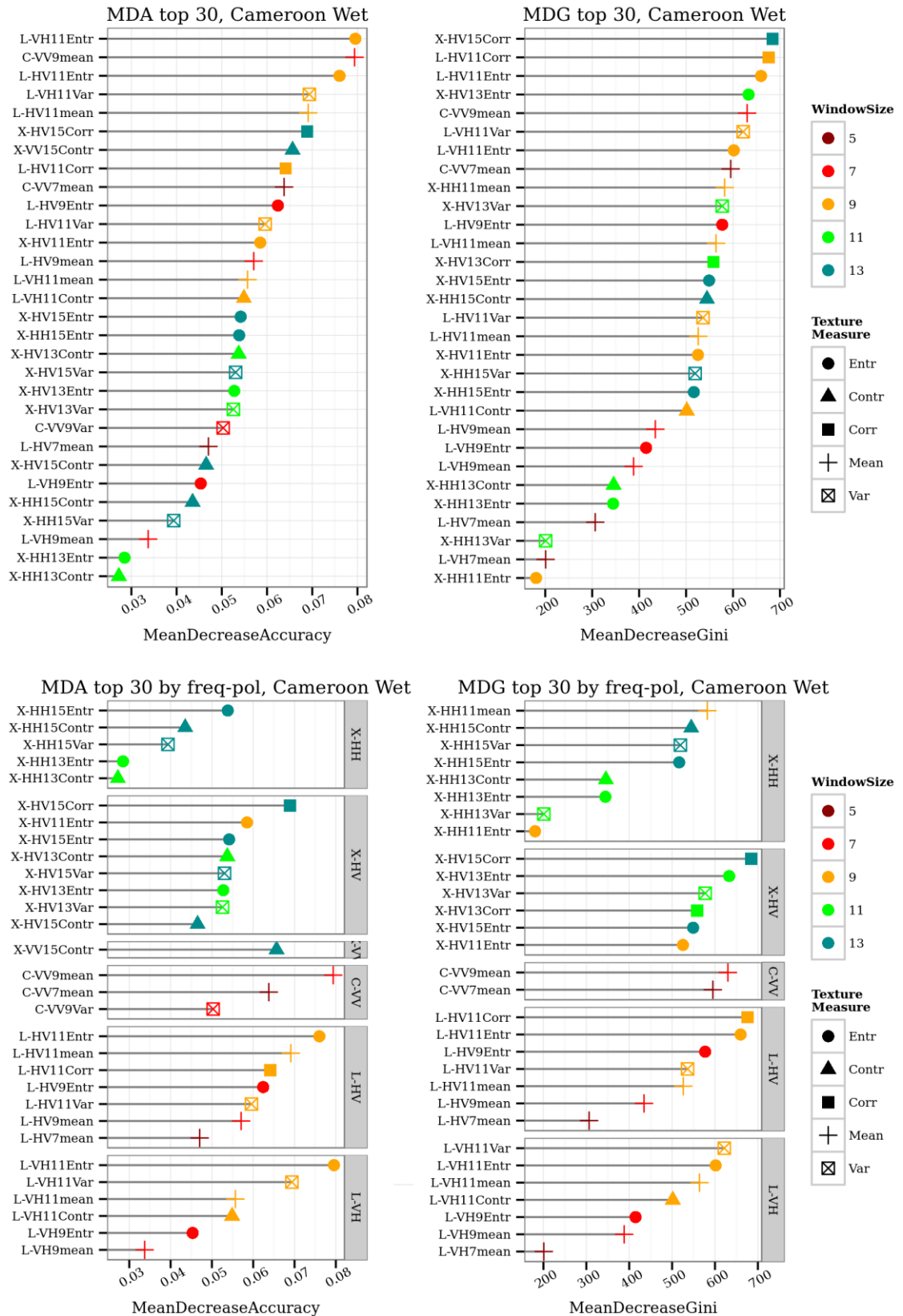


Figure 5.19 Top 30 variables based on MDA and MDG for the CW site random forest model. The top two graphs show variable 1-30 in ranking order, whereas the bottom two graphs are grouped by the original frequency-polarisation channels.



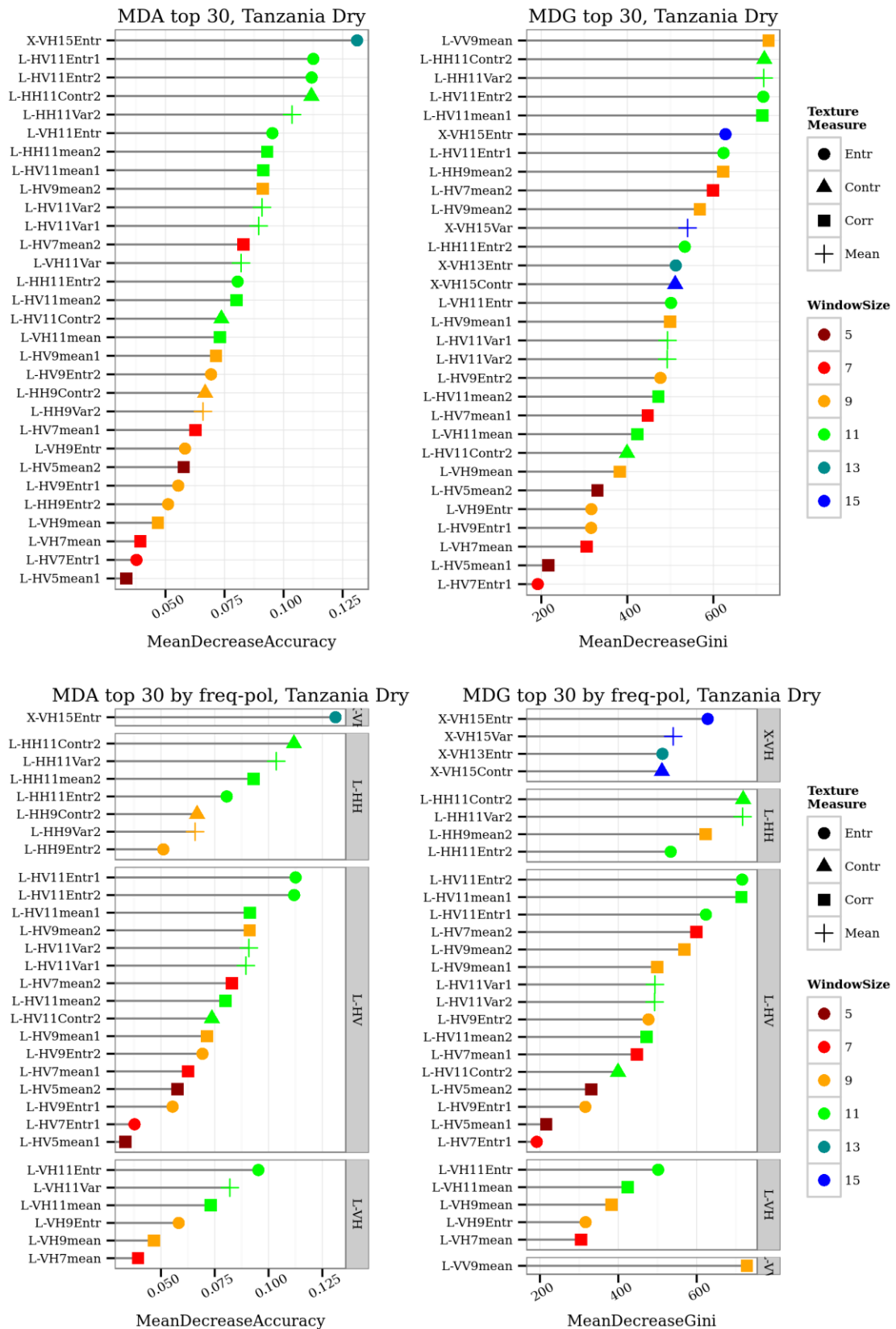


Figure 5.20 Top 30 variables based on MDA and MDG for the TD site random forest model. The top two graphs show variable 1-30 in ranking order, whereas the bottom two graphs are grouped by the original frequency-polarisation channels.

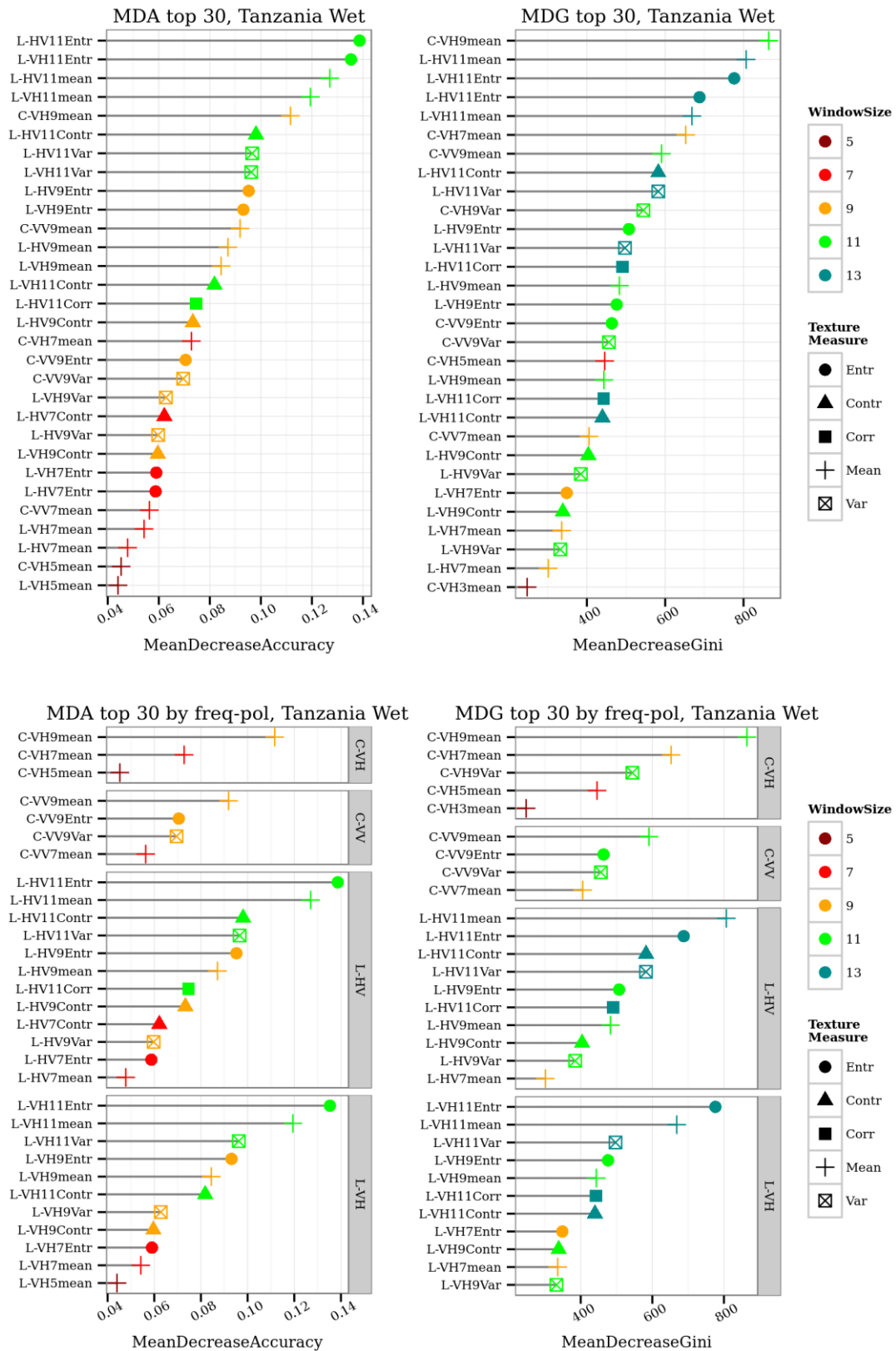


Figure 5.21 Top 30 variables based on MDA and MDG for the TW site random forest model. The top two graphs show variable 1-30 in ranking order, whereas the bottom two graphs are grouped by the frequency-polarisation groups.

### 5.3.3 Model based on the top 30 variables

The top 30 variables are used to develop random forest models for each of the sites. The variable importance, based on MDA and MDG, was used incrementally to develop a random forest model, first using one variable, then two variables, progressively using more variables until all the top 30 variables are used. The order of variables was determined by the MDA and MDG variable importance measures respectively, with the most important variable used first.

The MDA (Figure 5.22) and MDG measures of variable importance (Figure 5.23) yield similar results. It is notable that with using only the top 5 variables for the Cameroon dry season site, an overall classification accuracy of 88% can be obtained, based on MDA. The top 5 variables for the Cameroon dry season site are X-HV entropy (window size 15x15), X-HV entropy (13x13), X-HV correlation (15x15), X-HV entropy (11x11) and L-HH mean (11x11) (from Figure 5.18). The overall classification accuracy increases marginally to 90% with 10 variables, 91% with 15 variables and then decreases again slightly to 90% using the top 30 variables. For the top 30 MDG variables, the overall classification accuracy was marginally lower at 89% for the Cameroon dry season study site. For the Cameroon wet season study site, the overall and class-specific classification accuracies seem to stabilize with 10 variables, at 82% and 81% for the MDA and MDG variables respectively. The overall classification accuracies then climb gradually to 83% and 84% using the top 30 variables for the MDA and MDG variable importance measures, respectively.

The Tanzania dry study site has an overall classification accuracy of 55% using the top 10 variables from the MDA variables and 54% using the top 10 variables from the MDG variables. The overall classification accuracies of the MDA variables remains at about 55% up to the top 25 variables and only then increase to 61% with the top 30 variables. The overall classification accuracies of the MDG variables increase to 60% using the top 20 variables in then to 63% using the top 30 variables. The Tanzania wet study site has an overall classification accuracy of 48% using the top 10 MDA variables and 49% using

the top 10 MDG variables. The overall classification accuracy increases gradually to 50% using either the the top 30 MDA or MDG variables.

#### Classification accuracies for random forest models top 1 to 30 variables (MDA)

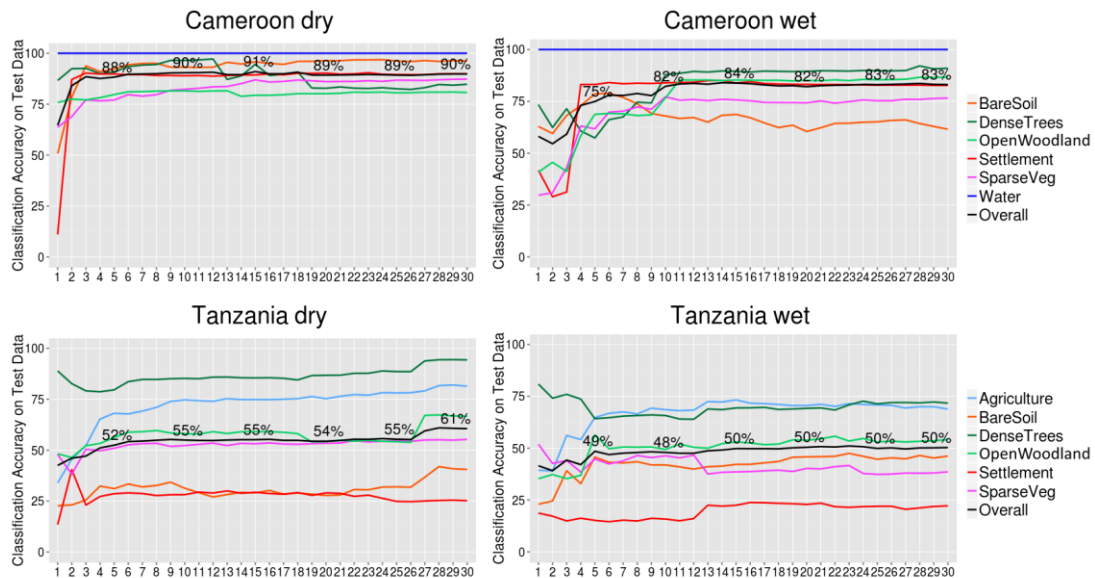


Figure 5.22 The overall and class-specific classification accuracies for the top 30 variables based on the MDA variable importance measure (considering backscatter coefficients, interchannel ratios and texture measures) for each site / season scenario.

To look at the practical implications of a multi-frequency application for land cover classification, the classification accuracies using all available variables were revisited (see Figure 5.1). There appears to be no one clear-cut, one-size-fits-all, scenario. However, some general deductions can be made. If only one frequency could be chosen for all the site-season scenarios, it would be L-band, which would currently be ALOS-2 PALSAR-2 data. L-band gives the best overall classification accuracy, with the exception of the Cameroon dry site, where X-band imagery outperforms L-band and C-band, with 92% overall classification accuracy. L-band as a single frequency gives 76% for the Cameroon dry season site, 78% for the Cameroon wet season site, 63% for the Tanzania dry season site and 51% for the Tanzania wet season site.

### Classification accuracies for random forest models top 1 to 30 variables (MDG)

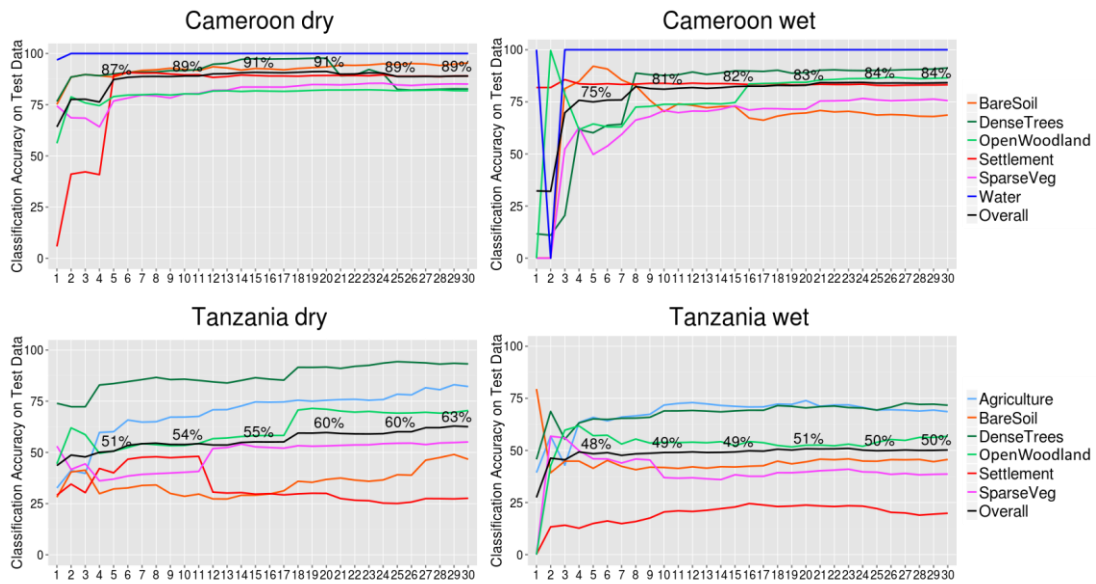


Figure 5.23 The overall and class-specific classification accuracies for the top 30 variables based on MDG variable importance measure (considering backscatter coefficients, interchannel ratios and texture measures) for each site / season scenario.

A combination of LX-bands, in general, give the highest classification accuracies as a dual-frequency application. For the Cameroon dry site, the classification accuracy using CX-bands was slightly higher at 92%, compared to the 90% using LX-bands, and for the Tanzania wet, no X-band data were available. However, the LX-band combination had a higher overall classification accuracy for the other three sites (Cameroon dry, Cameroon wet and Tanzania dry), than other dual-frequency combinations. The combination of L-band and X-band gives 90% overall classification accuracy for the Cameroon dry season site, 88% for Cameroon wet season site and 65% for the Tanzania dry season site. This would entail, for example, combining ALOS-2 PALSAR-2 data with TerraSAR-X data, for the satellites currently operational. The overall recommendation would therefore be to acquire L-band and X-band together, to maximize the benefit of using more than one frequency, without the complications of coordinating acquisitions from three satellites.

#### 5.3.4 Correlation between top 30 variables for MDA and MDG

Graphical representation of the correlation matrices for the MDA and MDG top 30 variables is shown for the Cameroon dry season site in Figure 5.24, for the Cameroon wet season site in Figure 5.25, the Tanzania dry season site in Figure 5.26 and for the Tanzania wet season site in Figure 5.27. Ellipses in combination with colour are used to represent each correlation matrix. Very high positive correlation is represented by an almost flat 45° angle dark blue ellipse, and a low positive correlation is represented by a light-blue circle. Any negative correlation would be a red colour and a highly negative correlation would show as a red almost flat ellipse in a 315° direction. For all the site-season combinations, there are very high correlations between the top 30 variables, often with >80% correlation. This were the case for the Cameroon wet and dry seasons and the Tanzania wet season, with between 15-20 very highly correlated variable in the top 30 list (Figures 5.24, 5.25 and 5.27). For the Tanzania dry season the proportion of highly correlated variables was between 20-25 variables in the top 30 list (Figure 5.26). An alternative model building strategy could be to instead include the top 5-10 variables from each frequency, and compare that model to the overall top 30 model. However, this route was not pursued as part of this research.

It has been shown in Strobl et al. (2008) that the standard variable importance methods from the random forest modelling show a preference towards highly correlated predictor variables. This is due to the nature of random forests of selecting only a portion of all the available variables for each tree during model training. A group of highly correlated variables can then all be chosen as important variables in a final model. In comparison, this is different to standard stepwise linear regression, where it is more likely to only choose one or two variables that represent the same information, after which other variables which are less highly correlated are more likely to be chosen.

To select a set of predictor variables that are less highly correlated, Strobl et al. (2008) suggested a technique called conditional permutation importance. However, in this research, only the standard variable importance measures of MDA and MDG were used

to select the most important variables for each site-season scenario. This was due to the implementation of the conditional permutation importance measure leading to long processing times with the data sets at hand. Nevertheless, the standard version of random forest model implementation in R resulted in high classification accuracies, regardless of high correlation between the variables in the random forest models.



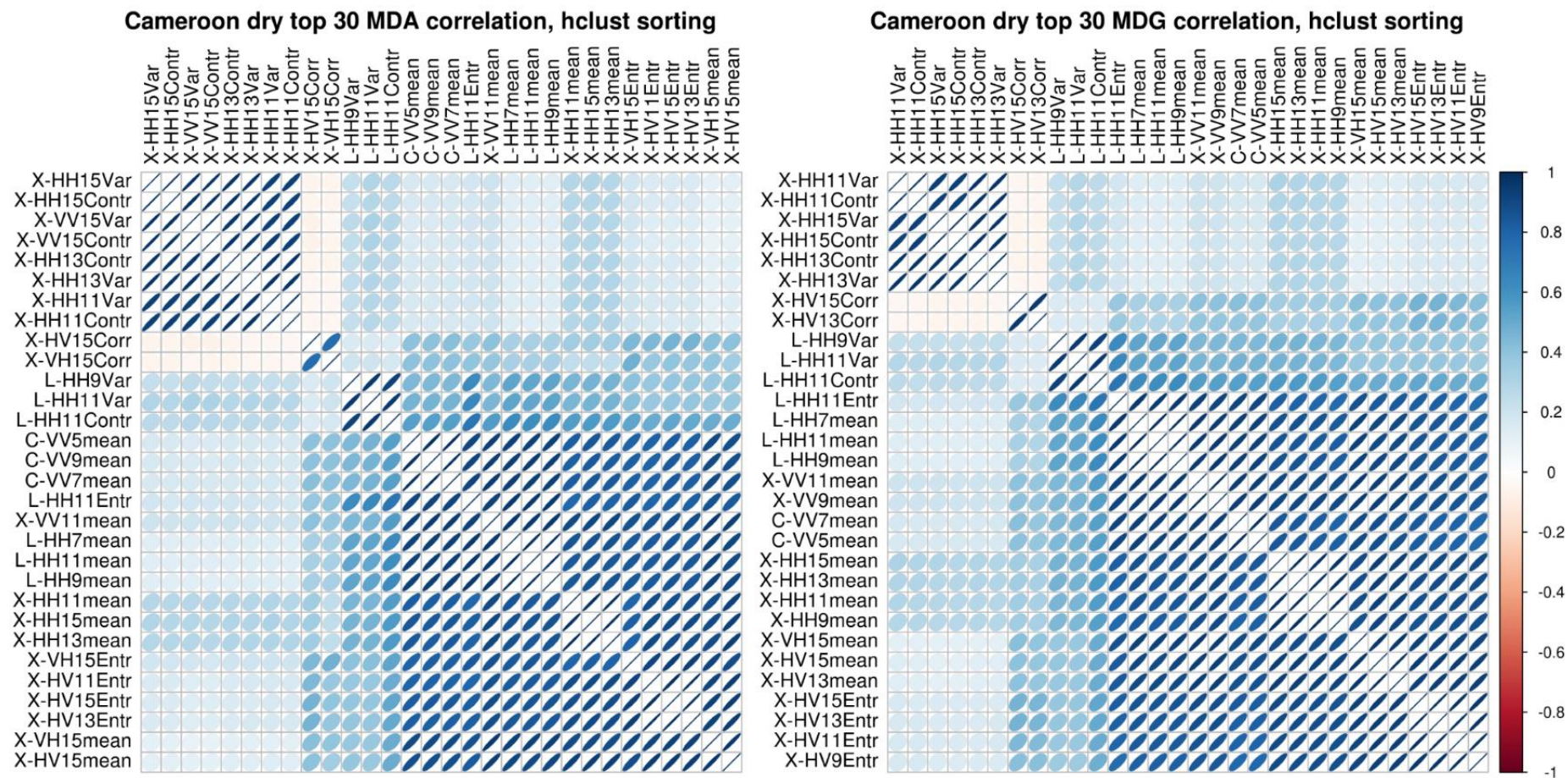


Figure 5.24 Correlation plots for the top 30 variables based on MDA and MDG for the Cameroon dry study site. Ordering of the variables are based on hierarchical clustering.



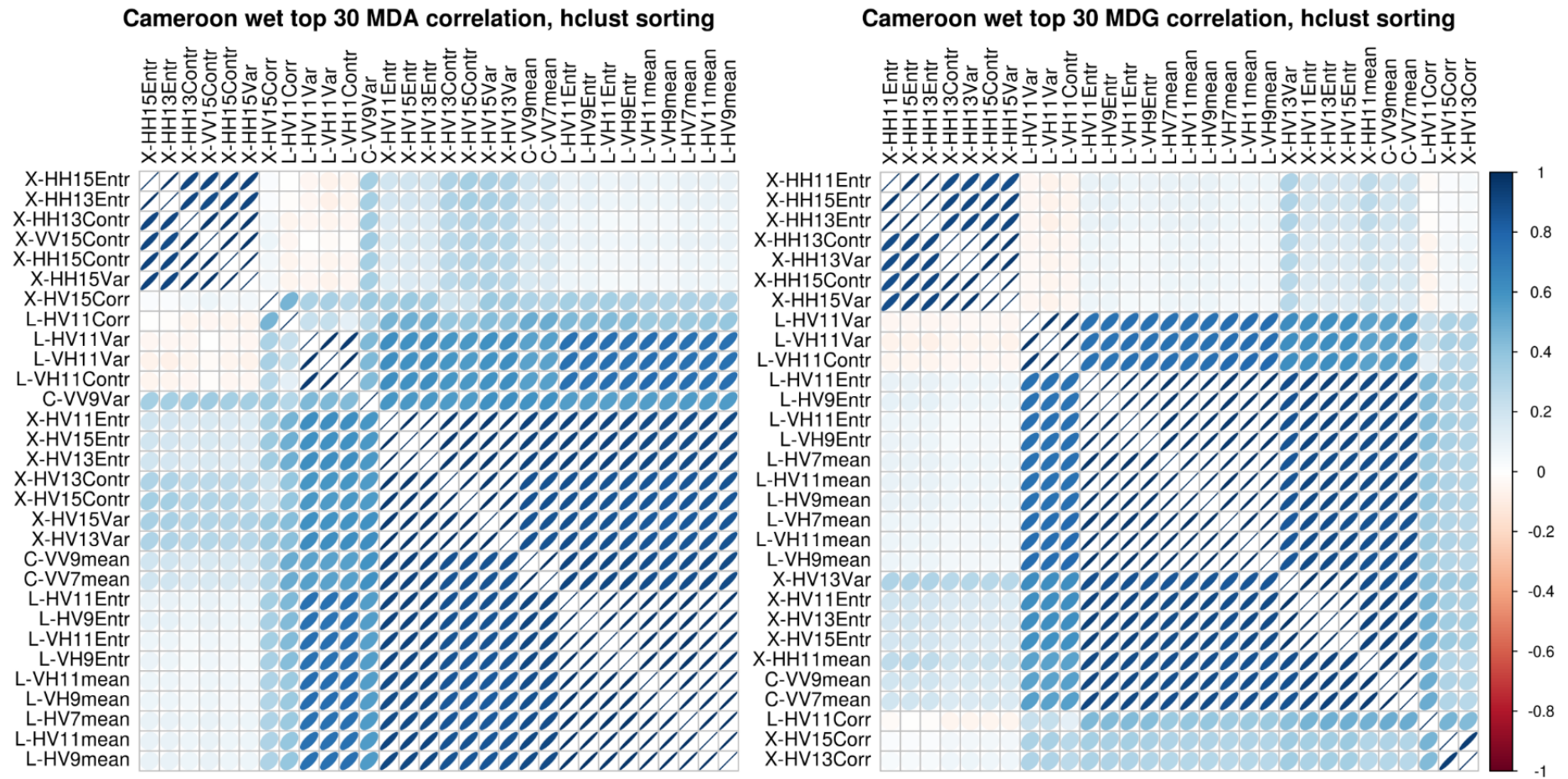


Figure 5.25 Correlation plots for the top 30 variables based on MDA and MDG for the Cameroon wet study site. Ordering of the variables are based on hierarchical clustering.

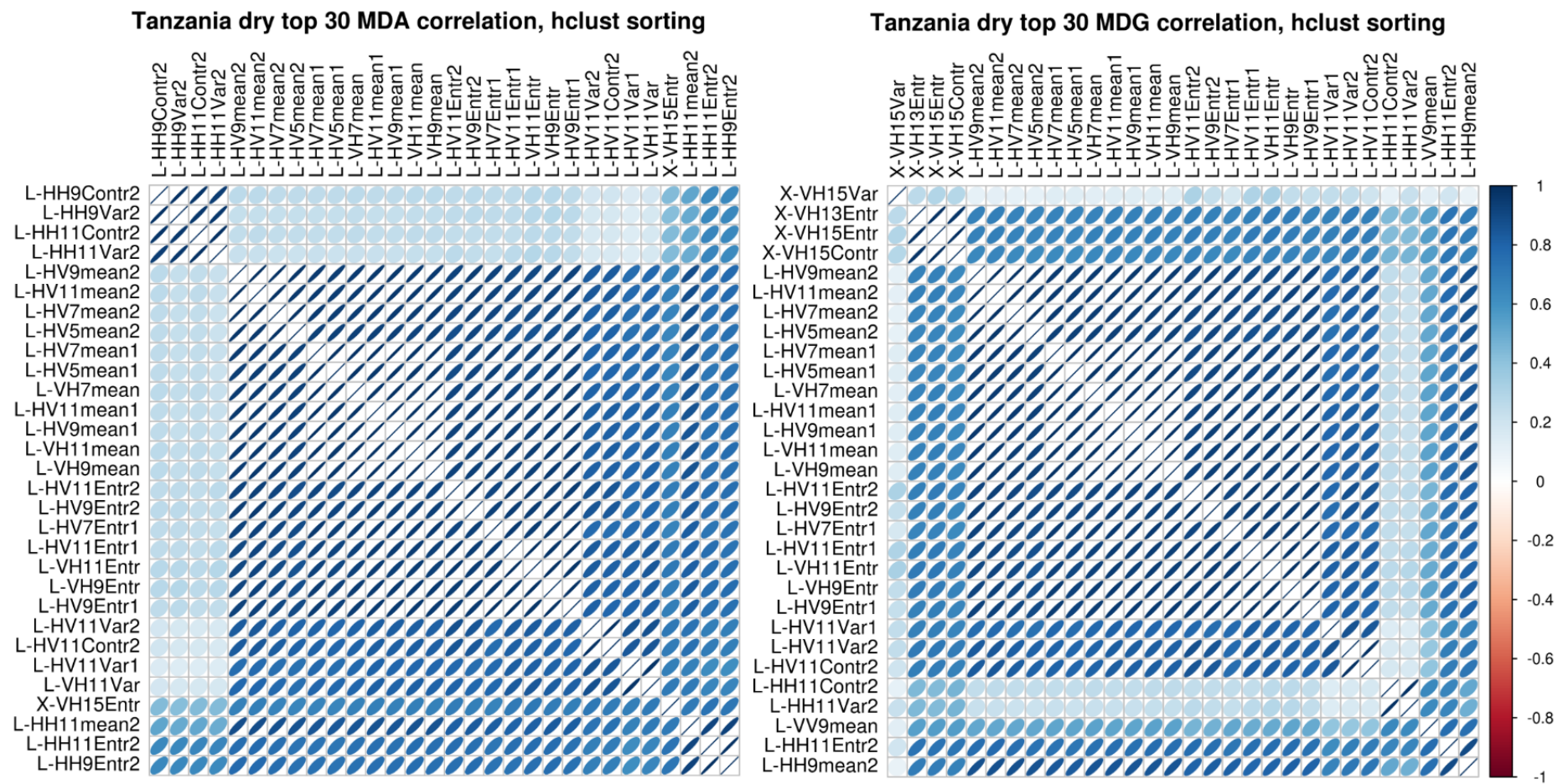
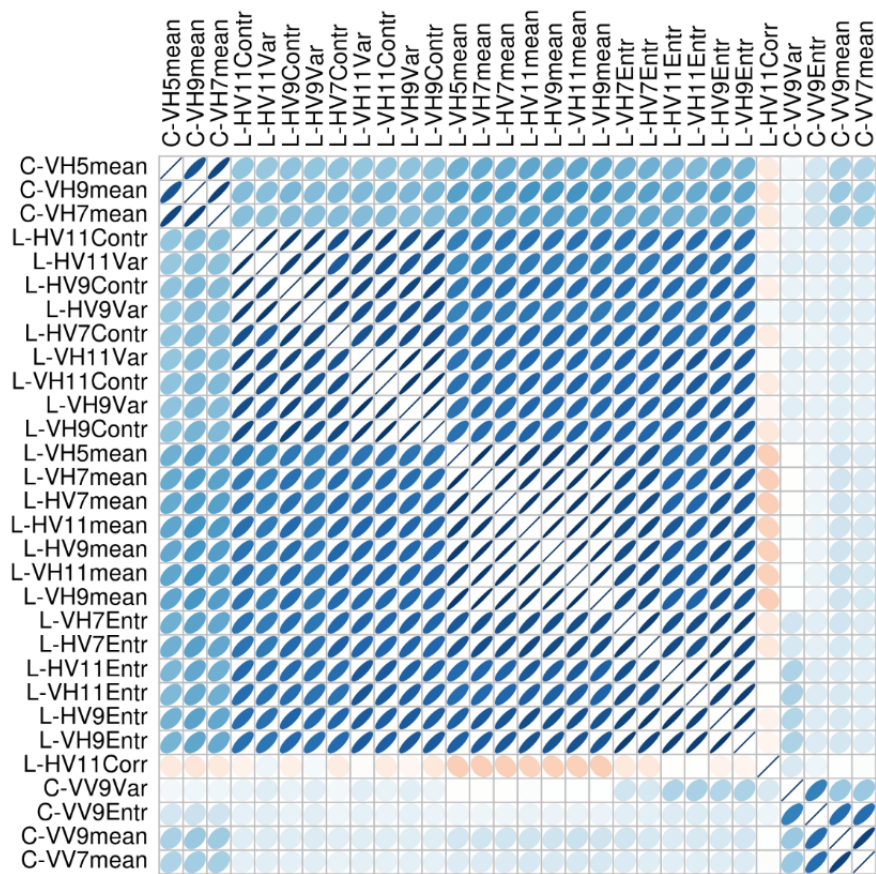


Figure 5.26 Correlation plots for the top 30 variables based on MDA and MDG for the Tanzania dry study site. Ordering of the variables are based on hierarchical clustering.



**Tanzania wet top 30 MDA correlation, hclust sorting**



**Tanzania wet top 30 MDG correlation, hclust sorting**

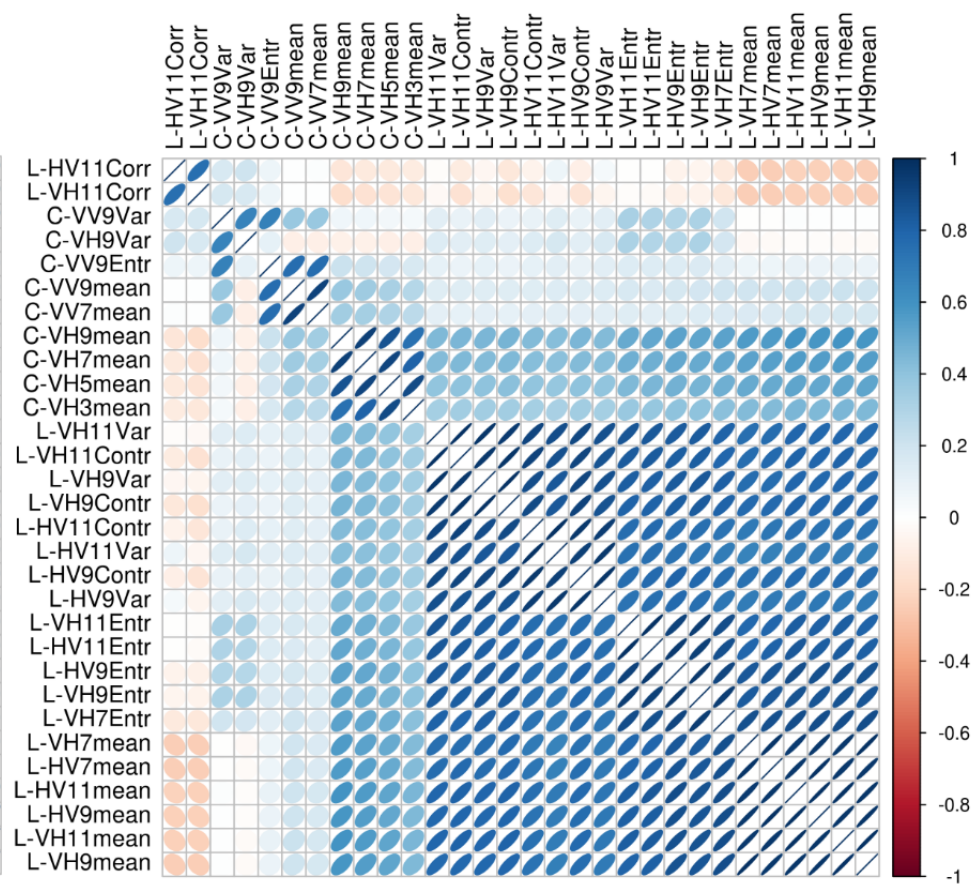


Figure 5.27 Correlation plots for the top 30 variables based on MDA and MDG for the Tanzania wet study site. Ordering of the variables are based on hierarchical clustering.

### 5.3.5 SVM vs random forest classification accuracies

Using the top 30 variables as selected by the MDA and MDG from the random forest modelling, SVM models are also trained based on incrementally adding variables from 1 to 30 for each site-season scenario. The overall and class-specific classification accuracies resulting from these SVM model classifications are shown in Figure 5.28 for the MDA top 30 variables and in Figure 5.29 for the MDG top 30 variables. The results are very similar to the random forest classification results for all site-season combinations, apart from the Tanzania dry season MDA top 30 SVM model, where the overall classification accuracy was significantly higher. The Tanzania dry season MDA and MDG SVM models both have an overall classification accuracy of 68%, compared to 61% and 63% for the random forest models based on the MDA and MDG top 30 variables respectively.

Classification accuracies for SVM models, top 1 to 30 variables (MDA)

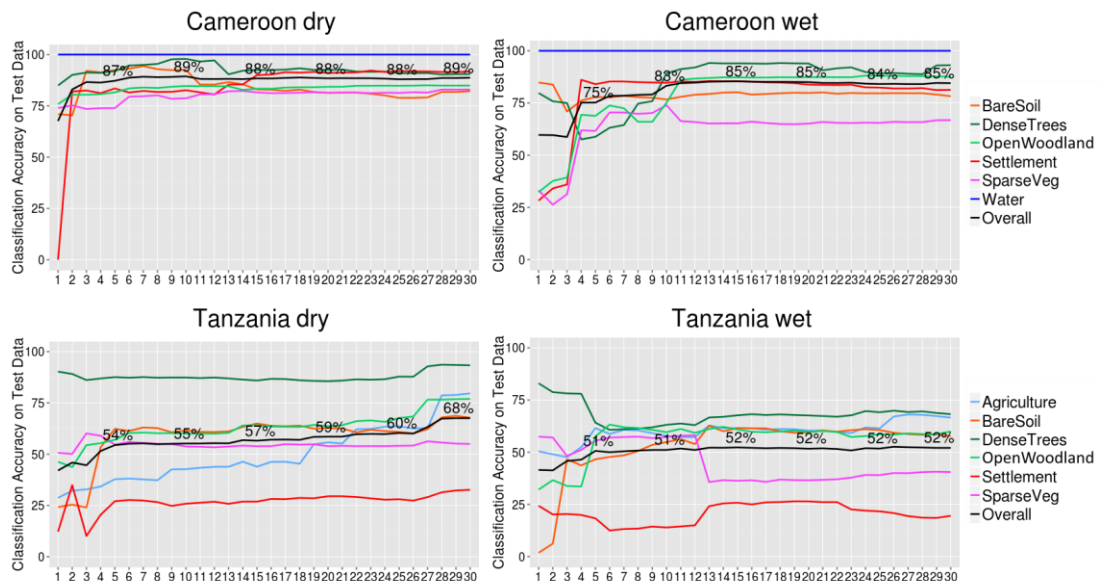


Figure 5.28 Overall and class-specific classification accuracies for the support vector machine classification based on the top 1 to 30 variables selected by the MDA variable importance measure in the random forest classification.

### Classification accuracies for SVM models, top 1 to 30 variables (MDG)

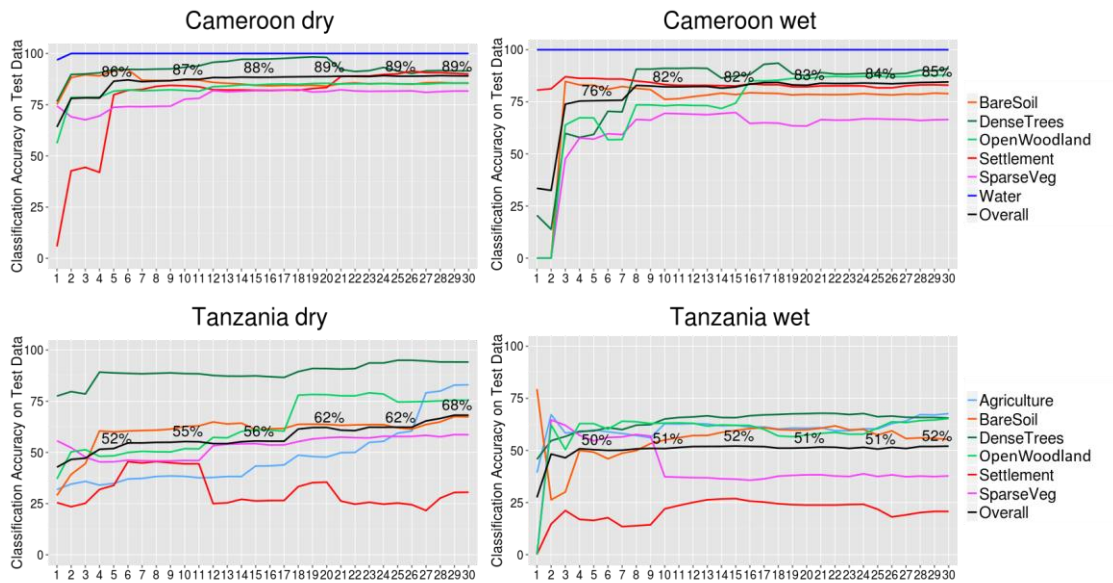


Figure 5.29 Overall and class-specific classification accuracies for the support vector machine classification based on the top 1 to 30 variables selected by the MDG variable importance measure in the random forest classification.

### 5.3.6 Random forest model applied on the complete layer stack of images

Up to this point, all the model training and land cover classification accuracy plots were generated using only the training and test samples as set out in Section 3.4. In this section the random forest models developed using the top 30 variables based on MDA and the MDG are used to classify the full scene where the SAR images overlap for the Cameroon study site.

The classification of the whole image for the Cameroon dry images of the random forest model based on the top 30 MDA variables are shown in Figure 5.30. The classified image was compared to the optical image available on Google Earth of the same area. Figure 5.31 shows a zoomed-in area of the random forest classification compared to the optical images at the most southern part of the lake in the Cameroon study site. The 90% classification accuracy as based on the test samples should be applicable to the whole image as well, as can be seen in the zoomed-in area of the Cameroon site in Figure 5.31.

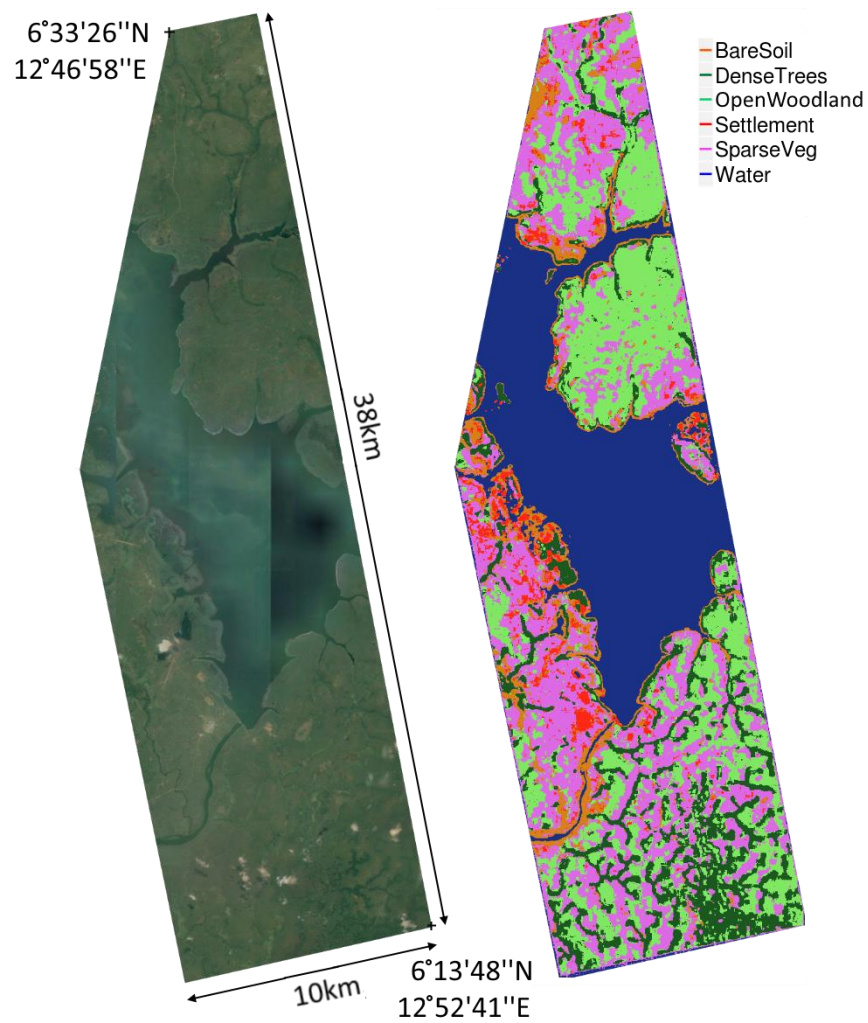


Figure 5.30 The Cameroon study site optical image available on Google Earth on the left compared the random forest classification of the whole image using the Cameroon dry top 30 MDA variables on the right. © 2015 Google, Image © 2015 DigitalGlobe, Image © 2015 CNES /Astrium.



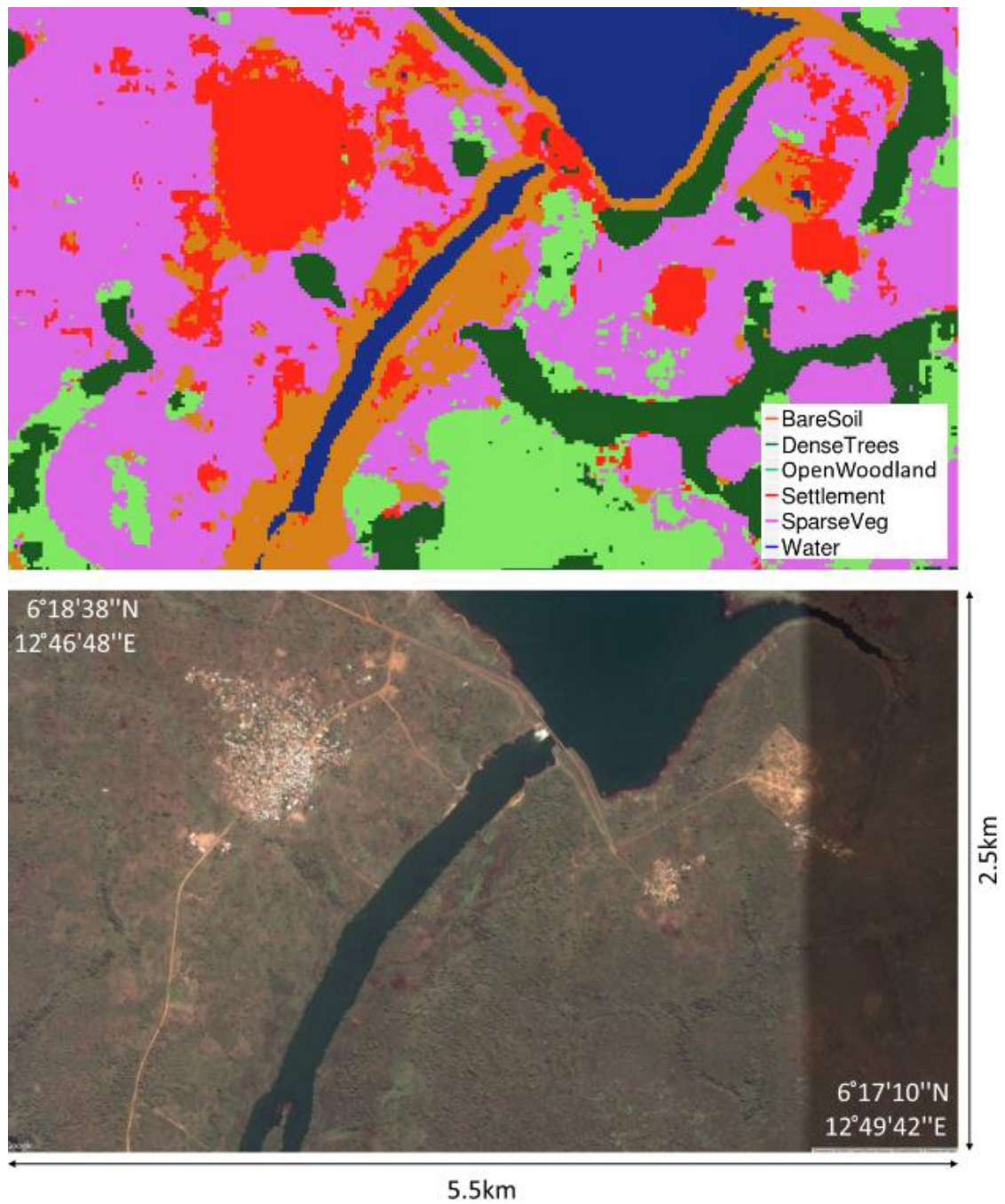


Figure 5.31 Zoomed in section of the random forest classification based on the MDA top 30 variables (top), compared to the very high resolution image of the same area of the Cameroon dry study site. © Google, Image © 2015 DigitalGlobe.

### 5.3.7 Summary of results

The development of a land cover classification model with a list of the top 5, 10 or 30 most important variables was first of all an exercise to increase the dimensionality of the available data. Once the interchannel ratios and texture measures were calculated and the elevation data layers added, a comprehensive list of predictor variables was available from which the most important variables can be chosen for each site-season scenario. The different combinations of variables for each site-season scenario were presented in Section 5.2.1. The two variable importance measures are the MDA and MDG, and both measures are used to select shortlists of the most important variables. The top 30 variables based on MDA and MDG were chosen and the results compared. The same top 30 variables based on the MDA and MDG as part of the random forest model output were then used to train an SVM models and the classification accuracies compared to the random forest models.

The highest overall classification accuracy from variable input scenario are summarized for the Cameroon dry season and Cameroon wet season study sites for the random forest models based on all the mentioned variable combinations, including the top 30 MDA random forest model, top 30 MDG random forest model, top 30 MDA SVM model and top 30 MDG SVM model for the Cameroon wet and dry season sites (Figure 5.32) and for the Tanzania wet and dry season sites (Figure 5.33). The results reveal that the top 30 models give overall classification accuracies very close to using all variables, and even slightly higher for the Tanzania wet and dry SVM top 30 classifications. The dry sites yield higher overall classification accuracies for both the Cameroon and Tanzania sites.



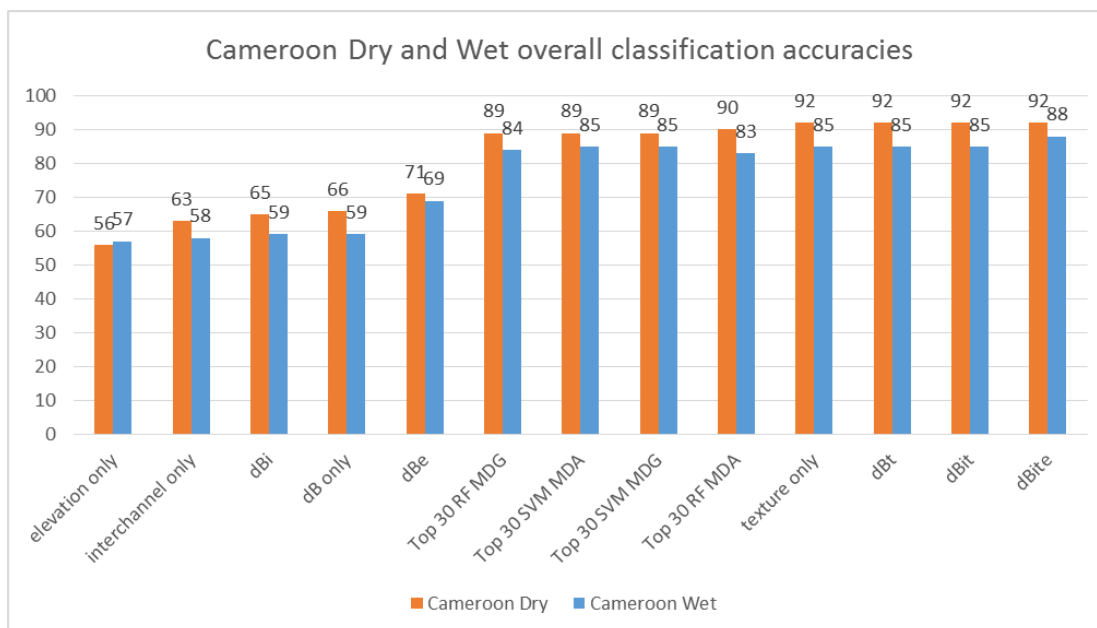


Figure 5.32 A summary of the results of the overall classification accuracies for the models based on different variable combinations for the Cameroon dry and Cameroon wet site. All models are random forest models apart from the two mentioned Top 30 SVM models.

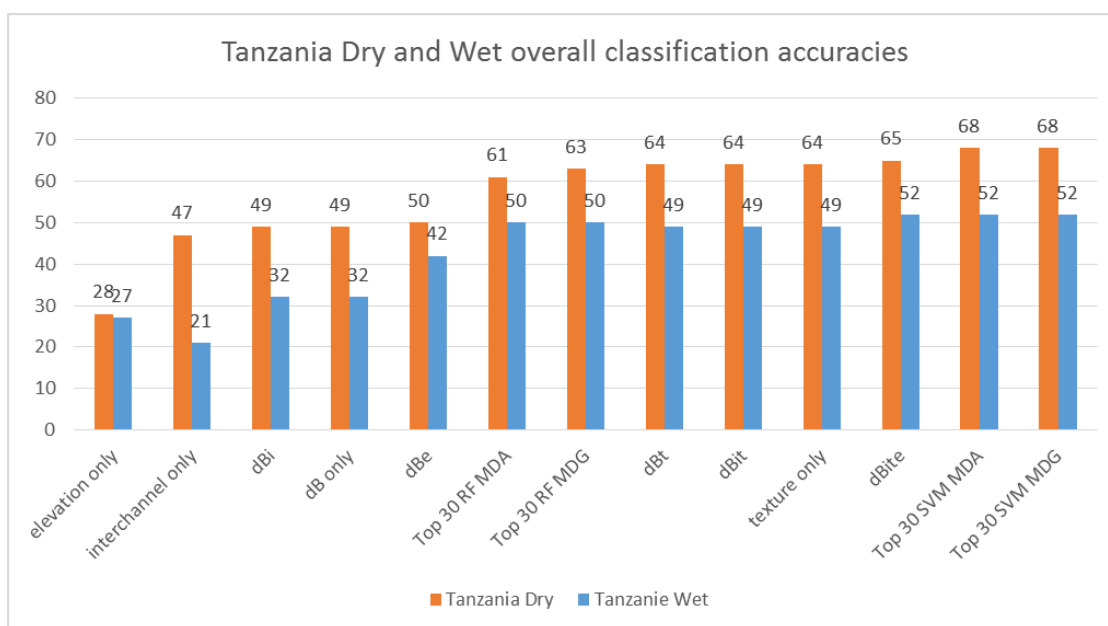


Figure 5.33 A summary of the results of the overall classification accuracies for the models based on different variable combinations for the Tanzania dry and Tanzania wet sites. All models are random forest models apart from the two mentioned Top 30 SVM models.

Table 5.2 Overall classification accuracies of random forest models using (a) single frequencies and (b) dual / three frequencies, with different variable combinations for the forested and semi-arid study sites.

### a) Single frequency

Ecoregion	Site	dB only (Fig. 5.2)			dB + i (Fig. 5.6)			dB + e (Fig. 5.7)			t only (Fig. 5.4)			dB + t (Fig. 5.8)			dB + i + t (Fig. 5.9)			dB + i + t + e (Fig. 5.1)			min	max
		1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd		
Forested	Cameroon dry	58	44	39	58	44	39	68	60	56	91	71	61	91	71	62	91	71	61	92	76	71	L	L
	Cameroon wet	48	48	38	48	47	38	65	62	56	73	69	52	73	69	52	74	69	52	78	77	63	C	C
Semi-arid	Tanzania dry	44	30	26	44	30	26	49	37	34	60	54	40	61	54	43	60	54	43	63	56	50	X	X
	Tanzania wet	27	21	-	27	21	-	37	34	-	46	27	-	46	28	-	47	28	-	51	36	-		

### b) Dual / three frequencies

Ecoregion	Site	dB only (Fig. 5.2)				dB + i (Fig. 5.6)				dB + e (Fig. 5.7)				t only (Fig. 5.4)				dB + t (Fig. 5.8)				dB + i + t (Fig. 5.9)				dB + i + t + e (Fig. 5.1)			
		LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd
Forested	Cameroon dry	66	63	62	52	65	63	61	51	71	70	69	63	91	92	90	78	91	92	90	78	91	92	90	79	91	92	90	80
	Cameroon wet	59	57	53	50	59	58	53	50	69	68	66	63	85	84	76	76	85	84	76	76	85	85	76	76	88	88	80	78
Semi-arid	Tanzania dry	49	48	45	35	49	48	45	34	50	50	50	41	63	64	61	51	64	64	61	52	64	64	62	52	65	65	64	56
	Tanzania wet	-	32	-	-	-	32	-	-	-	42	-	-	-	49	-	-	-	49	-	-	-	49	-	-	-	52	-	-

dB	backscatter coefficients
i	interchannel ratios
t	texture measures
e	elevation data

min	max
LC	LC
LX	LX
CX	CX
LCX	LCX

The four site / season combinations, with the overall classification accuracies of different feature combinations, are all summarized in Table 5.2. From this table, recommendations can be made for each ecoregion and season scenario. The single and dual frequencies are ordered in ranking order of 1st, 2nd and 3rd, in terms of overall classification accuracy. The LCX-bands classification accuracies are shown next to the dual frequency results. Table 5.2 reveals in one view all the classification accuracy results presented thus far. The best single frequency classification results for the forested regions are from X-band, and secondly L-band (92% and 77% for Cameroon dry and wet respectively, using X-band only; and 72% and 78% for Cameroon dry and wet respectively using L-band only). Texture measures add the most value from all the variable types (e.g overall classification accuracies increase from 68% to 91% for the Cameroon dry site; using X-band only; and from 49% to 60% for Tanzania dry using L-band only). The best dual and three frequency scenarios is only marginally higher than the single frequency results (e.g. for Cameroon dry the best dual frequency results are CX at 92%, which is the same as using only X-band at 92%; for Cameroon wet, the highest overall accuracy is from LX-bands at 88%, which is the most significant increase of all the sites; for Tanzania dry, the highest overall accuracy is from LX-bands at 65%, which is marginally higher than only L-band at 63%; and for Tanzania wet, the highest dual frequency combination is LC-band at 52%, which is marginally higher than using only L-band).

The recommended SAR frequencies along with an alternative recommendation are given in Table 5.3. Both Table 5.2 and Table 5.3 are discussed in more detail in Section 5.4, and in Section 8.1.1.

Table 5.3 The recommended SAR frequencies with high classification accuracies for land cover classification within specific ecoregions. This assumes all the available variables of backscatter coefficients, interchannel ratios, texture measures and elevation data are included in the models.

<b>Ecoregions considered</b>	<b>Season</b>	<b>Recommended SAR frequencies (overall classification accuracy)</b>	<b>Alternative recommendation (overall classification accuracy)</b>
<b>Both semi-arid and forested*</b>	Wet and dry	LX* or LC	L-band only
<b>Forested only</b>	Wet and dry	LX (wet: 90% and dry: 88%)	X-band only (wet: 92% and dry: 77%)
	Dry only	X-band only (92%)	LX (90%), CX (92%) or L-band only (76%)
	Wet only	LX (88%)	L-band only (78%), or X-band only (77%)
<b>Semi-arid</b>	Wet and dry	L-band only (dry: 63% and wet: 51%)	LC (dry: 64% and wet: 52%)
	Dry only	L-band only (63%)	LX (65%)
	Wet only	L-band only (51%)	LC (52%)

\*For both semi-arid and forested sites, LX band models can only be recommended for the Cameroon wet and dry, and Tanzania dry season, and not the Cameroon wet season with the current available data. The alternatives of LC and L-band only are recommended for all four site-season scenarios.

## 5.4 Discussion

For the Cameroon dry and wet season sites, the top 30 random forest models give accuracies very close to that achieved using all the available layers, as shown in Figure 5.32. For the Cameroon dry site, the complete data stack consists of 165 layers and the Cameroon wet complete data stack of 245 data layers. The overall classification accuracies using only the top 30 variables are very close to the overall classification accuracies using the full data stack for model development. As mentioned in Section 5.3.3, using only the top 5 variable for the Cameroon dry site achieved 88% overall classification accuracy and the top 10 variables for the Cameroon wet site achieved 82%

overall classification accuracies using only a fraction of the available feature layers (see Figures 5.22 and 5.23). This greatly increased the processing time and reduced the disk space required while still maintaining a very high overall classification accuracy.

It was surprising that for the Tanzania dry study site, the Top 30 MDA SVM and Top 30 MDG SVM models achieved higher classification accuracies (68%) compared to the top 30 random forest models as well as compared to the random forest models using all available data layers (65%) (Figure 5.33). As mentioned in Section 4.4, the complete data stack for the Tanzania wet study site was 134 layers and for the Tanzania dry study site 219 layers. For the Tanzania wet study site, the highest overall classification accuracy of 52% was achieved using either all the variables (dBite), or the top 30 SVM MDA or the top 30 SVM MDG variables. Similar to the Tanzania dry season SVM models, the wet season SVM models also outperform the top 30 MDA and MDG random forest models.

In the results section and discussion above, it was apparent that the data layers adding the most value to the classification accuracies are the texture layers, with the top 30 most important variables all being texture measures. This is after the elevation data were excluded from the top 30 variable selection, so that models were transferable to other forested and semi-arid areas regardless of the elevation above sea-level. The texture measures added considerably more information compared to using only the backscatter coefficients on their own, with an increase of between 15-25% overall accuracy for both the Cameroon and Tanzania study sites, between using only the backscatter coefficients and the top 30 model, which consists only of texture measures, for each study site. The comparison between the MDA and MDG variable importance measures does not yield a distinct winner, as neither the MDA nor the MDG variable importance measures gave consistent higher classification accuracies between the different site-season scenarios.

The class-specific classification accuracies, as presented in Section 5.3.3 for the random forest models and Section 5.3.5 for the SVM models, allow the researcher to select the

best model for a specific land cover class of interest. This is because, even though the overall classification accuracies may increase with the number of variables in some cases, the class-specific accuracy may decrease at a certain point. For example, the classification accuracy for *settlements* for the Tanzania dry season random forest model was higher using the top 5 or top 10 variables compared to using all 30 variables, based on the MDG variables importance measure (see Figure 5.23). Similarly, for the Cameroon wet season site, using the top 5 or top 10 MDA variables achieved a higher classification accuracy for the *sparse vegetation* class, than using the top 30 MDA variables.

Therefore, the best combination of variables depends on both the study site and whether the overall accuracy or class-specific accuracies are the most important to achieve. For the Cameroon dry and wet sites, the top 10 variables already give very high overall classification accuracy (similar to the top 30 variables). The Tanzania wet season study sites gave similar results between the top 10 and the top 30 variables, whereas the top 30 variables for the Tanzania dry season study sites achieves much better overall classification accuracies compared to using only the top 10 variables, with 63% compared to 54% for the MDG variables, and 61% compared to 55% for the MDA variables.

The SVM models using the same top 30 MDA or MDG variables from the random forest modelling gave either the same or slightly higher overall classification accuracies, with a notable increase for the Tanzania dry season study site for the SVM vs the random forest models. It was noted, however that the classification of the SVM model takes considerably longer to run on the complete stack of images, compared to the random forest models for all the site-season scenarios. In Chapter Six the transferability of the models developed in this chapter is assessed to additional study sites in the DRC and Chad and in Chapter Seven the SAR signatures of the backscatter coefficients and the derived layers are compared.

## 5.5 Conclusion

Through an extensive investigation of random forest models of single, dual and three-frequency scenarios and of the various combinations from backscatter coefficients, interchannel ratios, texture measures and elevation data, several conclusions can be made. Firstly, it should be noted that it was possible to extract the LCCS land cover classes that were identified in Section 3.4, to a relatively high level of accuracy. For the Cameroon dry site, all land cover classes were extracted at > 75% classification accuracy. For the Cameroon wet season, all classes were extracted > 75%, apart from the sparse vegetation class that were 60% classification accuracy. The class specific classification accuracies for the Tanzania dry site were > 75% for the dense trees and agricultural classes, 65% for open woodland, 52% for sparse vegetation and < 50% for bare soil and settlement. For the Tanzania wet site were dense trees and agriculture at ~70%, open woodland and bare soil ~50%, sparse vegetation at 40% and settlement at 24%.

There are some inconsistencies between the forested and semi-arid sites. First of all, the classification accuracies are much lower for the semi-arid sites (63% for the Tanzania dry, 51% for the Tanzania wet, using only L-band), compared to the forested site (78% for the Cameroon wet, 76% for the Cameroon dry, using only L-band). This could be explained by several factors. The land cover areas are more varied and small mosaics of different land cover classes in the Tanzania site, compared to the larger homogenous land cover of the Cameroon site. Additionally, for the Tanzania wet season, no X-band images were available, which could have increased the classification accuracy for this site. The settlement class in particular was not classified well for either the Tanzania dry or wet season (~25%), so by excluding this class, the overall classification accuracy should increase notably (Figures 5.22, 5.23). The other classes that did not perform well are the sparse vegetation and bare soil classes. This indicates that there is some confusion to separate these classes, and this could be resolved by assessing the rainfall events close to the SAR image acquisition more closely; or by acquiring high resolution

images at a similar time to the SAR images; or by actual field visits, to understand the vegetation growth on the ground better.

When assessing a view over all the site-season scenarios, L-band gives the highest classification accuracy, as a single frequency. For two frequencies, and using all features (or any that include texture measures), the combination of LX-bands give the highest overall classification accuracies in general. This can be attributed to the combination of long (L-band) and short (X-band) frequencies to observe different parts of the vegetation, i.e. the L-band reflects the stems (everything greater than ~23.5cm), whereas the X-band reflects the leaves and canopy (everything greater than ~3cm). It can be argued however, that it is also due to the higher spatial resolution from X-band (5m) and L-band (10m). However, as single frequencies, L-band gave higher overall classification accuracy for all sites apart from the Cameroon dry site.

The addition of texture measures increases the classification accuracies with 15-25% for all site-season scenarios, when compared to using only backscatter coefficients. The texture measures enhances many of the land cover classes, as in the examples shown in Section 4.3.2. Settlements stand out through the entropy and contrast texture measures from X-band and the mean and entropy texture measures emphasises the bright and dark areas corresponding to dense trees and bare soil areas. Using the top 30 variables gave classification accuracies very close to using all the variables for each site / season scenario, for both the MDA and MDG variables. The SVM models performed similar to the random forest model for the Cameroon sites, and gave slightly higher overall classification accuracies for the Tanzania sites, than the random forest models, when using the same top 30 variables. This was not expected, and the reason for this is not currently certain. This will have to be investigated more fully, and possibly checked with when using different samples, to see if the same results are maintained. The next step is to assess how transferable these models are, when transferring the model to additional sites from a different geographic region in Chapter Six.



## **Chapter Six      Transferability of random forest classification models on additional study sites in the DRC and Chad**

The hypothesis that SAR models are transferable between different seasons of the same site, and to additional sites of a similar ecoregion, is tested in this chapter. The classification accuracies of inter-season and inter-site transferability of the models trained in Chapter 5 are compared for both the forested and semi-arid ecoregions, to test the hypothesis.

### **6.1      Introduction**

Most remote sensing studies focus on applying models on a single geographic location. The aim of this chapter is to assess the transferability of the developed random forest models, and to assess the accuracy of transferring the algorithms to a different season and different geographic location, although still to a similar ecoregion. This will assess how applicable the models are to new data.

The robustness of the developed classification algorithms needs to be tested for various scenarios based on available data. To determine if an algorithm is transferable and applicable to use operationally, it is important that the algorithms give similar results for different datasets and areas (Waske and Braun, 2009). Since the main model development dataset and the additional season / additional site do not all have the same frequency-polarisation images available, only the frequency-polarisation images that are available for both the main and additional sites are used to train the random forest models that are used to test transferability.

The additional test site for the forested ecoregion is located in the DRC and for the semi-arid ecoregion in Chad (see Sections 3.1.2 and Section 3.1.4). Both sites were previously used for the SAFER project, and the findings will therefore be applicable to emergency response situations surrounding Internally Displaced Persons (IDP) and refugee camps (SAFER, 2012).

## 6.2 Specific methodology on transferability and generalization of classification algorithms

The transferability was tested by applying the models that were developed on the main development sites on the additional season data and additional site data for each ecoregion.

Two land cover class definition approaches are considered as shown in Table 6.1. The first land cover class definition is where the model is trained on the Cameroon wet season land cover classes, namely *bare soil*, *dense trees*, *settlement*, *sparse vegetation* and *water* (Table 6.1, 1a and 2a). The *open woodland* class is left out of this chapter, since it is not present in the DRC wet dataset.

The second land cover class definition is where the model is trained on both the main development and the additional site datasets. For the inter-season comparison for the forested sites, the model is therefore trained on both the Cameroon wet season and Cameroon dry season training datasets, and applied to the Cameroon wet season and Cameroon dry season test datasets (Table 6.1, 1b). For the inter-site comparison for the forested sites, the model is trained on both the Cameroon wet season and the DRC wet season training datasets, and then applied to the Cameroon wet season and DRC wet season test datasets (Table 6.1, 2b).

For the semi-arid study sites, the transferability of the random forest models is also assessed. The models developed on the Tanzania dry season data are applied to the Tanzania wet season data, for an inter-season comparison, and to the Chad dry season data, for an inter-site comparison. Again, the originally selected ‘training’ samples are used for the model development and comparison. Similar to the forested sites, two land cover class scenarios are assessed: first, by applying the model directly on the new data with the same classes, and secondly, by using site-specific land cover classes and developing and testing the model on both sites. The training and test site combinations, along with the land cover classes, are shown in Table 6.2. For the direct application of the model, the semi-arid land cover classes are *agriculture*, *bare soil*, *dense trees*,

*settlement* and *sparse vegetation* (Table 6.2, 3a and 4a). The site-specific land cover classes for the semi-arid inter-season comparison are given in Table 6.2, 3b; and the site-specific land cover classes for inter-site comparison in Table 6.2, 4b.

The direct application of the model with the same land cover classes assesses the transferability of the model in a direct way. The training and application of the model using the site-specific land cover classes aim to determine if the land cover classes between the different seasons and sites, can be separated successfully by the classification algorithm. This will prove that the land cover classes are in fact not the same (e.g. dense trees in Cameroon vs dense trees in the DRC), and can be seen as distinct classes. As a result, the number of land cover classes increases, and a refined LCCS land cover definition can be deducted to classify the land cover classes between the sites as separate land cover classes.

Table 6.1 The training data, test data and different land cover class scenarios for the forested sites.

Nr	Model training dataset	Model test dataset	Land cover classes (directly applied and site-specific)	Results section / Figure number
1a)	Cameroon wet season (training samples)	Cameroon dry season (training samples)	<i>Directly applied:</i> bare soil dense trees settlement sparse vegetation water	Section 6.3.1.1 / Figure 6.1
1b)	Cameroon wet and Cameroon dry season (training samples)	Cameroon wet and Cameroon dry season (test samples)	<i>Site-specific:</i> CW bare soil CW dense trees CW settlement CW sparse vegetation CW water CD bare soil CD dense trees CD settlement CD sparse vegetation CD water	Section 6.3.1.1 / Figure 6.2
2a)	Cameroon wet season (training samples)	DRC wet season (training samples)	<i>Directly applied:</i> bare soil dense trees settlement sparse vegetation water	Section 6.3.1.2 / Figure 6.3
2b)	Cameroon wet and DRC wet season (training samples)	Cameroon wet and DRC wet season (test samples)	<i>Site-specific:</i> CW bare soil CW dense trees CW settlement CW sparse vegetation CW water DW bare soil DW dense trees DW settlement DW sparse vegetation DW water	Section 6.3.1.2 / Figure 6.4

Table 6.2 The training data, test data and different land cover class scenarios for the semi-arid sites.

Nr	Model training dataset	Model test dataset	Land cover classes (directly applied or site-specific)	Results section / Figure number
3a)	Tanzania dry season (training samples)	Tanzania wet season (training samples)	<i>Directly applied:</i> agriculture bare soil dense trees settlement sparse vegetation	Section 6.3.2.1 / Figure 6.5
3b)	Tanzania dry and Tanzania wet season (training samples)	Tanzania dry and Tanzania wet season (test samples)	<i>Site-specific:</i> TD agriculture TD bare soil TD dense trees TD settlement TD sparse vegetation TW agriculture TW bare soil TW dense trees TW settlement TW sparse vegetation	Section 6.3.2.1 / Figure 6.6
4a)	Tanzania dry season (training samples)	Chad dry season (training samples)	<i>Directly applied:</i> agriculture bare soil dense trees settlement sparse vegetation	Section 6.3.2.2 / Figure 6.7
4b)	Tanzania dry and Chad dry season (training samples)	Tanzania dry and Chad dry season (test samples)	<i>Site-specific:</i> TD agriculture TD bare soil TD dense trees TD settlement TD sparse vegetation ChD agriculture ChD bare soil ChD dense trees ChD settlement ChD sparse vegetation	Section 6.3.2.2 / Figure 6.8

## 6.3 Results

For both the forested and semi-arid ecoregions, random forest models were developed for different input variable scenarios. The combination of input variables considered here is taken from the possible scenarios in Table 5.1 from Section 5.3.1. The scenarios chosen are the different frequency-combinations of (i.e. L- , C- and X-band, LC-, LX-, CX- and LCX-bands) of at most the top 30 variables of specific feature-type combinations. The chosen variables are based on the MDG variable importance measure from the random forest model development output, as discussed in Chapter Five.

### 6.3.1 Transferability of Cameroon wet season random forest models

Several random forest models are trained on the Cameroon wet season data and applied directly on the Cameroon dry season data in Section 6.3.1.1, and to the DRC wet season data in Section 6.3.1.2. The models are first applied directly, and secondly using the site-specific land cover classes. The site-specific classes implies that the Cameroon land cover classes and that DRC land cover classes are seen as distinct land cover classes. The model is therefore trained on the Cameroon and DRC training datasets and applied on the Cameroon and DRC test datasets. Instead of only classifying only bare, dense trees, settlement, sparse vegetation and water classes, the number of classes were expanded to include CW\_bare, CW\_densetrees, CW\_settlement, CW\_sparsevegetation, CW\_water, DW\_bare, DW\_densetrees, DW\_settlement, DW\_sparsevegetation and DW\_water. By doing this, the land cover classes from both sites are seen as separable land cover classes, and instead of classifying five land cover classes, ten land cover classes are classified by the algorithm.

#### 6.3.1.1 Cameroon wet season models applied to Cameroon dry season data

Figure 6.1 shows the overall and class-specific classification accuracies for the different input variable scenarios for the random forest models trained on Cameroon wet season

training data and applied directly on the Cameroon dry season training data, using the same land cover classes.

Classification accuracies of random forest models, built on Cameroon wet season data and applied directly to Cameroon dry season data using the same classes

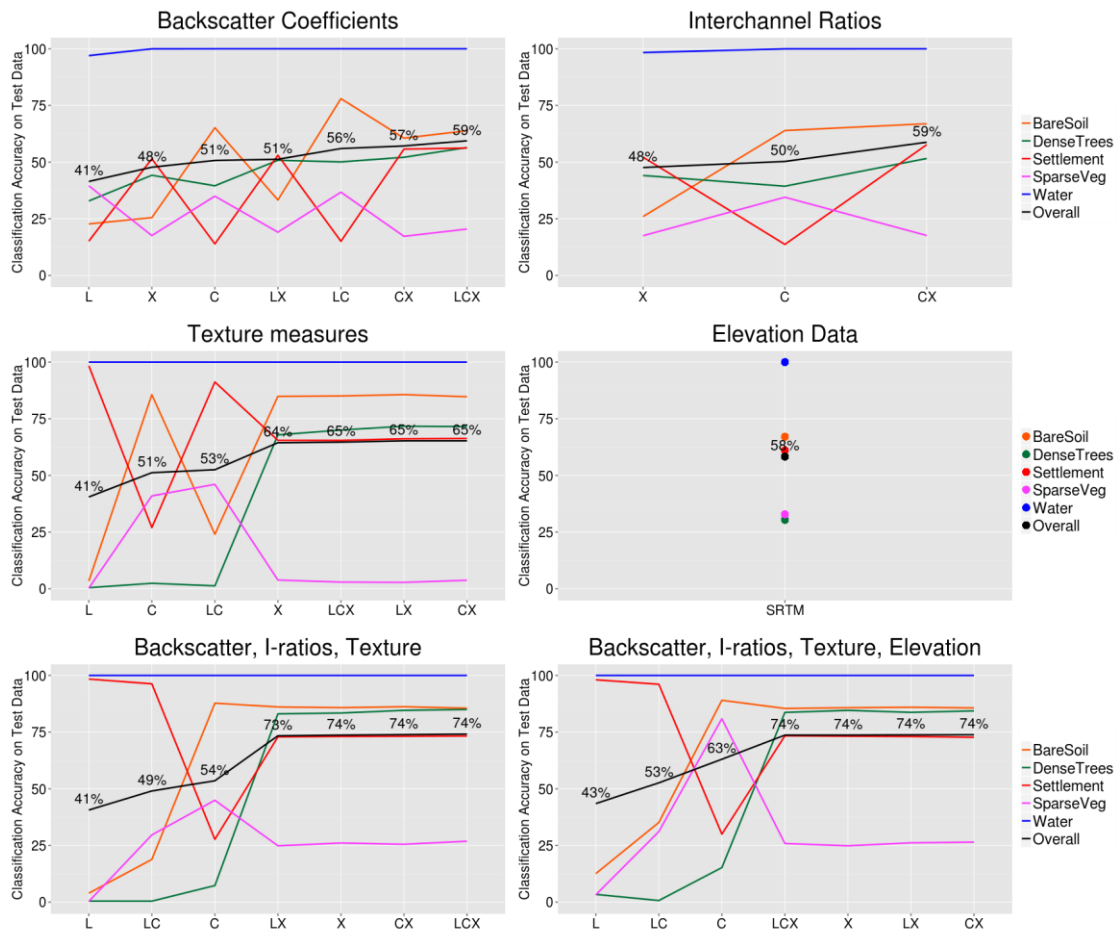


Figure 6.1 Classification accuracies of random forest models based on Cameroon wet season training samples, applied directly to Cameroon dry season samples. Several combinations of input variables for the various frequency combinations are shown.

Figure 6.2 shows the overall and class-specific classification accuracies for the random forest models built on the Cameroon wet season training data and applied to the Cameroon dry season training data, using the site-specific land cover classes.

Classification accuracies of random forest models, built on Cameroon wet season data and applied to Cameroon dry season data, using site-specific classes

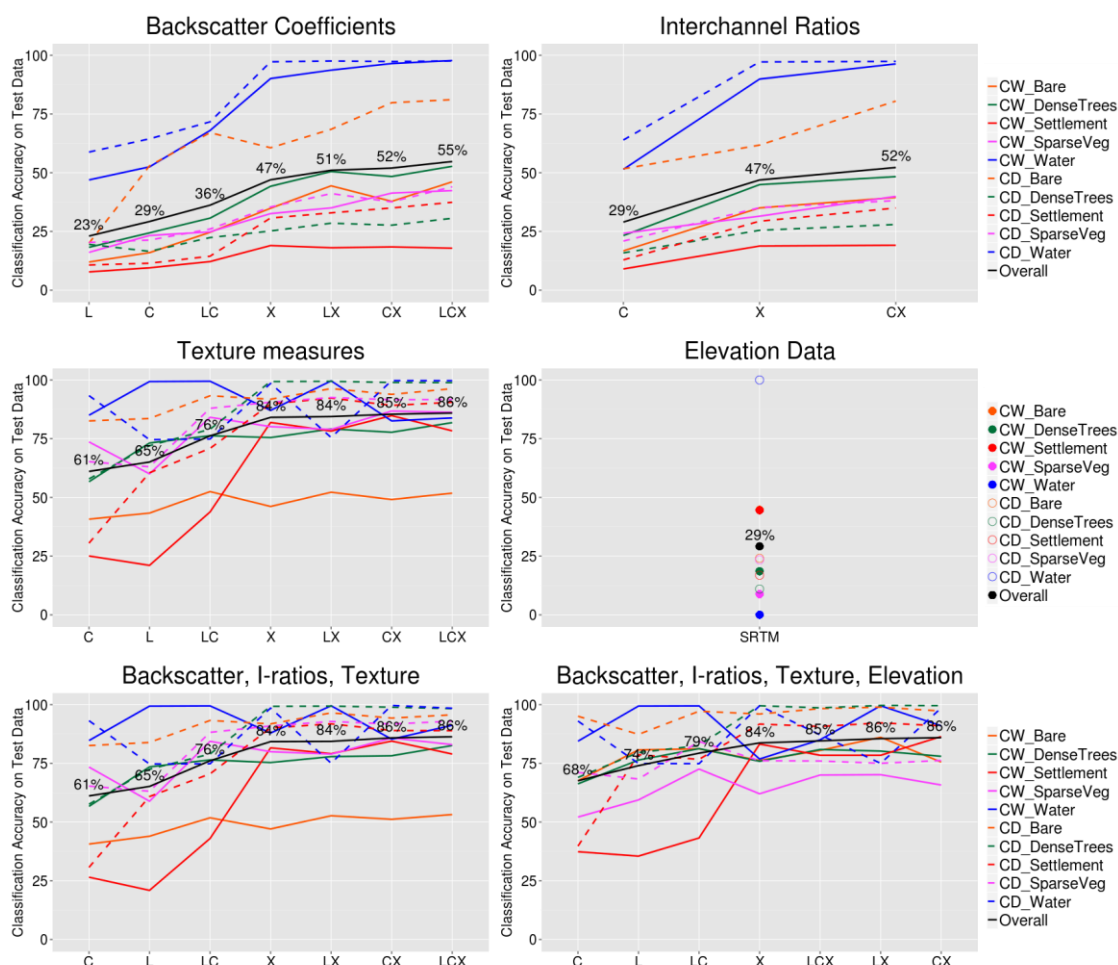


Figure 6.2 Classification accuracies of random forest models based on Cameroon wet season data, applied to Cameroon dry season data, with site-specific classes. Several combinations of input variables for the various frequency combinations are shown.

### 6.3.1.2 Cameroon wet season models applied to DRC wet season data

Figure 6.3 shows the overall and class-specific classification accuracies for the different input variable scenarios for the random forest models trained on Cameroon wet season training data and applied directly on the DRC wet season training data, using the same land cover classes.



Classification accuracies of random forest models, built on Cameroon wet season data and applied directly to DRC wet season data using the same classes

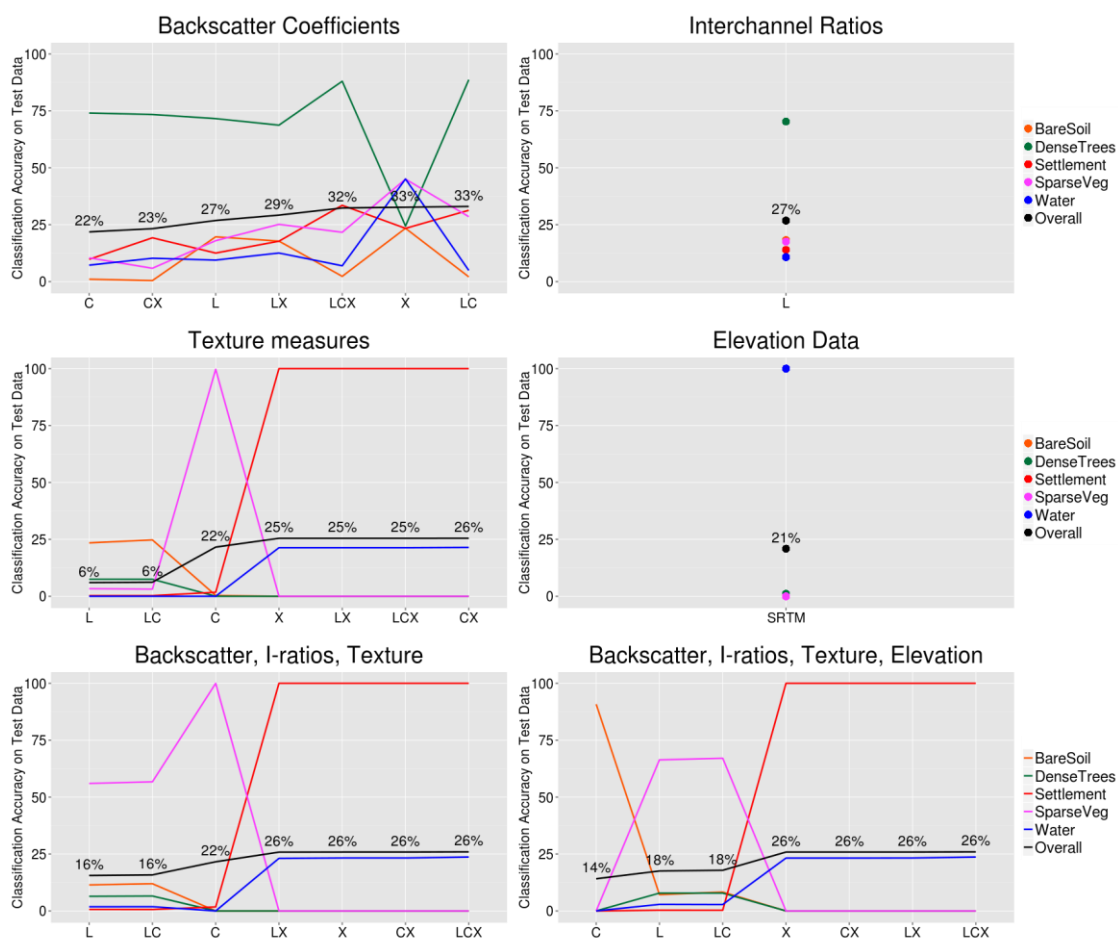


Figure 6.3 Classification accuracies of the random forest models based on Cameroon wet season data, applied directly to DRC wet season data. Several combinations of input variables for the various frequency combinations are shown.

Figure 6.4 shows the overall and class-specific classification accuracies for the random forest models built on the Cameroon wet season training data and applied to the DRC wet season training data, using the site-specific land cover classes. The results of the Cameroon wet season models applied on the Cameroon dry season data and the DRC wet season data are discussed in detail in Section 6.4.

Classification accuracies of random forest models, built on Cameroon wet season data and applied to DRC wet season data, using site-specific classes

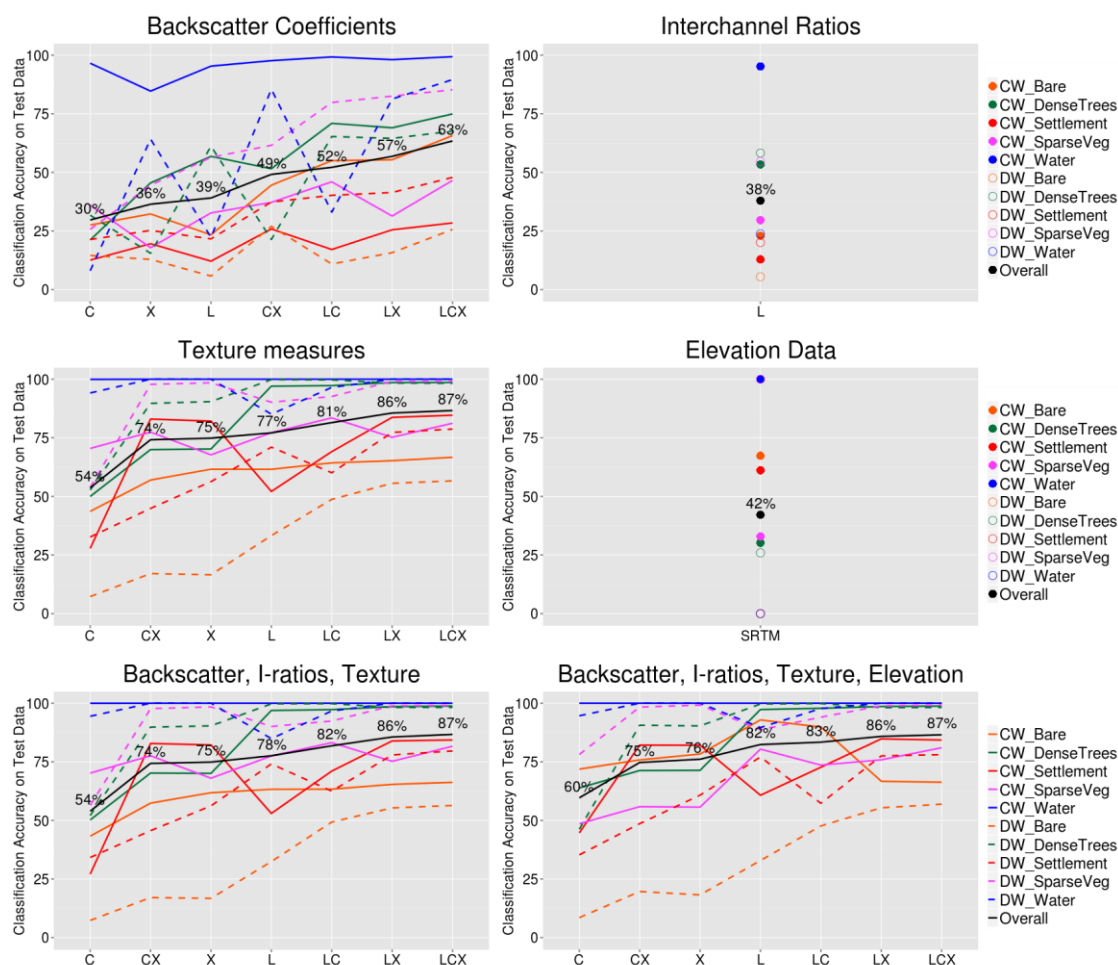


Figure 6.4 Classification accuracies of random forest models based on Cameroon wet season data, applied to DRC wet season data, with site-specific classes. Several combinations of input variables for the various frequency combinations are shown.

### 6.3.2 Transferability of Tanzania dry season random forest models

For the semi-arid study sites, several random forest models are trained on the Tanzania dry season data and applied on the Tanzania wet season data in Section 6.3.2.1, and to the Chad dry season data in Section 6.3.2.2. The models are first applied directly, and secondly using the site-specific land cover classes.

### 6.3.2.1 Tanzania dry season random forest model applied to Tanzania wet season data

Figure 6.5 shows the overall and class-specific classification accuracies for the different input variable scenarios for the random forest models trained on Tanzania dry season training data and applied directly on the Tanzania wet season training data, using the same land cover classes.

Classification accuracies of random forest models, built on Tanzania dry season data and applied directly to Tanzania wet season data using the same classes

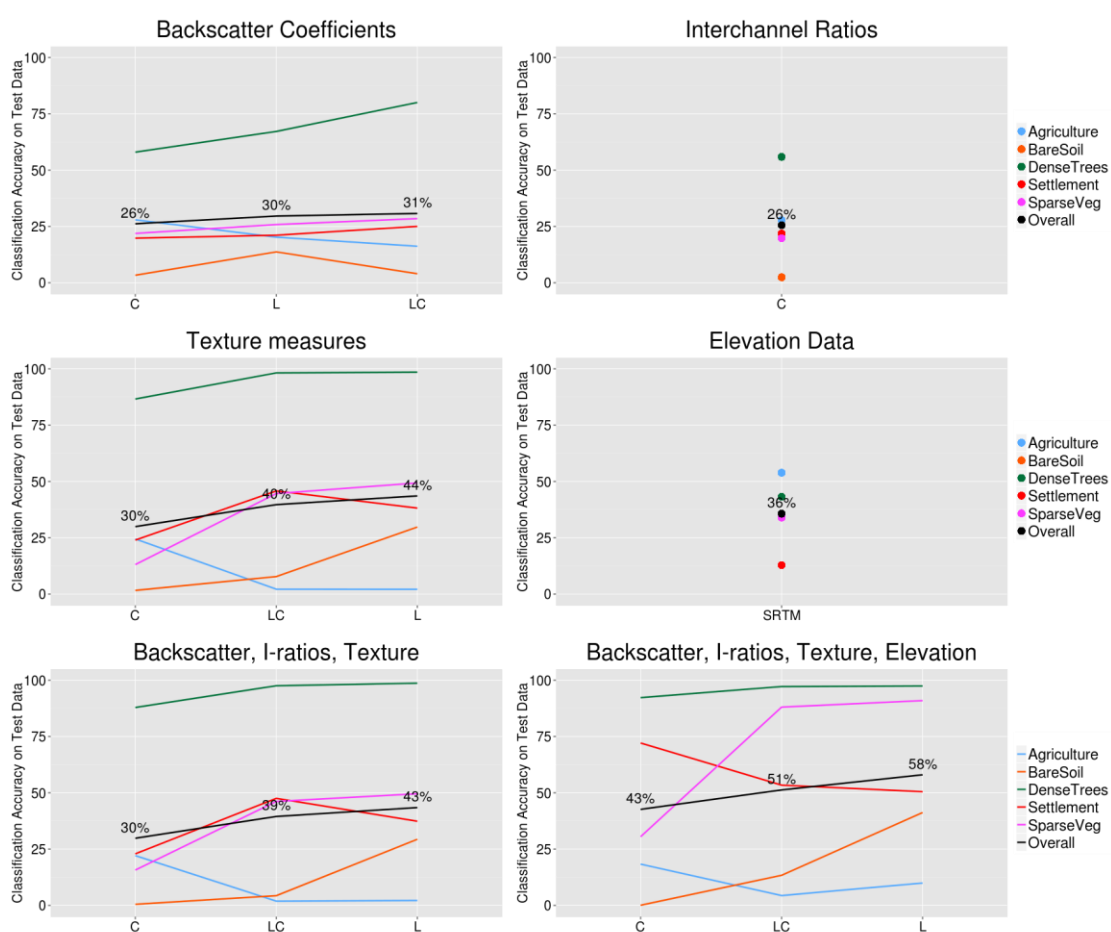


Figure 6.5 Classification accuracies of random forest models based on Tanzania dry season data, applied directly to Tanzania wet season data. Several combinations of input variables for the various frequency combinations are shown.

Figure 6.6 shows the overall and class-specific classification accuracies for the random forest models built on the Tanzania dry season training data and applied to the Tanzania wet season training data, using the site-specific land cover classes.

Classification accuracies of random forest models, built on Tanzania dry season data and applied to Tanzania wet season data, using site-specific classes

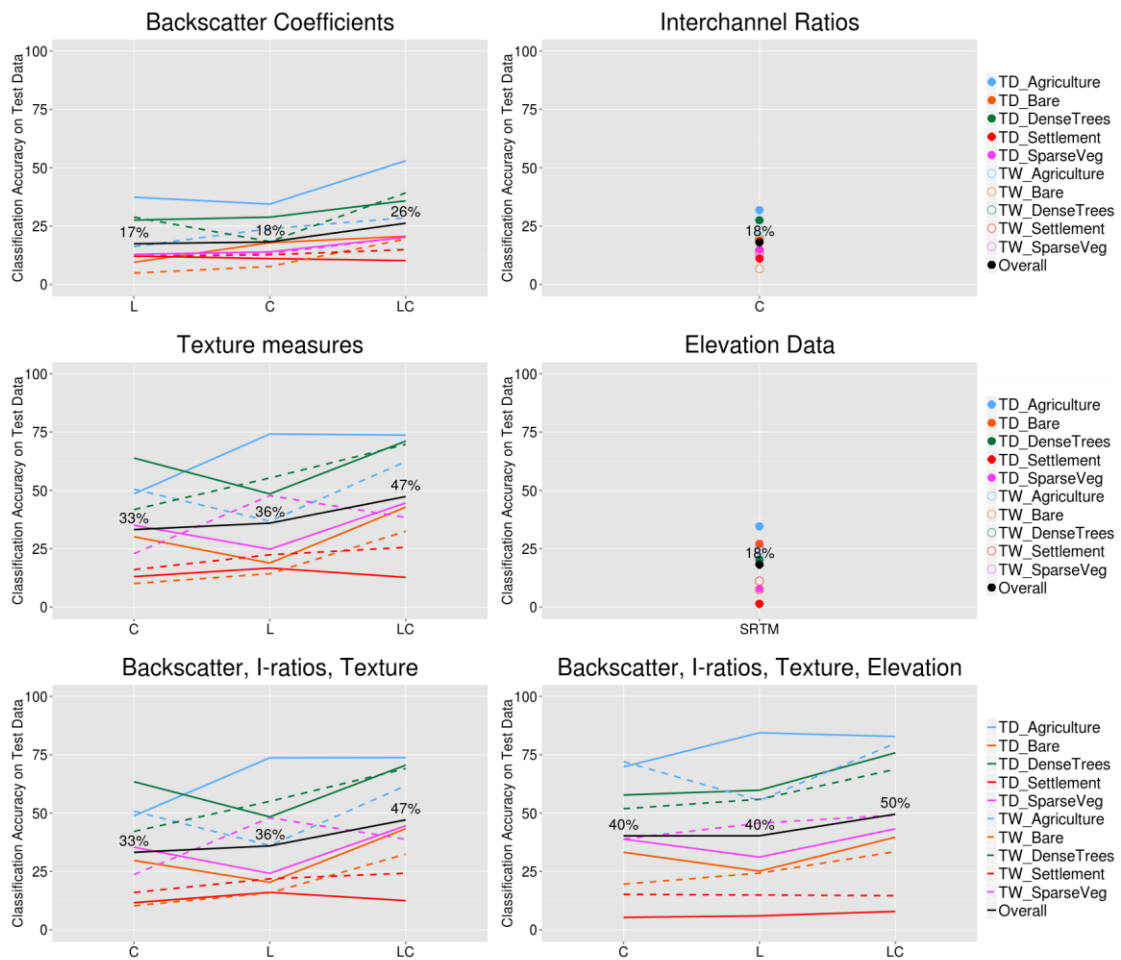


Figure 6.6 Classification accuracies of random forest models based on Tanzania dry season data, applied to Tanzania wet season data, with site-specific classes. Several combinations of input variables for the various frequency combinations are shown.

### 6.3.2.2 Tanzania dry season random forest model applied to Chad dry season data

Figure 6.7 shows the overall and class-specific classification accuracies for the different input variable scenarios for the random forest models trained on Tanzania dry season training data and applied directly on the Chad dry season training data, using the same land cover classes. Since only L-band data are available for the Chad dry season site, only L-band data could be used as part of the random forest models.

Classification accuracies of random forest models, built on Tanzania dry season data and applied directly to Chad dry season data using the same classes

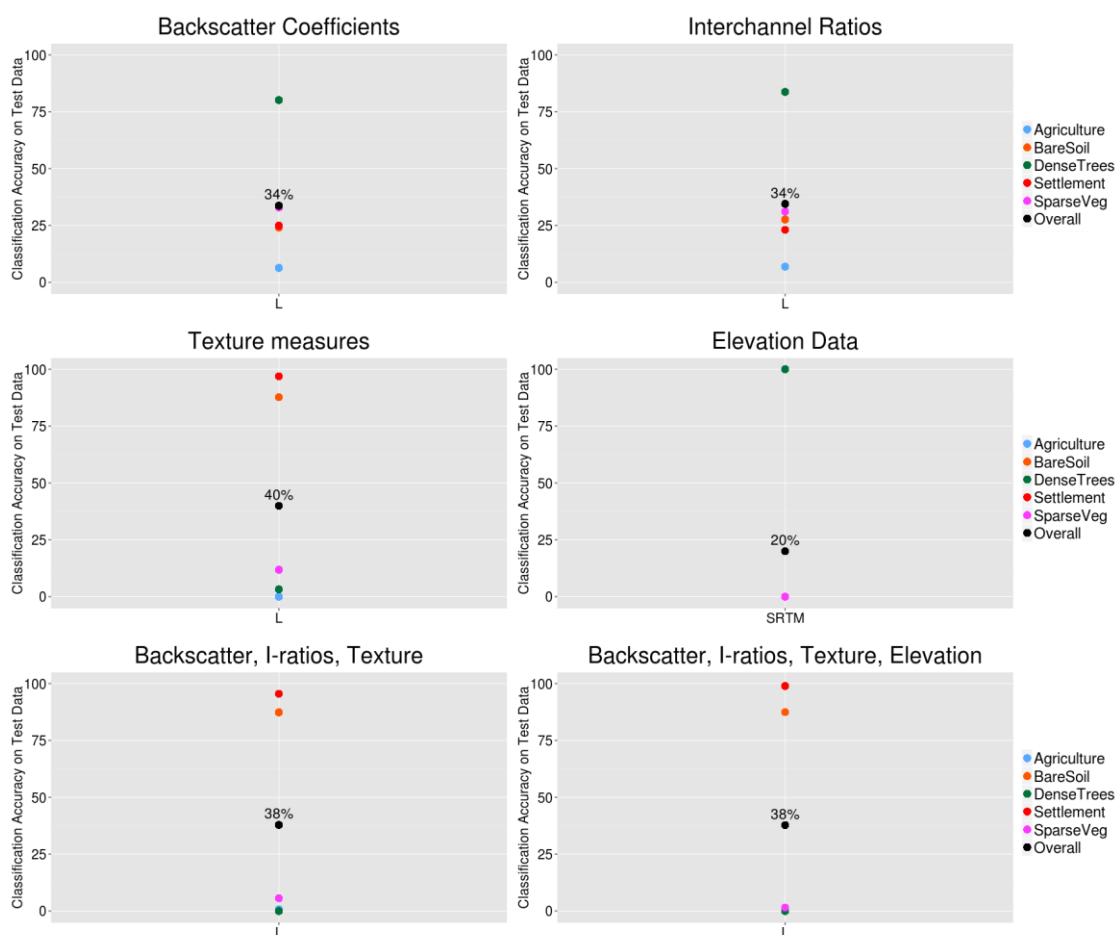


Figure 6.7 Classification accuracies of random forest models based on Tanzania dry season data, applied directly to Chad dry season data. Several combinations of input variables are shown. Only L-band data overlapped between the two sites.

Figure 6.8 shows the overall and class-specific classification accuracies for the random forest models built on the Tanzania dry season training data and applied to the Chad dry season training data, using the site-specific land cover classes.

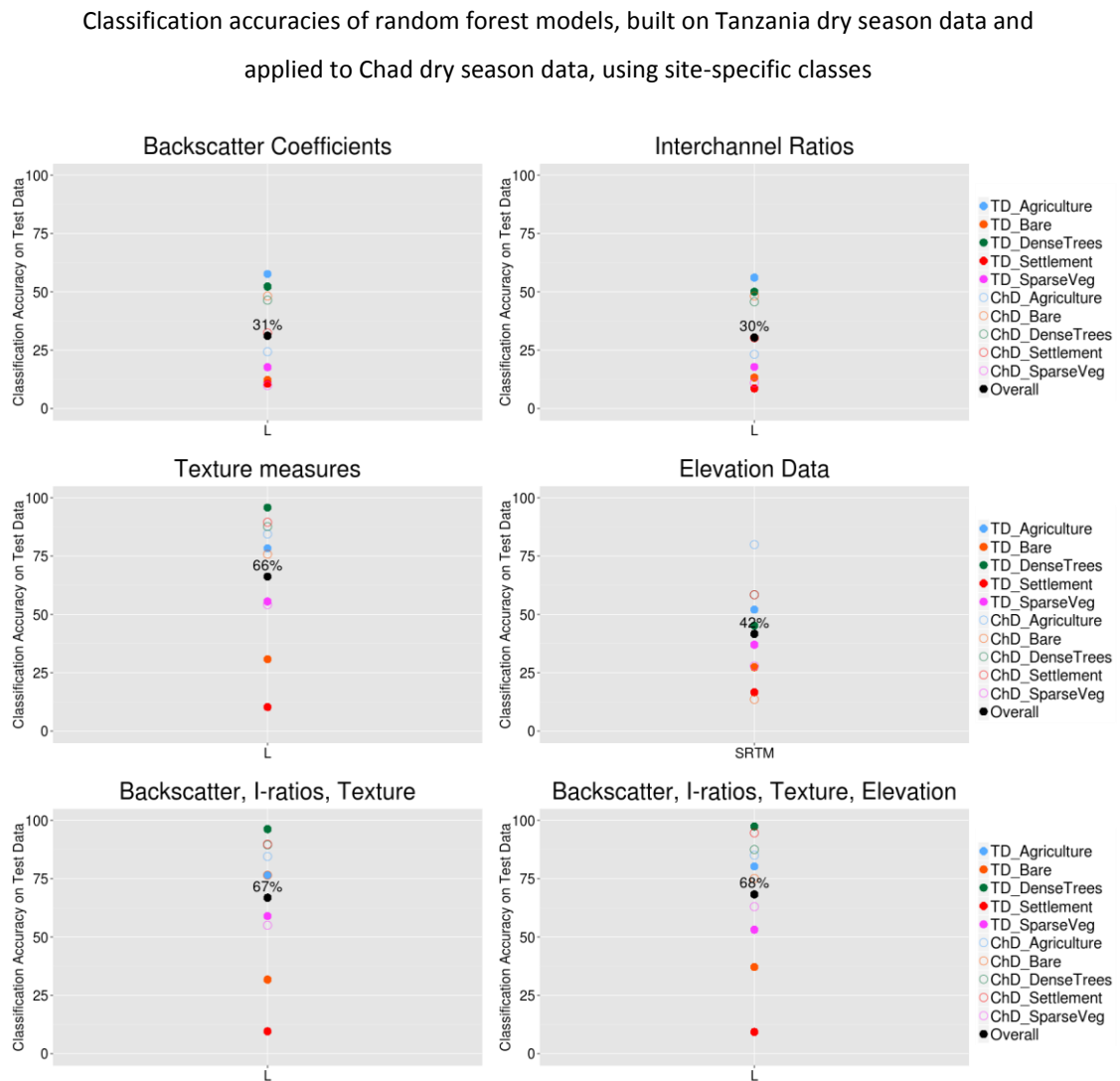


Figure 6.8 Classification accuracies of random forest models based on Tanzania dry season data applied to Chad dry season data, with site-specific classes. Several combinations of input variables are shown. Only L-band data overlapped between the two sites.

## 6.4 Summary of results

The results of the models which test the transferability of the forested sites are summarized in Table 6.3a for the single frequency SAR random forest models and in Table 6.3b for the multi-frequency SAR random forest models. Additional graphical representations of the same results summary are given in Figure 6.9 for the forested single frequency models and in Figure 6.10 for the forested multi-frequency models. The transferability results for the semi-arid sites are summarised in Table 6.4a for the single frequency SAR random forest models and in Table 6.4b for the multi-frequency SAR random forests. Similarly to the forested models, additional graphical representations of the same results are given in Figure 6.11 for the semi-arid single frequency models and in Figure 6.12 for the semi-arid multi-frequency models. In Table 6.3 and Table 6.4, the intensity of the colours represents the level of the overall classification accuracies, with higher classification accuracies showing as more intense colours. The results are summarised showing the model with the highest overall classification accuracy (1<sup>st</sup>), then the second highest (2<sup>nd</sup>) and then the third highest/ lowest (3<sup>rd</sup>) overall classification accuracy for each group.

The random forest models trained and tested on the Cameroon wet season data are referred to the forested ecoregion baseline models. In Table 6.3, the classification accuracy results from the random forest models trained on the Cameroon wet season data and applied on the Cameroon dry season (see Figure 6.1 and Figure 6.2) and the DRC wet season data (see Figure 6.3 and Figure 6.4) are compared to the forested baseline models (as marked with a \* in Table 6.3 and presented in Figures 5.1 to 5.6). The model trained and tested on the Cameroon dry data are given for reference.

The random forest models trained and tested on the Tanzania dry season data are referred to the semi-arid ecoregion baseline models. In Table 6.4 the classification accuracy results from the random forest models trained on the Tanzania dry season data and applied on the Tanzania wet season (see Figure 6.5 and Figure 6.6) and the Chad dry season data (see Figure 6.7 and Figure 6.8) are compared with the semi-arid baseline

models (as marked with a \* in Table 6.4 and presented in Figures 5.1 to 5.6). The Tanzania wet season model is given for reference.

The comparisons of the transferability of the random forest models for both the forested results (in Table 6.3 and Figures 6.9 – 6.10) and semi-arid results (in Table 6.4 and Figures 6.11 – 6.12) results are grouped by an increasing level of variable complexity. The groups are backscatter coefficients only (from Figure 5.2), interchannel ratios only (from Figure 5.3), elevation data only (from Figure 5.5), texture measures only (from Figure 5.4), backscatter coefficients, interchannel ratios and texture measures (from Figure 5.6), and backscatter coefficients, interchannel ratios, texture measures and elevation data (from Figure 5.1).



Table 6.3 Summary table of the overall classification accuracies from the random forest models, which test the transferability of the forested sites. The (a) single frequency and (b) dual / three frequency combinations are shown.

a) Forested single frequency

Site comparison	Land cover	Fig.	dB only (5.2)			i only (5.3)			SRTM	t only (5.4)			dB + i + t (5.6)			dB + i + t + e (5.1)		
			1st	2nd	3rd	1st	2nd	3rd		1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd
Cameroon dry, training vs test	same	5.x	58	44	39	58	44	-	56	91	71	61	91	71	61	92	76	71
Cameroon wet, training vs test *	same	5.x	48	48	38	48	47	38	57	73	69	52	74	69	52	78	77	63
Cameroon wet vs dry	same	6.1	51	48	41	50	48	-	58	64	51	41	74	54	41	74	63	43
	site-specific	6.2	47	29	23	47	29	-	29	84	65	61	84	65	61	84	74	68
Cameroon wet vs DRC wet	same	6.3	33	27	22	27	-	-	21	25	22	6	26	22	16	26	18	14
	site-specific	6.4	39	36	30	38	-	-	42	77	75	54	78	75	54	82	76	60

min	max
L	L
C	C
X	X
elev	elev

b) Forested dual/ three frequencies

Site comparison	Land cover	Fig.	dB only (5.2)				i only (5.3)				t only (5.5)				dB + i + t (5.6)				dB + i + t + e (5.1)			
			LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd
Cameroon dry, training vs test	same	5.x	66	63	62	52	-	63	-	-	91	92	90	78	91	92	90	79	91	92	90	80
Cameroon wet, training vs test *	same	5.x	59	57	53	50	58	57	53	50	85	84	76	76	85	85	76	76	88	88	80	78
Cameroon wet vs dry	same	6.1	59	57	56	51	-	59	-	-	65	65	65	53	74	74	73	49	74	74	74	53
	site-specific	6.2	55	52	51	36	-	52	-	-	86	85	84	76	86	86	84	76	85	86	86	79
Cameroon wet vs DRC wet	same	6.3	32	33	29	23	-	-	-	-	25	26	25	6	26	26	26	16	26	26	26	18
	site-specific	6.4	63	57	52	49	-	-	-	-	87	86	81	74	87	86	82	74	87	86	83	75

dB	backscatter coefficients	min	max
i	interchannel ratios	LC	LC
t	texture measures	LX	LX
e	elevation data	CX	CX
	* baseline model	LCX	LCX

Table 6.4 Summary table of the overall classification accuracies from the random forest models, which test the transferability of the semi-arid sites. The (a) single frequency and (b) dual / three frequency combinations are shown.

a) Semi-arid single frequency

Site comparison	Land cover	Fig.	dB only (5.2)			i only (5.3)			e only (5.5)	t only (5.4)			dB + i + t (5.6)			dB + i + t + e (5.1)		
			1st	2nd	3rd	1st	2nd	3rd	SRTM	1st	2nd	3rd	1st	2nd	3rd	1st	2nd	3rd
Tanzania wet, training vs test	same	5.x	27	21	-	21	-	-	27	46	27	-	47	28	-	51	36	-
Tanzania dry, training vs test *	same	5.x	44	30	26	44	29	-	28	60	54	40	60	54	43	63	56	50
Tanzania dry vs wet	same	6.5	30	26	-	26	-	-	36	44	30	-	43	30	-	58	43	-
	site-specific	6.6	18	17	-	18	-	-	18	36	33	-	36	33	-	40	40	-
Tanzania dry vs Chad dry	same	6.7	34	-	-	34	-	-	20	40	-	-	38	-	-	38	-	-
	site-specific	6.8	31	-	-	30	-	-	42	66	-	-	67	-	-	68	-	-

min max  
L L  
C C  
X X  
elev elev

b) Semi-arid dual/ three frequencies

Site comparison	Land cover	Fig.	dB only (5.2)				i only (5.3)				t only (5.5)				dB + i + t (5.6)				dB + i + t + e (5.1)			
			LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd	LCX	1st	2nd	3rd
Tanzania wet, training vs test	same	5.x	-	32	-	-	-	-	-	-	-	49	-	-	-	49	-	-	-	52	-	-
Tanzania dry, training vs test *	same	5.x	49	48	45	35	-	47	-	-	63	64	61	51	64	64	62	52	65	65	64	56
Tanzania dry vs wet	same	6.5	-	31	-	-	-	-	-	-	-	40	-	-	-	39	-	-	-	51	-	-
	site-specific	6.6	-	26	-	-	-	-	-	-	-	47	-	-	-	47	-	-	-	50	-	-
Tanzania dry vs Chad dry	same	6.7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	site-specific	6.8	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

dB	backscatter coefficients	min	max
i	interchannel ratios	LC	LC
t	texture measures	LX	LX
e	elevation data	CX	CX
	* baseline model	LCX	LCX

### Forested ecoregion, single frequency

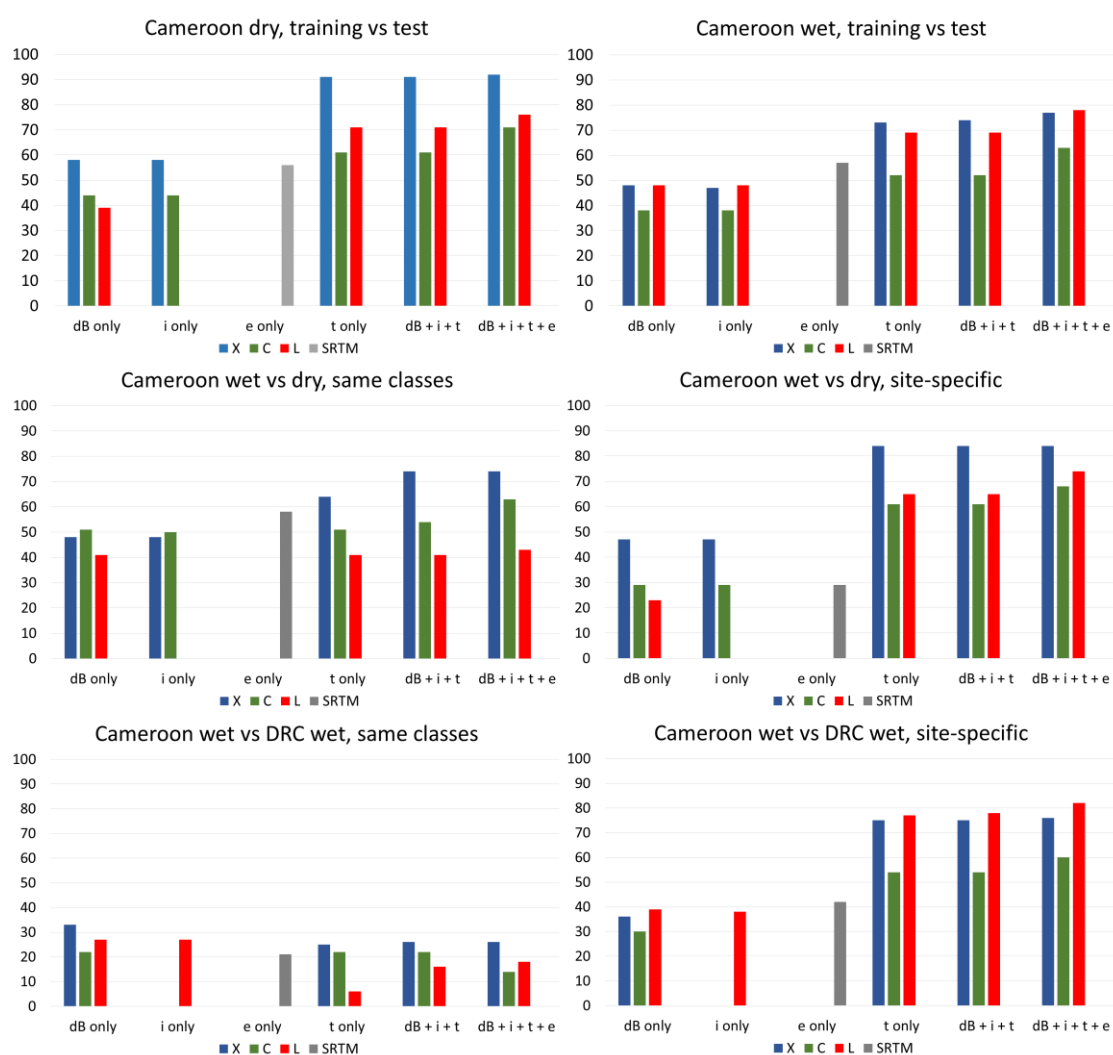


Figure 6.9 The overall classification accuracies from the random forest models, to test the transferability of the models for the forested sites, using single frequencies. This is the same data from table 6.3a represented as a bar charts.

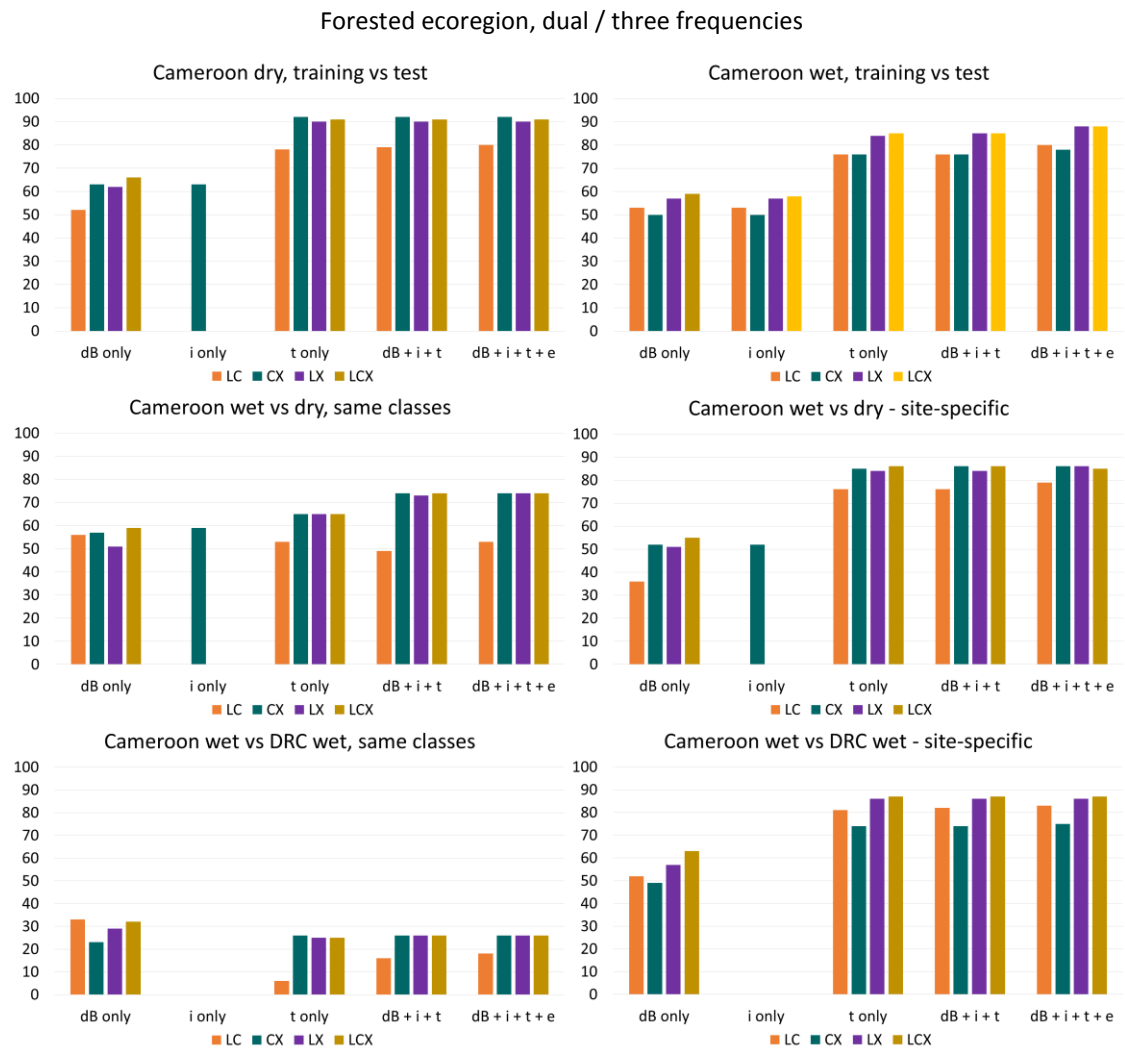


Figure 6.10 The overall classification accuracies from the random forest models, to test the transferability of the models for the forested sites, using dual/three frequencies. This is the same data from table 6.3b represented as a bar charts.

### Semi-arid ecoregion, single frequency

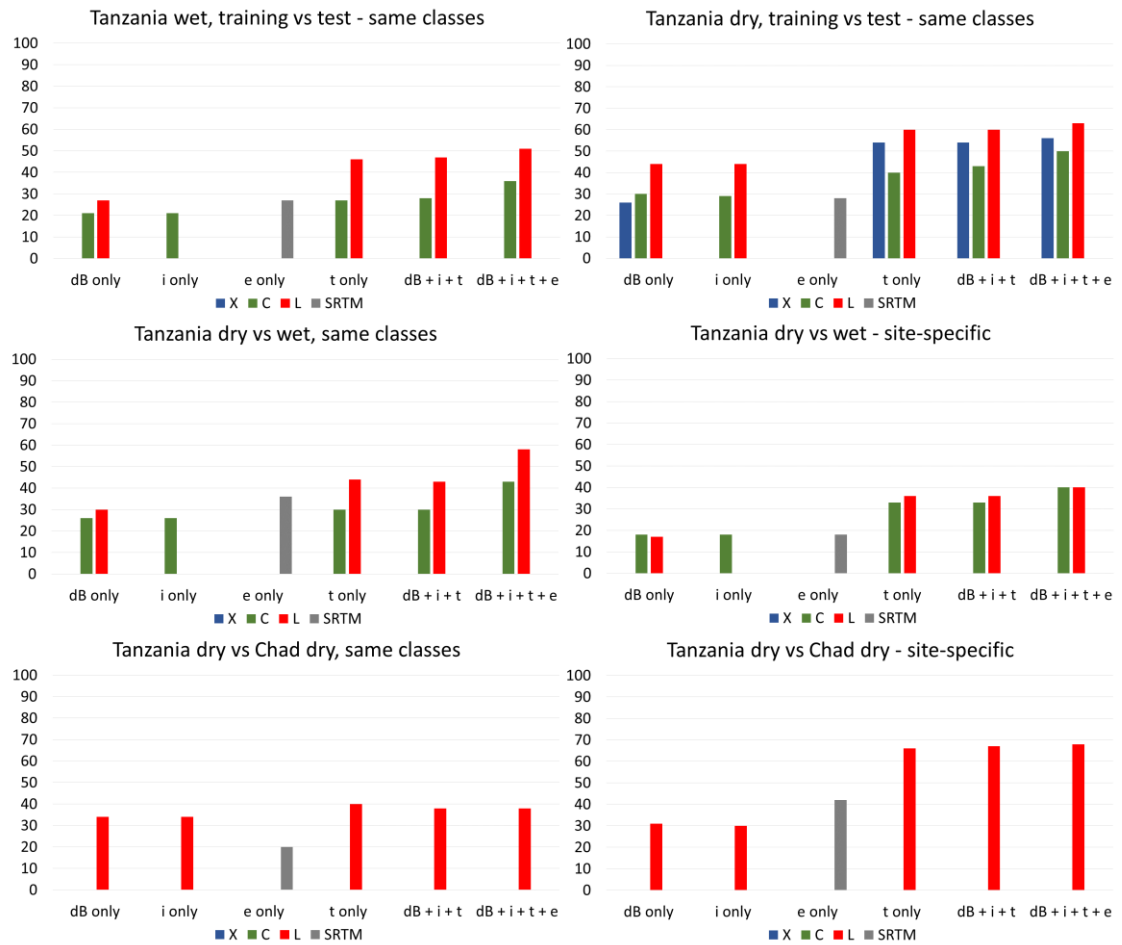


Figure 6.11 The overall classification accuracies from the random forest models, to test the transferability of the models for the semi-arid sites, using single frequencies. This is the same data from table 6.4a represented as a bar charts.

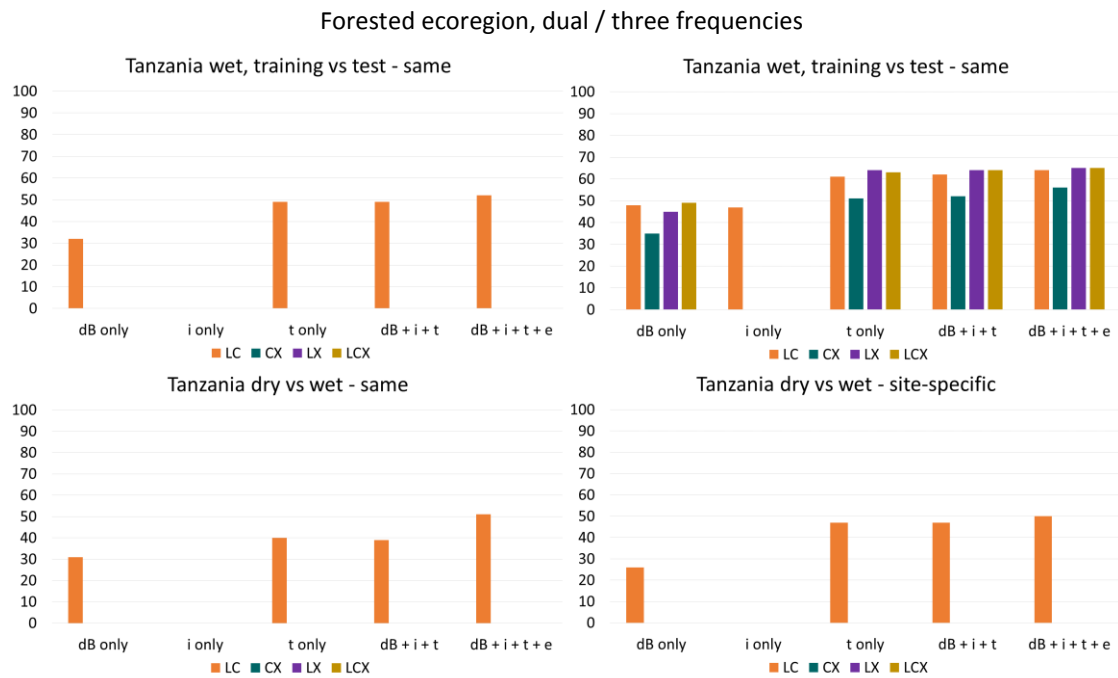


Figure 6.12 The overall classification accuracies from the random forest models, to test the transferability of the models for the semi-arid sites, using dual/three frequencies. This is the same data from table 6.4b represented as a bar charts.

## 6.5 Discussion

The results are next discussed from the following viewpoints: groups of increasing variable complexity (Section 6.5.1); single, double or three frequencies (Section 6.5.2); and forested vs semi-arid transferability comparison (Section 6.5.3).

### 6.5.1 Groups of increasing variable complexity

The groups of increasing variable complexity used to test the transferability of the random forest models are the focus for this discussion section. The groups of variables discussed are backscatter coefficients (Section 6.4.1.1), interchannel ratios (Section 6.4.1.2), elevation data (Section 6.4.1.4) and texture measures (Section 6.5.1.3).

#### 6.5.1.1 Backscatter coefficients

Using only backscatter coefficients produced results much lower than using all the available variables and specifically any model with the addition of texture measures. For the forested sites, the Cameroon wet season model applied to the Cameroon dry season using the same land cover classes was similar to the baseline model (59% for LCX models). The Cameroon wet season models applied to dry season data, perform marginally lower to the Cameroon dry season models applied to dry season data (59% compared to 66% for the LCX, backscatter coefficient only models).

For the semi-arid sites the inter-season comparison transfer of the Tanzania dry season model to the Tanzania wet season data does worse than the Tanzania dry season baseline model (31% vs 48% for the LC, backscatter coefficient only models).

The inter-site comparison for the forested ecoregion using only backscatter coefficients performed slightly better than the other variable combinations that include the additional feature layers. This was unexpected, since all the other scenarios have higher classification accuracies when additional layers are included. However, the overall classification accuracies of the forested inter-site comparison were very low at 33% for the LC model or 33% using X-band only. This is higher than the LCX-bands scenario using all the variables at 26% overall classification accuracy.

The inter-site comparison for the forested ecoregion using site-specific land cover class scenarios resulted in an overall classification accuracy of 63% for the LCX backscatter coefficient only model, which is higher than the LCX baseline model with an overall classification accuracy of 59%.

The inter-site comparison for the semi-arid ecoregion using only backscatter coefficients could only be assessed with the L-band data (see Section 3.2). The classification accuracy was lower, at 34% using the same land cover classes and 31% using site-specific land cover classes, compared to the baseline model with 44% overall classification accuracy.

### 6.5.1.2 Interchannel ratios

The interchannel ratios can only be calculated when either a dual polarized or quad polarized image are available. Only a small number of interchannel ratios are available on the inter-season and inter-site comparisons. When the interchannel ratios are available, the overall classification results are either the same or slightly lower than the overall classification accuracies using backscatter coefficients only.

### 6.5.1.3 Elevation data

Using only the elevation data, which includes the elevation, slope and aspect data as discussed in Section 4.3.3, gave higher overall classification accuracies than expected. This can be explained by a topographical relationship between land cover and elevation in each site. This is especially the case for the Cameroon site, which has a relatively rugged terrain, and with land cover classes following the terrain pattern. The dense trees are most often in the valleys, and open woodland more prevalent on the hills. The lower planes are more prevalent to sparse vegetation and bare soil, closer to the river. As a result, the elevation data serves as good indication of land cover class (56% classification accuracy for Cameroon dry and 57% classification accuracy for Cameroon wet training vs test data). The Tanzania site is much flatter, and as a result the relation between elevation and land cover class was less pronounced (27% classification accuracy for Tanzania wet and 28% classification accuracy for Tanzania dry, training vs test datasets).

For the inter-season comparison, the elevation data models gives overall classification accuracies higher than the backscatter coefficient only models using the best single frequencies (58% for Cameroon wet vs dry elevation only model compared to 51% for Cameroon wet vs dry C-band backscatter coefficient only model; and 36% for Tanzania dry vs wet elevation only model and 30% for Tanzania dry vs wet L-band backscatter coefficient only model). The inter-season comparison using the site-specific land cover class options performed worse than directly applying the same land cover classes (29% overall classification accuracy for the Cameroon wet vs dry site-specific land cover classes vs 58% for applying the land cover classes directly; and 18% overall classification



accuracy for the Tanzania dry vs wet site-specific land cover classes vs 36% for applying the land cover classes directly).

There was a large enough difference in the elevation between the two sites for both the forested and semi-arid sites, to distinguish the site-specific land cover classes to reasonable degree of accuracy while only using the elevation data. The elevation data gave an overall classification accuracy of 42% for the Cameroon wet season vs DRC wet season for the site-specific land cover class models, which is higher than the site-specific single frequency (L-band) backscatter coefficient only model with 39% overall classification accuracy. Using the same land cover classes only gave 21% overall classification accuracy. For the semi-arid inter-site comparison, the elevation data only gave 42% overall classification accuracy using the site-specific classes and only 20% using the same land cover classes.

#### 6.5.1.4 Texture measures

It is noted that the addition of texture measures to any of the models lead to a large increase in overall classification accuracies, for both ecoregions and for all frequency combinations, inter-season and inter-site comparisons, with the exception of the Cameroon wet season model applied to the DRC wet season data using the same land cover classes.

The texture measures only model for the forested sites using LCX-band for the inter-season comparison, gives an overall classification accuracy of 65% applying the same land cover classes directly. This is 20% lower than the LCX texture measure only baseline model with 85% overall classification accuracy. The inter-season and inter-site LCX texture measures only models, perform slightly higher at 86% and 87% respectively, compared to the baseline model at 85% overall classification accuracy.

The semi-arid inter-site comparison of the LC-band texture measures only model is lower at 40% overall classification accuracy using the same land cover classes and 47% using site-specific land cover classes, compared to the LC-band baseline model of 61%.

The semi-arid inter-site comparison could only be undertaken with L-band data, and the Tanzania dry season model applied on Chad dry season data gave 40% overall classification accuracy using the same land cover classes, which is lower than the baseline model of 60%. Using the site-specific land cover classes, the L-band model of the Tanzania dry model applied on the Chad dry data gave an overall classification accuracy of 66%, compared to the baseline model with 60% overall classification accuracy.

The models with more additional variables than the texture measures only scenario (dB + i + t and dB + i + t + e), resulted in overall classification accuracies slightly higher or very similar to the models using only texture measures. This shows that the texture measures has the largest contribution to the model, and that the additional layers or backscatter coefficients, interchannel ratios and elevation data, only has a marginal additional contribution to the models for both ecoregions and for the inter-season and inter-site comparisons.

For the second and third best frequency choices for the single frequency scenarios, the addition of elevation data to the model of backscatter coefficient, interchannel ratios and texture measures, does lead to a substantial increase in overall classification accuracy for both the forested and semi-arid models. This can therefore be a relatively easy way to increase classification accuracy results when only single frequency data are available. For the dual-frequency and three-frequency scenarios, the addition of elevation data does not lead to the same level of overall classification accuracy increase.

### 6.5.2 Single, double or three frequencies

For each site comparison scenario, recommendations are made for both single frequency and dual / three frequency options in Table 6.5. This is achieved by taking the highest overall classification accuracies from the single frequency and dual / three frequency scenarios in Table 6.3 and Table 6.4. For the forested sites, the recommended single frequency is X-band. The main dual frequency recommendation for the forested sites are either LX-band or CX-bands. The LX-band recommendation is consistent for all the forested inter-season and inter-site scenarios. The recommended single frequency for the inter-season and inter-site model transfers for the semi-arid ecoregion is L-band, and the recommended dual-frequency is LC-bands. For the semi-arid inter-season comparison, only L-band and C-band data were available and for the semi-arid inter-site comparison, only L-band data were available (see Section 3.2). For this reason, the discussion only includes L-band and C-band for the semi-arid inter-season scenario, and L-band for the semi-arid inter-site scenario.

It is noteworthy that the single frequencies give overall classification accuracies very close to the dual-frequency scenarios. It is also surprising that the best dual-frequency scenarios gave overall classification accuracies at the same level to the LCX models, and the three-frequency model gave higher overall classification accuracy only for one scenario (i.e. for the Cameroon wet season model applied to the DRC wet season data using site-specific land cover classes).

Table 6.5 Summary of recommended frequency and variable recommendations for transferring random forest models to different season and additional site scenarios. Overall classification accuracies are given in brackets for the respective scenarios.

Model transfer site scenario	Land cover type	Recommended single-frequency		Recommended dual/three frequencies	
		Band	Variables used	Bands	variables
<i>Forested inter-season</i> (Cameroon wet vs dry )	same	X	All variables (74%)	LX / CX	All variables (74%)
	site-specific	X	Texture measures only (84%)	LX / CX	All variables (86%)
<i>Forested inter-site</i> (Cameroon vs DRC, wet)	same	X	Backscatter coefficients only (33%)	LX / CX	All variables (26%) (lower than single frequency models, therefore rather use X-band only)
	site-specific	L	All variables (82%)	LX LCX	Texture measures only (86%) Texture measures only (87%)
<i>Semi-arid inter-season</i> <sup>(1)</sup> (Tanzania dry vs wet)	same	L	Texture measures only (58%)	LC	All variables (51%)
	site-specific	L / C	All variables (40%)	LC	All variables (50%)
<i>Semi-arid inter-site</i> <sup>(2)</sup> (Tanzania vs Chad, dry)	same	L	Texture measures only (40%)	N/A (only L-band available)	
	site-specific	L	All variables (68%)	N/A (only L-band available)	

<sup>(1)</sup>only L-band and C-band data available (no X-band) for semi-arid inter-season comparison

<sup>(2)</sup>only L-band data available (no C-band or X-band) for semi-arid inter-site comparison

If only single frequency data are available, it is recommended to add SRTM elevation data to increase the classification accuracy with relative ease. When two or three frequency data are available, the addition of elevation data only has a marginal effect. It is highly recommended to use texture measures for both inter-site and inter-season transfer of the models, regardless of whether single or dual/three frequency data are used. This applies to both the forested and semi-arid ecoregions.

### 6.5.3 Forested vs semi-arid transferability comparison

The inter-season transfer of the random forest models for the Cameroon site, using the same land cover classes, gave reasonably high overall classification accuracies, with the highest overall classification accuracy from the LCX or CX models at 74%, compared to 88% for the LCX and CX baseline models. The forested inter-season comparison using site-specific land cover classes was much closer to the baseline models, with the highest overall classification accuracy of 85% using LX-bands compared to the 88% of the LX-bands baseline model. It can therefore be deduced that the land cover classes between the Cameroon dry and wet season sites are sufficiently different for the random forest model to be able to distinguish a meaningful difference, and therefore classify the site-specific land cover classes correctly. The Cameroon wet and dry season data are however similar enough for the model trained on the wet season data to be applied to the dry season data with still a relatively high level of overall classification accuracy (see Section 7.3.1.2).

The highest overall classification accuracies for the semi-arid inter-season comparison are 51%, using the same land cover classes, and 50%, using the site-specific land cover classes using the LC-bands model with all the variables. This is in comparison to the LC-bands baseline model using all the variables with 65% overall classification accuracy. The overall classification accuracies of the same, and site-specific, land cover classes are therefore much closer to each other for the semi-arid site, than for the forested site. For the semi-arid dry season and wet season data, it can be deduced that the land cover

classes are quite similar between the wet and dry season samples (see also Section 7.3.2.2).

The forested and semi-arid inter-site comparisons using the same land cover classes did not perform very well, and much lower than expected. The highest overall classification accuracy for the forested inter-site comparison was 33%, using X-band data with only the backscatter coefficients. The highest overall classification accuracy for the semi-arid inter-site comparison was 40%, using L-band data with only texture measures. This is much lower than the inter-site comparisons using the site-specific classes, where the forested sites gave 87% for the LCX-bands model using all variables (compared to 88% for the LCX baseline model with all the variables), and 68% for the semi-arid inter-site comparison using L-band data with all the variables (compared to 63% for the L-band baseline model with all the variables). There appears to be a substantial difference between the land cover classes of the Cameroon wet and DRC wet study sites (see Section 7.3.1.3), and also between the land cover classes of the Tanzania dry season and Chad dry season study sites (see Section 7.3.2.3).

It should be noted that the overall classification results in Chapter Six are not 100% comparable to the overall classification accuracies presented in Chapter Five, since Chapter Five included the *open woodland* class. The *open woodland* class was not present in either of the additional sites (i.e. neither in the DRC site for the forested ecoregion, or for the Chad site for the semi-arid ecoregion). The *open woodland* class varied between the third or fourth highest classification accuracy out of the six classes for the Cameroon and Tanzania sites. The overall classification accuracies of all the inter-season and inter-site scenarios presented in Chapter Six, are therefore slightly higher, compared to when *open woodland* would have been included. However, the overall conclusions should remain the same, regardless of whether the *open woodland* class is included or not.

## 6.6 Conclusion

In this chapter it is shown that the random forest models are transferable from the wet to the dry season for the forested sites, and from the dry to the wet season for the semi-arid sites, while still maintaining a reasonable level of classification accuracy. The algorithms should be transferable, with the assumptions that the following conditions remain similar between the training data and the new data on which the model is applied:

- land cover classes are similar
- acquisition conditions are similar, such as rainfall, vegetation water content and soil moisture
- elevation are at a similar altitude

Should these conditions not be similar, the model will not be transferable, or at least not to the same degree of accuracy as achieved on the training dataset. The exact limits and degree of acceptable change in the landscape parameters will still need to be determined, to be able to quantify whether the model is transferable or not to new data.

For the forested sites, the inter-season transfer decreases from 88% for the LCX baseline model to 74% for the LCX-model using all the available variables. For the semi-arid site, the overall classification accuracy decreases marginally from 63% to 58% using L-band models, using all the available variables.

For the inter-site model transfer, it is therefore recommended to rather collect samples from both sites, and train the models to incorporate the site-specific land cover classes. It appears that in both the forested and semi-arid inter-site comparisons, the land cover classes of the additional sites are substantially different, to such a degree that a direct transfer of the model using the same land cover classes is not feasible.

For the forested ecoregion the use of X-band is recommended for single frequency SAR land cover applications. For the semi-arid ecoregion, with the data available, L-band is recommended above C-band data for any inter-season land cover classification. Due to

the X-band orthorectification errors (see Section 4.1), no conclusion can be made with regards to X-band for semi-arid inter-season model transferability.

Using dual or three frequencies for the inter-season and inter-site transfer of the random forest models for the forested sites, the use of LX-band data gives consistently good results for both the same and site-specific land cover classes (with the exception of the Cameroon wet season vs DRC wet season data using the same land cover classes). The use of texture measures only give overall accuracies close to using all the feature variables.

The summary tables (Table 6.3, 6.4 and 6.5) in this Chapter can be used by the remote sensing community as a reference of which frequency combinations are recommended for the forested and semi-arid ecoregions. The summary tables can also give an indication of expected overall classification accuracies for the transfer of the models to both additional season and additional site data, when similar data to that which was used in this study is available.



## **Chapter Seven SAR signatures and Kullback-Leibler divergence measures of multi-frequency SAR over semi-arid and forested sites**

To enable the application of a model trained on one dataset with known outcomes on a new dataset with unknown outcomes, one of the fundamental questions to ask is whether the new dataset land cover classes have similar statistical distributions for the dataset variables as the original training dataset land cover classes? If the new dataset does have similar statistical distributions, then it can be assumed that the model will be reliable and give reliable predictions on the new data. However, if the statistical distributions are remarkably different, then the model will not give reliable results. In the case of working with satellite imagery, as in this study, a land cover classification model is often trained on a set of satellite imagery that is acquired of a specific geographical location with specific climatic conditions. A model trained and developed on one set of satellite images can most reliably be applied on another set of data that is of the same, or similar, geographical location and similar climatic conditions. In particular, the specific scattering mechanisms related to the specific vegetation type, such as the land cover classes of sparse vegetation or dense trees, should be similar for the model to be applicable to additional data.

To address the third and final hypothesis, the statistical distributions of the SAR backscatter coefficients and additional calculated layers were compared for different land cover classes. The comparison was performed first visually and secondly with a statistical measure to compare the distance between distributions. The comparison was undertaken not only between the land cover classes of the same site, but also between the land cover classes of the wet and dry seasons, as well as between the main development site and the additional site for both ecoregions. If the distributions are similar, then the model can be applied to the new data. However, if the distributions are different, then the model is not applicable to the new data and should be used with caution. To determine an acceptable level of deviation between the statistical

distributions of the training dataset and the new dataset, is part of what this chapter aims to address.

## 7.1 Introduction

There are several measures to compare statistical distributions with each other, known as separability indices. A selection of measures are the Fisher criterion, the Bhattacharyya distance and the Kullback-Leibler divergence (KLD). These separability indices have all been applied to separating land cover classes within SAR images in the literature. The Fisher criterion is used to compare texture measure separability of SAR images of sea-ice (Clausi, 2002), whereas the Bhattacharyya distance is used to compare class separability within SAR images of selected crop classes (Anys, 1995) and rangeland classes related to oil-fields (Kwarteng et al., 2008). The KLD is related to the relative entropy measure, which is the basis of the Maximum Entropy (MaxEnt) classifier (Elith et al., 2011; Phillips and Dudik, 2008). Additionally, the KLD has been used to compare texture measures within SAR images (Mathiassen et al., 2002) and to classify SAR imagery (Qin et al., 2015).

In this study, there are different scenarios where it is necessary to use a separability index to compare the statistical distributions of selected land cover classes:

- a) The comparison between the different land cover types of the training dataset.
- b) The comparison of the land cover types between the training data and the test data, from the same study site and season.
- c) The comparison of land cover types between the wet and dry seasons, but of the same geographical locations, also referred to as an inter-season comparison.
- d) The comparison of land cover types of the same season, but of different geographical locations, also referred to as an inter-site comparison.

For scenario (a), the separability index needs to be as high as possible, to ensure that the land cover classes are distinguishable and that it is possible to classify the land cover classes successfully. This scenario is not covered in this chapter, since it is a feature

selection application of a separability index, and in this project this is done by the variable importance measure as part of the random forest classification in Section 5.3.

In scenario (b), the separability index needs to be low to ensure that the test data are similar to the training data, although not always exactly the same. The reason for having a separate test dataset is to ensure that the model can generalise to a reasonable extent, with new data provided.

For scenario (c) and (d) the separability index is a test to assess whether the model developed on the training dataset is still relevant for application on new data. Scenario (c) compares the wet and dry season of the same geographical area (inter-season comparison), whereas scenario (d) compares study sites that are in different geographical locations but of similar ecological conditions (inter-site comparison). The reason for comparing the statistical distributions of the land cover classes in scenario (d) is that although land cover classes can look similar using visual interpretation of optical images, the SAR backscatter coefficients and additionally calculated layers such as interchannel ratios, texture measures and elevation data can be quite different. In cases when the distributions of the SAR backscatter coefficients are too different, it will be unsuitable to use the same model developed on one geographical location, on another dataset of a different geographical location. For the data used specifically in this study, the distribution comparisons are detailed as in Table 7.1.

*Table 7.1 Scenarios for assessing the separability between land cover classes.*

<b>Scenario</b>	<b>Comparison</b>	<b>Forested site</b>	<b>Semi-arid site</b>
a)	Land cover classes separability	Output from RF variable importance in Section 5.3	Output from RF variable importance in Section 5.3
b)	Training vs test	CW-training vs CW-test	TD-training vs TD-test
c)	Wet vs dry	CW-training vs CD-training	TD-training vs TW-training
d)	Site 1 vs Site 2	CW-training vs DW-training	TD-training vs ChD- training

The data collected are only from a certain timestamp and is only a trace of the full phenological cycle of the land cover over an annual cycle. This should be kept in mind, as well as the fact that the data in this study are from a broad wet and dry season, and not collected from a detailed daily, weekly or monthly interval.

There is, therefore, a trade-off between the ideal sampling scenario compared to the real-world data availability scenario. The ideal sampling scenario would involve every land cover class in every biome collected daily (or perhaps hourly), simultaneously, by all possible SAR frequency-polarisation acquisition modes, with optical imagery and ground-truth data at exactly the same time as the SAR satellite data collection. The real world data availability scenario is only an approximation of the ideal sampling scenario. This should be kept in mind when attempting to apply these findings to newly acquired SAR imagery. This highlights why it is important not to overfit a model, and to ensure that the model generalizes sufficiently to be flexible when using new data to classify the land cover.

#### 7.1.1 Cameroon wet season training vs test samples

Figures 7.1 to 7.4 shows the data distributions for the Cameroon wet season training vs test land cover comparisons. It can be seen that the training and test samples are similar, which is as expected. These figures are drawn to get an insight to the data, and as a double check between the training and test sample distributions.

The backscatter coefficients are shown in Figure 7.1 for the CW-training vs CW-test samples. It can be seen that the training and test datasets are similarly distributed. It is also notable that apart from the *water* class, the statistical distribution of the remaining classes of *bare*, *settlement*, *sparse vegetation* and *dense trees* are largely overlapping for all the backscatter coefficients channels. This highlights the effectiveness of SAR data to be used for mapping water boundaries during flood events (Martinez and Le Toan, 2007; Cruz, 2010; Pulvirenti et al., 2011). This can be explained by the SAR signal being reflected away from the SAR sensor, due to the smooth water surface, and as a result,

has low backscatter coefficient values. The main challenge is therefore how to separate the different vegetation classes, which are currently largely overlapping with each other.

Figure 7.2 show the interchannel ratios for the CW-training vs test samples. Here all the land cover classes overlap with each other, and it confirms why the interchannel ratios did not rank high by the MDA and MDG variable importance measures in Section 5.3.2. Figure 7.3 shows the top 10 texture measures for the CW-training vs CW-test samples. Here the classes are more separable, and it becomes apparent why the variable importance measures showed that the texture measures are the most important variables. The elevation data in Figure 7.4 shows a slight variation between the training and test samples.

Cameroon wet season training vs test samples: backscatter coefficients

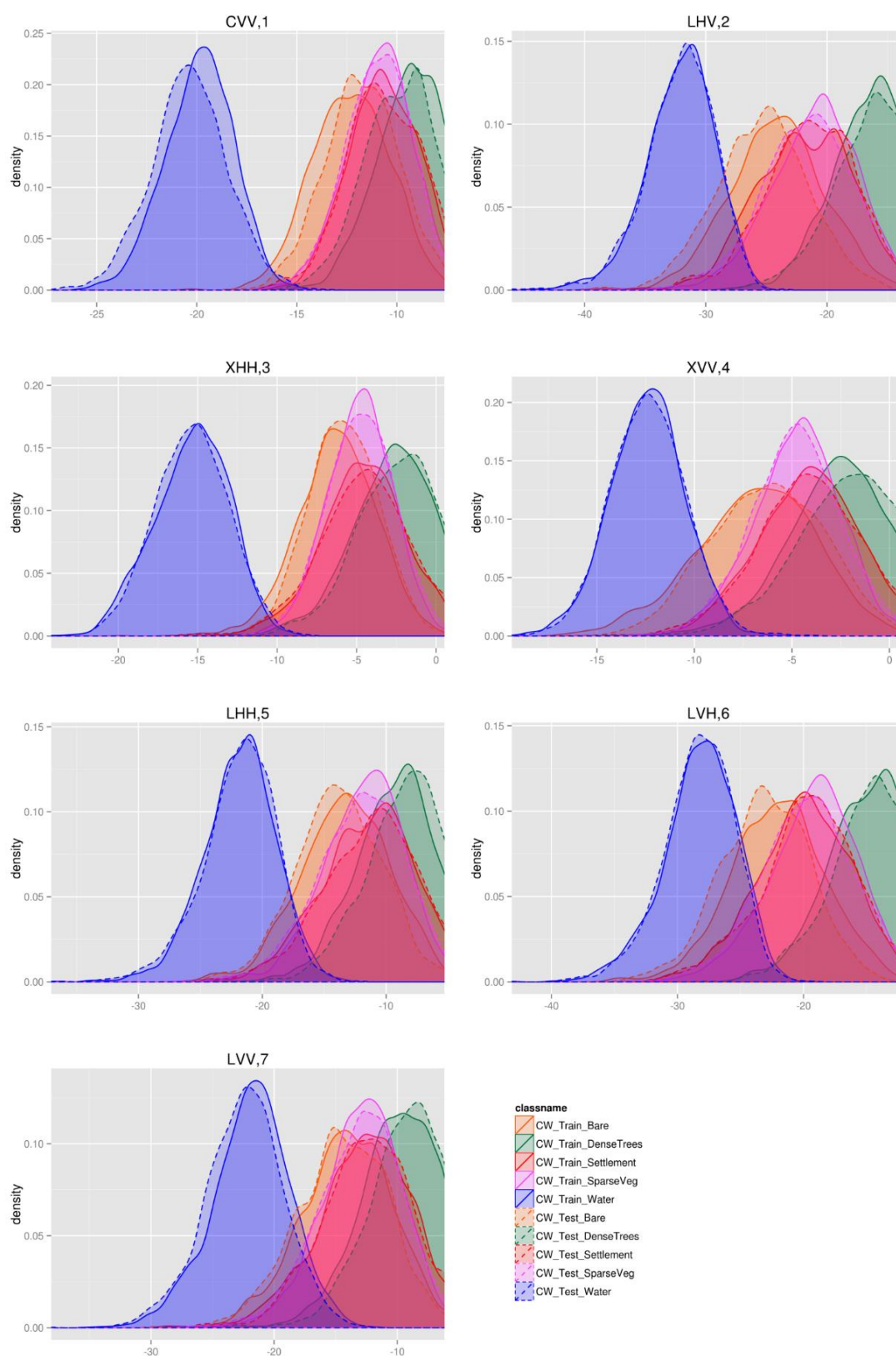


Figure 7.1 Distribution of backscatter coefficients for the Cameroon wet season training vs test samples. Order based on MDG variable importance of a random forest model of the backscatter coefficients.

## Cameroon wet season training vs test samples: interchannel ratios

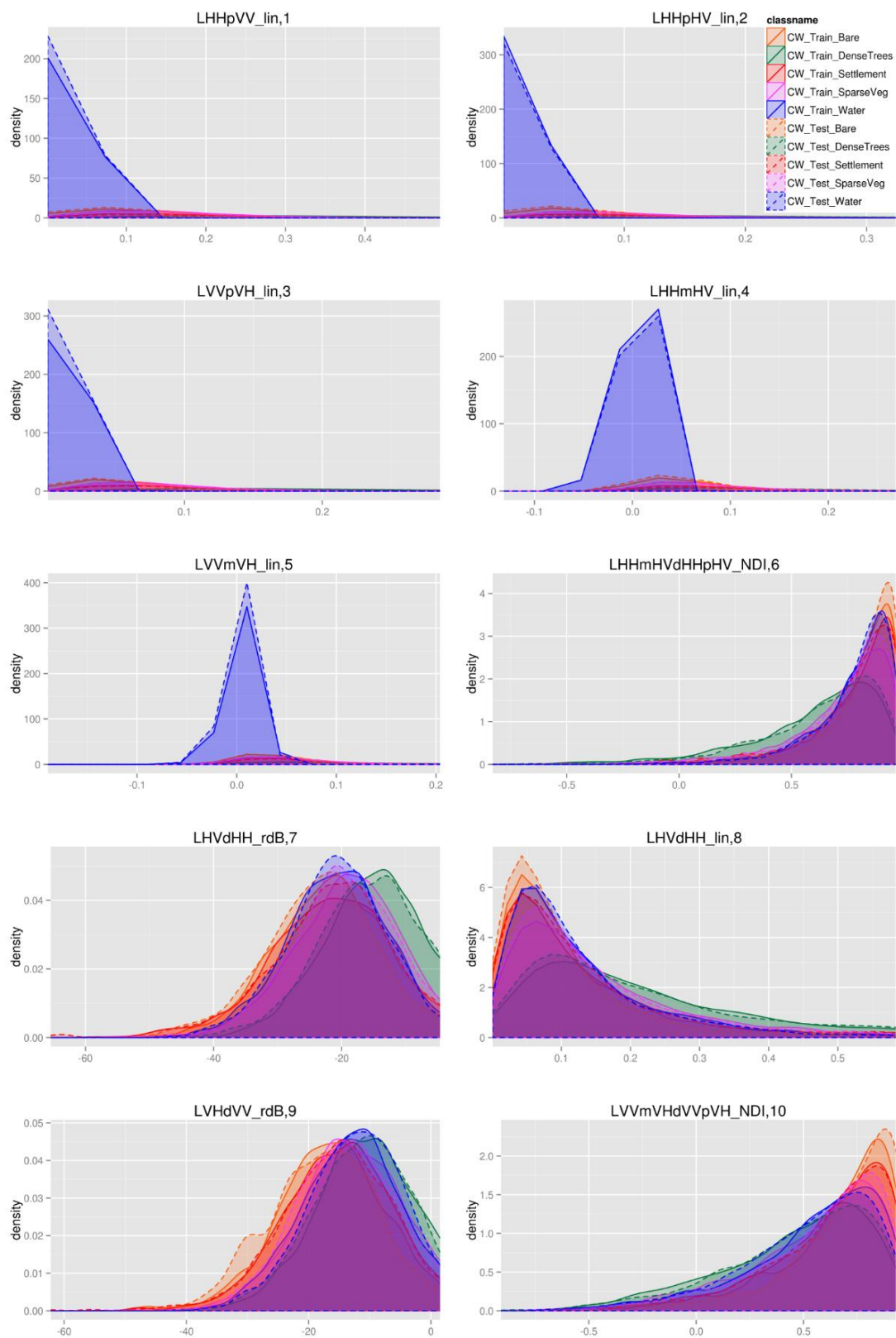


Figure 7.2 Distribution of the top 10 interchannel ratios for the Cameroon wet season training vs test samples. Order based on MDG variable importance of a random forest model of the interchannel ratios.

### Cameroon wet season training vs test samples: texture measures

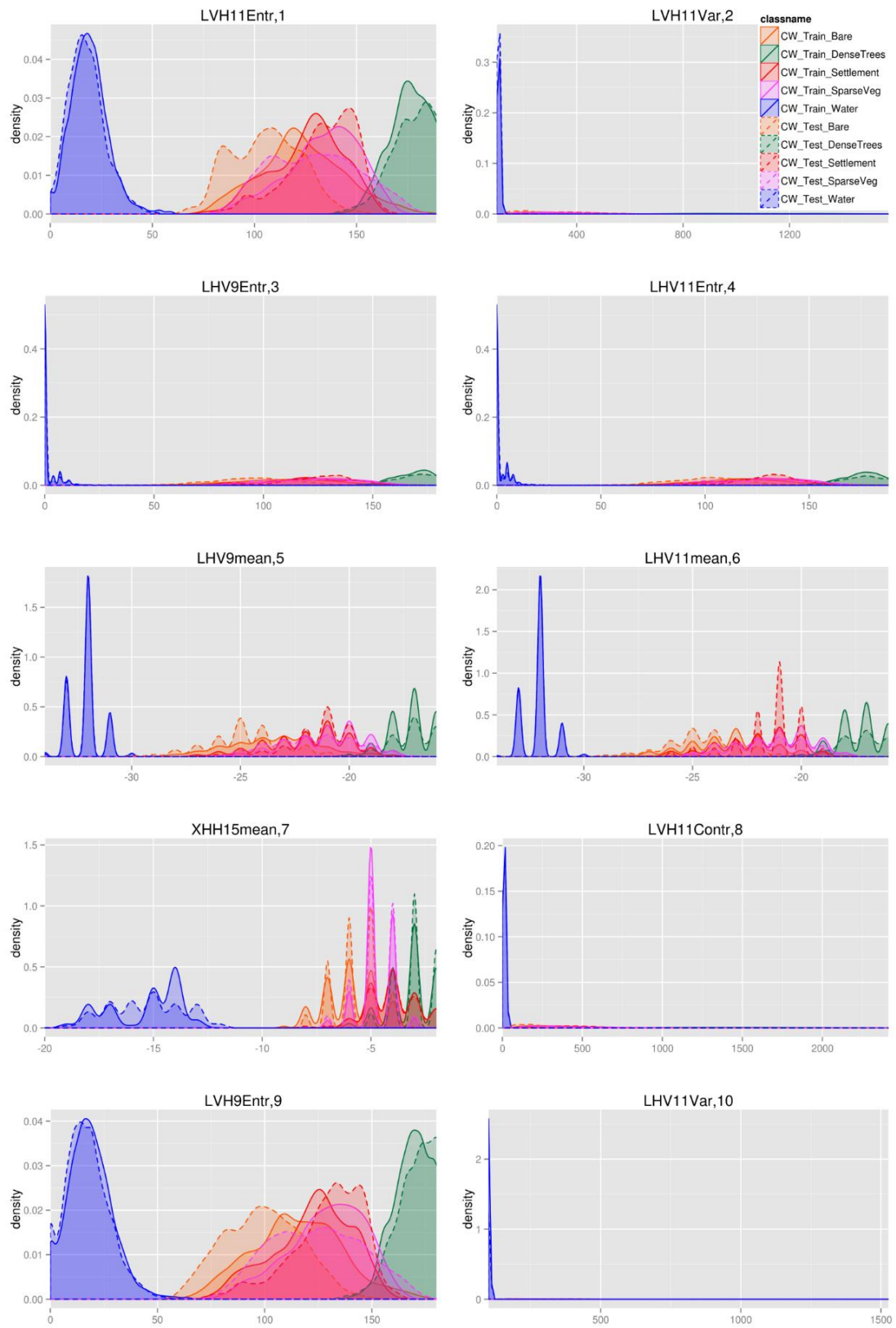


Figure 7.3 Distribution of the top 10 texture measures for the Cameroon wet season training vs test samples. Order based on MDG variable importance of a random forest model of the texture measures.



### Cameroon wet season training vs test samples: elevation data

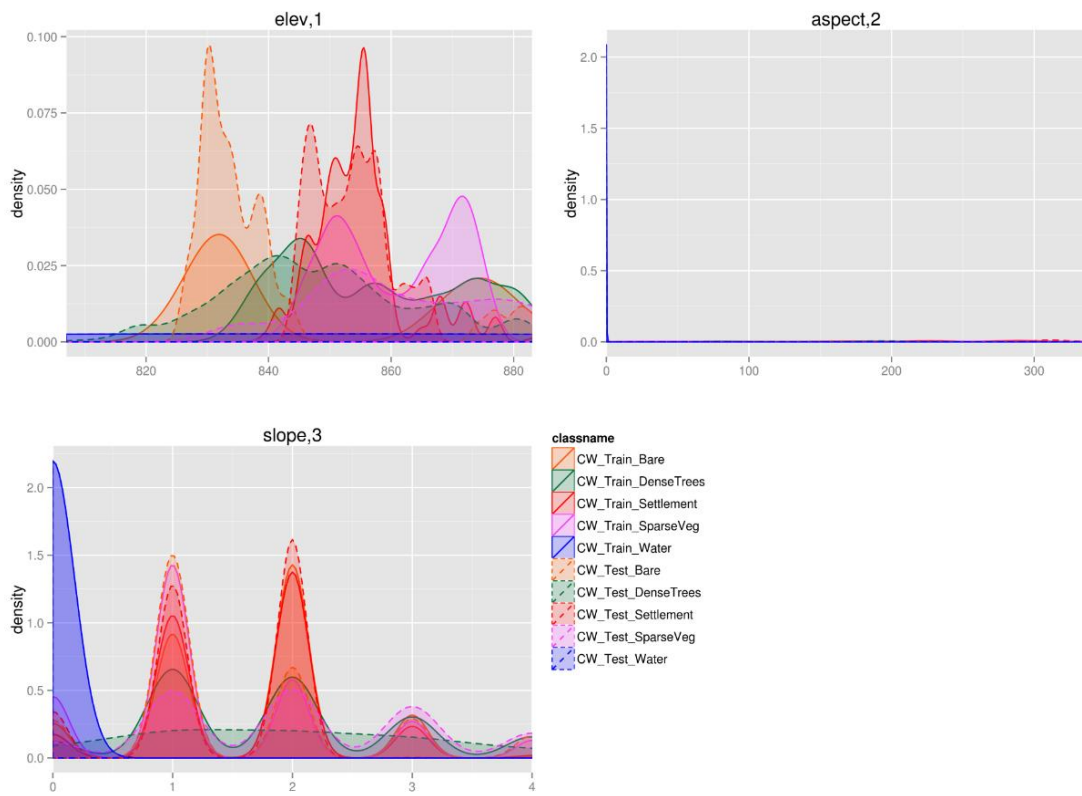


Figure 7.4 Distribution of elevation, slope and aspect data for the Cameroon wet season training vs test samples. Order based on MDG variable importance of a random forest model of the elevation variables.

### 7.1.2 Tanzania dry season training vs test samples

Figures 7.5, 7.6, 7.7 and 7.8 show the comparison of the statistical distributions of the land cover classes: *agriculture*, *bare*, *dense trees*, *sparse vegetation* and *settlement* for the Tanzania dry season training vs test samples, for the backscatter coefficients, interchannel ratios, texture measures and elevation data respectively. The training and test data are relatively similar for all the backscatter coefficient channels and calculated information layers, as expected. Similar to the forested ecoregion graphs, the comparison between the Tanzania dry training vs test samples were just to make sure that the land cover classes are similar between the training and test samples. There is, however, a much larger overlap for all the classes for the semi-arid ecoregion sites, compared to the forested ecoregion sites. This explains why the classification accuracies for the semi-arid sites were much lower than the forested sites in Chapter Five and Chapter Six.

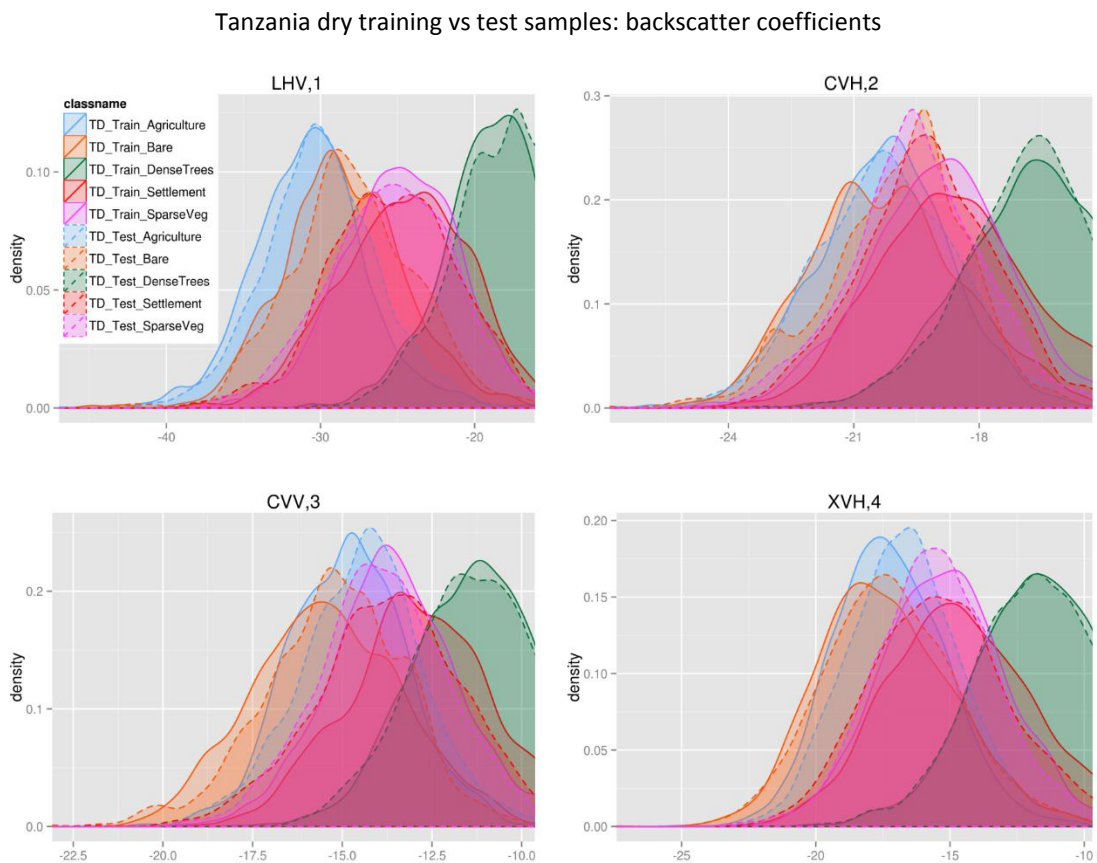


Figure 7.5 Distribution of backscatter coefficients for the Tanzania dry training vs test samples. Order based on MDG variable importance of a random forest model of the backscatter coefficients.

### Tanzania dry training vs test samples: interchannel ratios

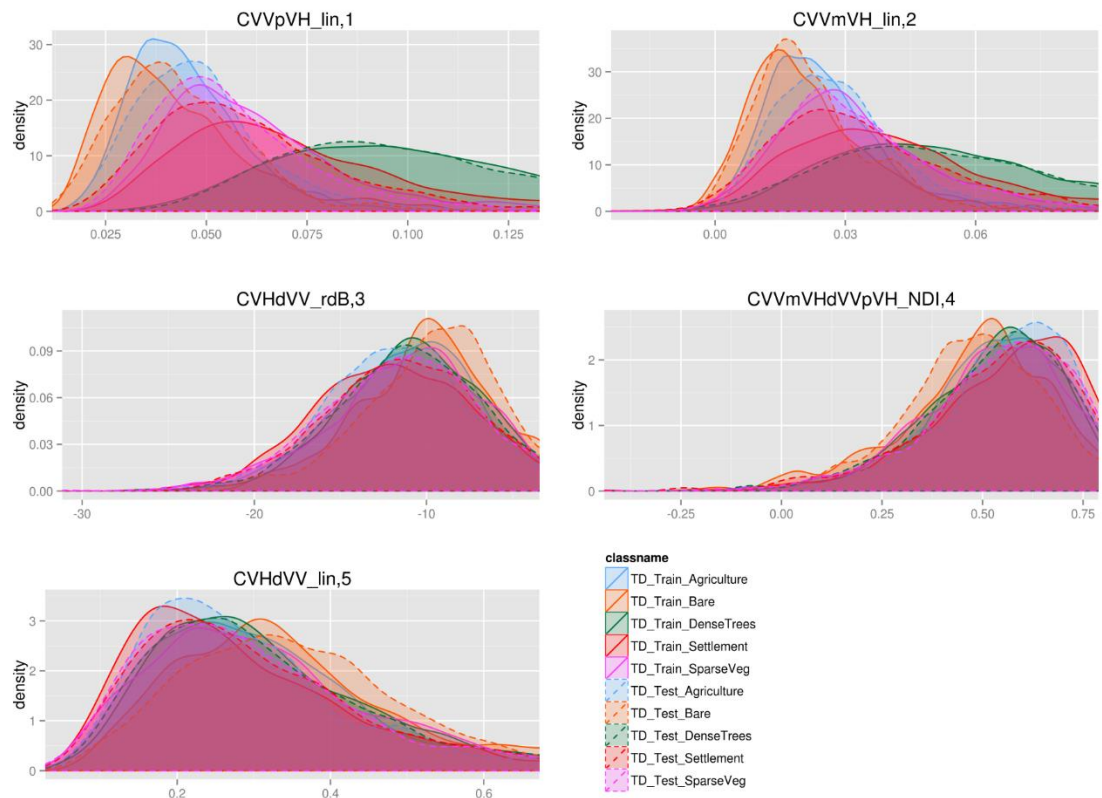


Figure 7.6 Distribution of the interchannel ratios for the Tanzania dry training test samples. Order based on MDG variable importance of a random forest model of the interchannel ratios.

### Tanzania dry training vs test samples: texture measures

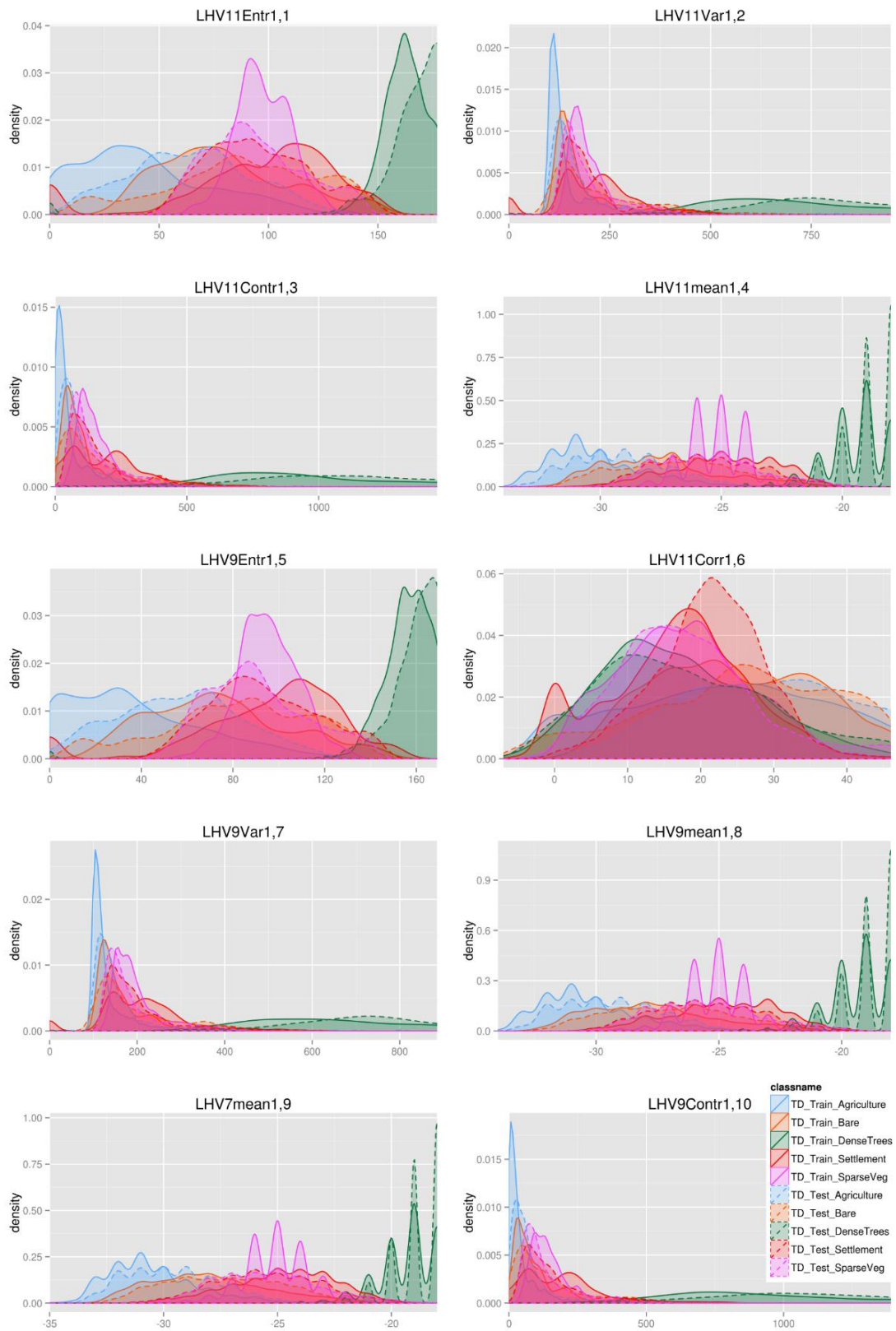


Figure 7.7 Distribution of the top 10 texture measures for the Tanzania dry training vs test samples. Order based on MDG variable importance of a random forest model of the texture measures.

### Tanzania dry training vs test samples: elevation data

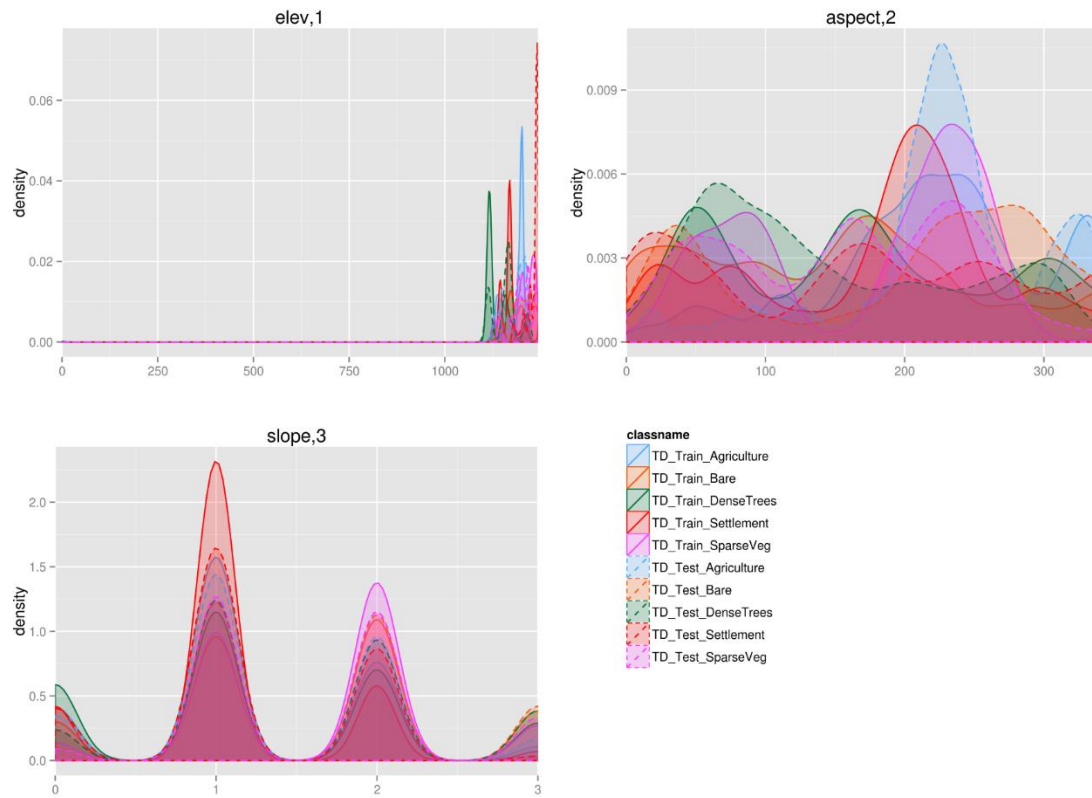


Figure 7.8 Distribution of the elevation, slope and aspect for the Tanzania dry training vs test samples. Order based on MDG variable importance of a random forest model of the elevation data.

## 7.2 Specific methodology for SAR signature extraction

The Kullback-Leibler divergence (KLD) (Kullback and Leibler, 1951; Kullback, 1987), also referred to as Relative Entropy (RE), is the relative comparison between two probability distributions  $p$  and  $q$  on a variable  $X$ . The discrete version of the KLD between  $p$  and  $q$  is given by Phillips et al. (2004):

$$KLD(p \parallel q) = \sum_{x \in X} p(x) \ln \frac{p(x)}{q(x)} \quad \text{Equation 7-1}$$

The KLD between the different distributions are minimized in the Maximum Entropy model (MaxEnt) (Phillips and Dudik, 2008; Elith et al., 2011). In this study, the KLD will be used to assess the comparison for the forested sites: between the Cameroon wet season training and test samples, the Cameroon wet vs dry season samples and then the Cameroon wet vs DRC wet season samples. Similarly, a comparison was undertaken for the semi-arid sites: between the Tanzania dry season training and test samples, the Tanzania dry vs wet season samples and between the Tanzania and Chad dry season samples. A list of the land cover sample data distributions evaluated is given in Table 7.2. The scenarios for which the KLD measures are calculated for the different land cover classes are also outlined in Table 7.2.

Table 7.2 List of land cover sample data distribution comparisons, both visual and by the KLD, for the forested and semi-arid sites.

Scenario (x)	Comparison	Visual comparison: forested site (Section 7.3.1.x)	Visual comparison: semi-arid site (Section 7.3.2.x)	KLD: forested site (Section 7.3.3.x)	KLD: semi-arid site (Section 7.3.4.x)
(1)	Training vs test	Cameroon wet season	Tanzania dry season	Cameroon wet season	Tanzania dry season
	Backscatter coefficients	✓	✓	✓	✓
	Interchannel ratios	✓	✓	✗	✗
	Texture measures	✓ (top 10)	✓ (top 10)	✓ (top 8)	✓ (top 8)
	Elevation data	✓	✓	✗	✗
(2)	Wet vs dry	Cameroon wet vs dry season	Tanzania dry vs wet season	Cameroon wet vs dry season	Tanzania dry vs wet season
	Backscatter coefficients	✓	✓	✓	✓
	Interchannel ratios	✓	✓	✗	✗
	Texture measures	✓ (top 10)	✓ (top 10)	✓ (top 8)	✓ (top 8)
	Elevation data	✓	✓	✗	✗
(3)	Site1 vs Site2	Cameroon wet vs DRC wet season	Tanzania dry vs Chad dry season	Cameroon wet vs DRC wet season	Tanzania dry vs Chad dry season
	Backscatter coefficients	✓	✓	✓	✓
	Interchannel ratios	✓	✓	✗	✗
	Texture measures	✓ (top 10)	✓ (top 10)	✓ (top 8)	✓ (top 8)
	Elevation data	✓	✓	✗	✗

## 7.3 Results

The statistical distributions of the land cover samples of the SAR backscatter coefficients, interchannel ratios, texture measures and elevation data are presented in Section 7.3.1 for the forested sites and for the scenarios shown in Table 7.2. Similarly, the visual comparison of the SAR signatures of the semi-arid sites are presented in Section 7.3.2. The KLD for the land cover classes of the forested sites are shown in Section 7.3.3, with the KLD for the land cover classes of the semi-arid sites presented in Section 7.3.4.

### 7.3.1 Visual comparison of SAR signatures for the forested sites (Cameroon and DRC)

In this section, the statistical distributions, of the backscatter coefficients, interchannel ratios, texture measures and elevation data for the forested sites, are compared for the site combinations as shown in Table 7.3.

*Table 7.3 The site comparisons for the forested sites with corresponding figure numbers.*

<b>Site1 samples</b>	<b>Site2 samples</b>	<b>Backscatter coefficients figure nr</b>	<b>Interchannel ratios figure nr</b>	<b>Texture measures figure nr</b>	<b>Elevation data figure nr</b>
CW-training	CD-training	7.9	7.10	7.11	7.12
CW-training	DW-training	7.13	7.14	7.15	7.16

#### 7.3.1.1 Cameroon wet vs dry season samples

Figures 7.9, 7.10, 7.11 and 7.12 show the same four graphs of the CW-training samples compared to the CD-training samples. This comparison was undertaken to assess whether a model developed on the wet season data can be applied directly to the dry season data. The classes between the wet and dry seasons look quite similar, although a definite distribution shift is noticed for all the land cover classes. There is a noticeable difference between the CW *water* and CD *water* classes for backscatter coefficient layer X-VV, texture measures L-HH mean (9x9, 11x11 window size) and X-VV mean (11x11, 13x13, 15x15 window sizes). The other



classes are relatively similar, with the most noticeable difference being the *bare* class, as shown in the C-VV mean (9x9 window size) and X-HH mean (15x15 window size) texture measures.

Cameroon wet vs dry season samples: backscatter coefficients

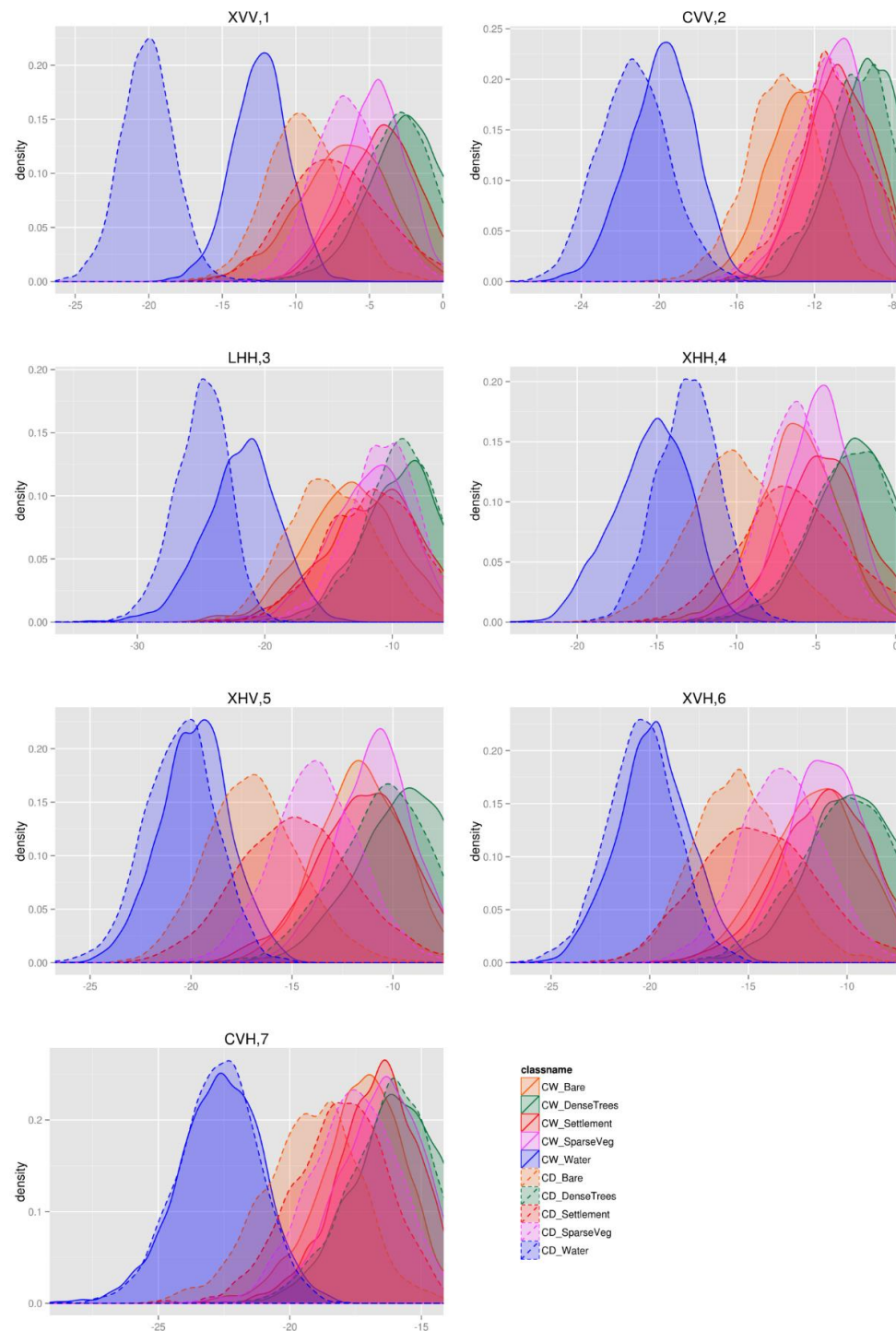


Figure 7.9 Distribution of backscatter coefficients for the Cameroon wet vs dry season samples. Order based on MDG variable importance of a random forest model of the backscatter coefficients.

### Cameroon wet vs dry season samples: interchannel ratios

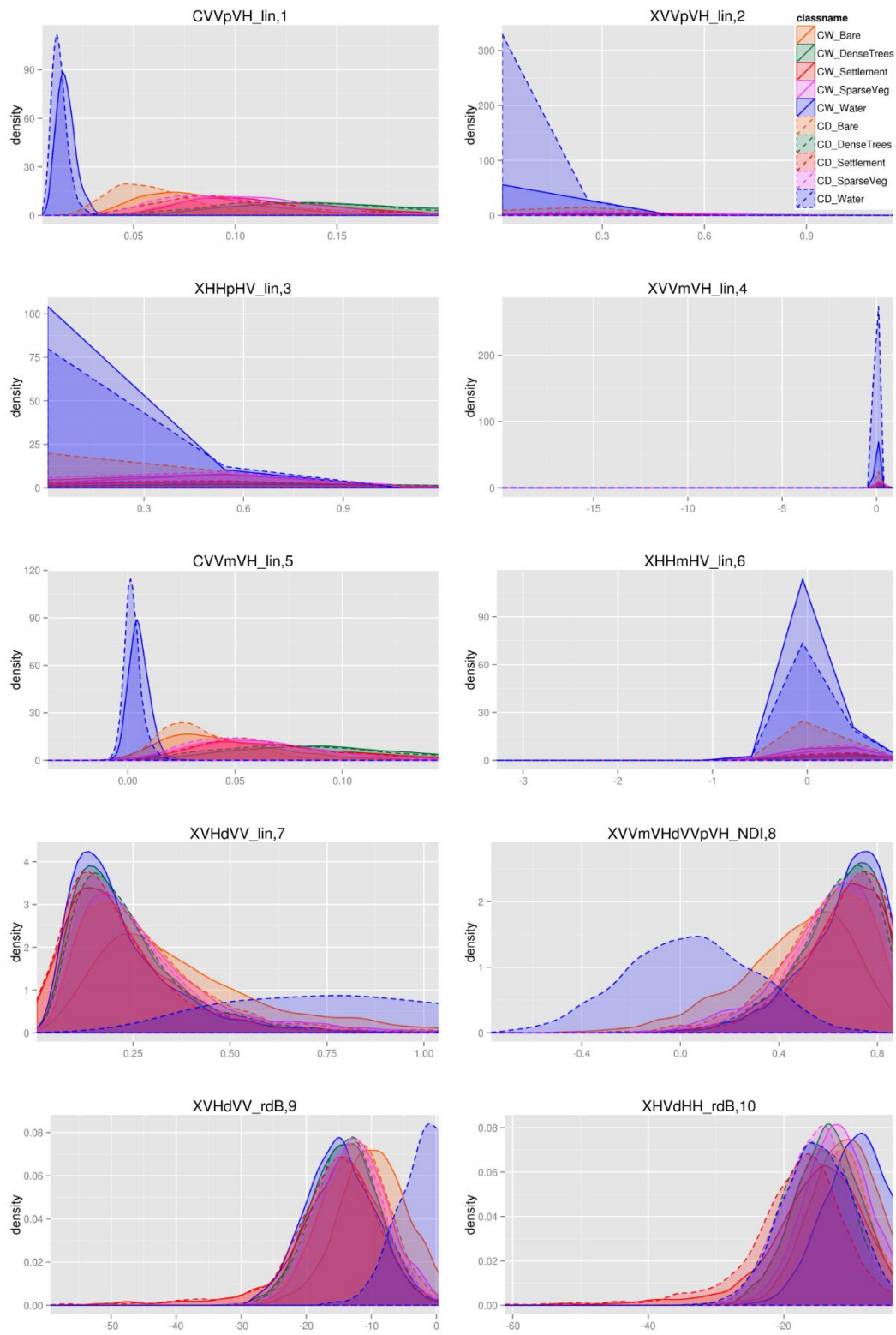


Figure 7.10 Distribution of the top 10 interchannel ratios for the Cameroon wet vs dry season samples. Order based on MDG variable importance of a random forest model of the interchannel ratios.

### Cameroon wet vs dry season samples: texture measures

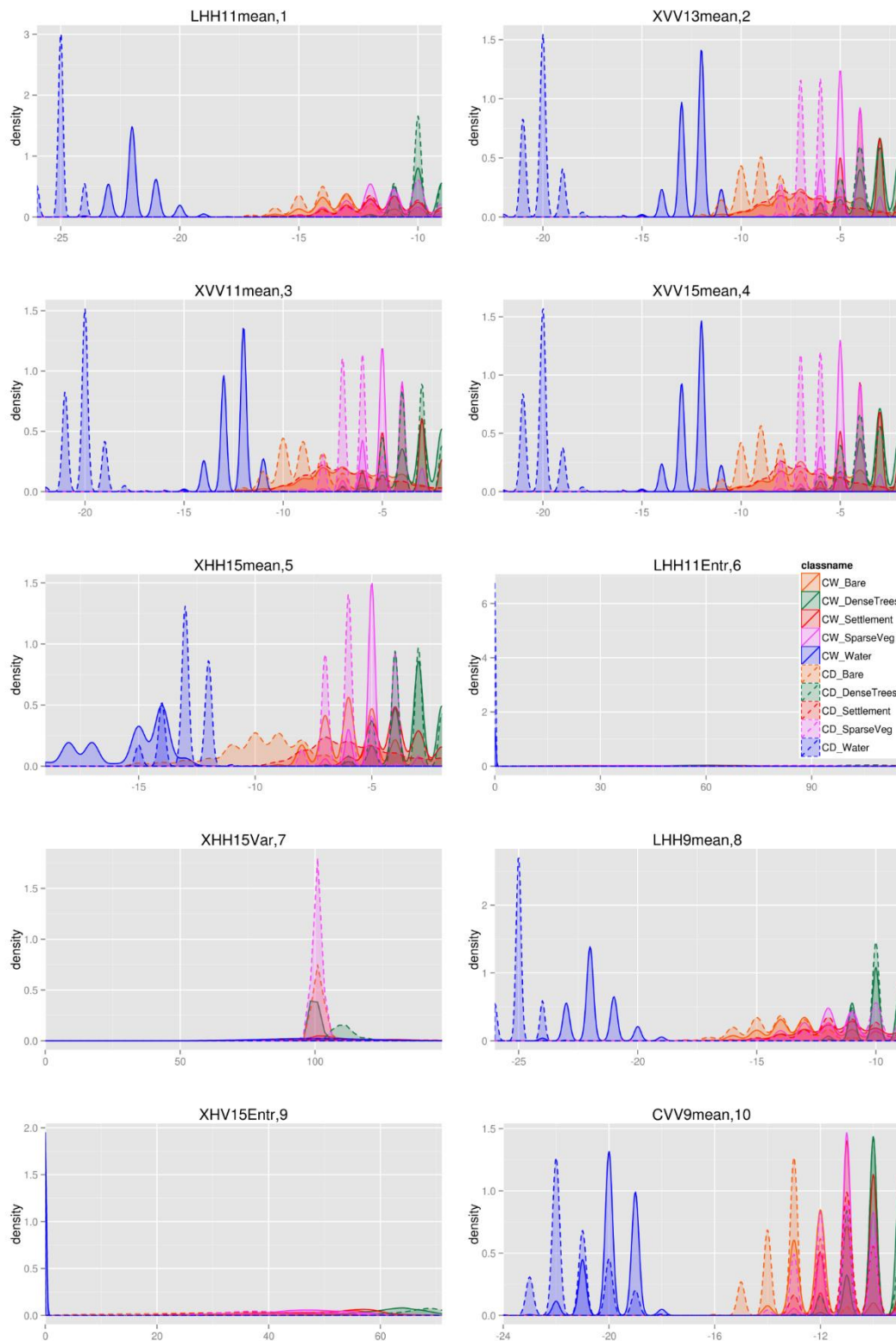


Figure 7.11 Distribution of the top 10 texture measures for the Cameroon wet vs dry season samples. Order based on MDG variable importance of a random forest model of the texture measures.

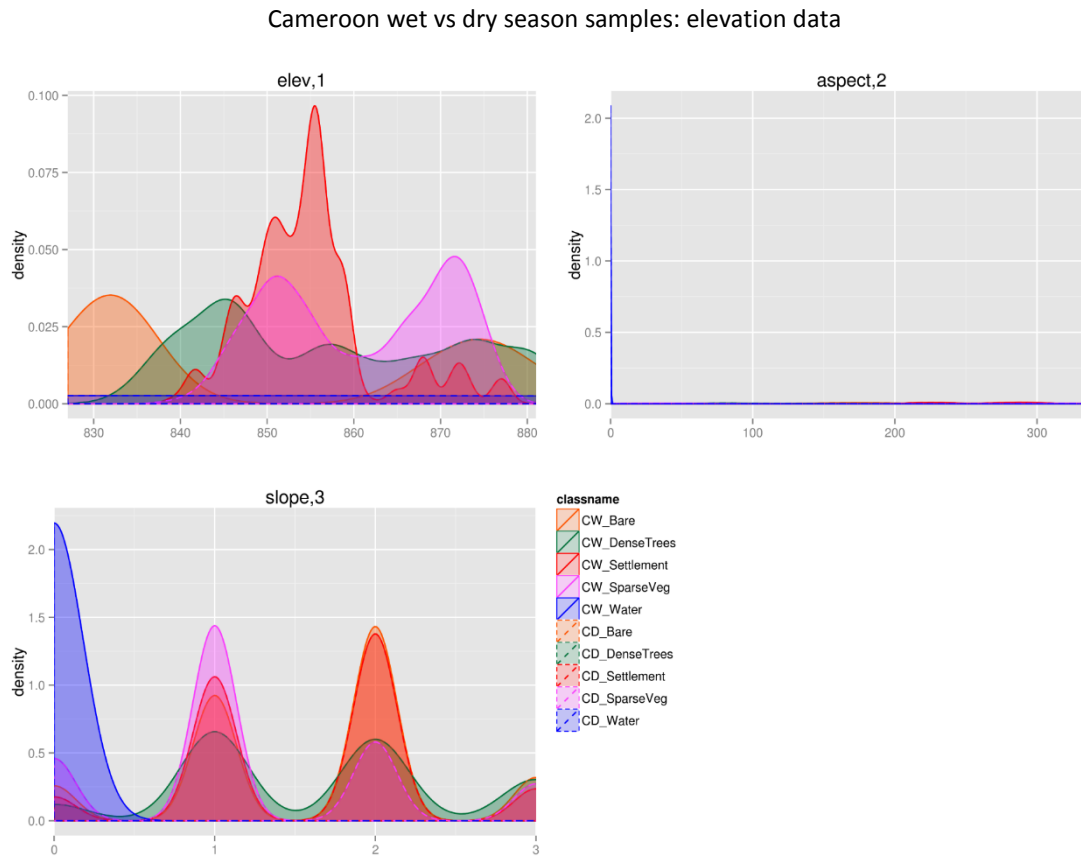


Figure 7.12 Distribution of elevation, slope and aspect data for the Cameroon wet vs dry season samples. Order based on MDG variable importance of a random forest model of the elevation variables.

### 7.3.1.2 Cameroon vs DRC wet season samples

Figures 7.13, 7.14, 7.15 and 7.16 show statistical distributions of the backscatter coefficients, interchannel ratios, top 10 texture measures and elevation data for the Cameroon vs DRC wet season training samples. Here is a clearer difference between all the classes, though still a large overlap between the Cameroon and DRC wet season classes. The difference in the classes between the two sites is most noticeable in the C-VV backscatter coefficient graph in Figure 7.13, with the *water* classes clearly different and all the DRC wet classes giving higher  $\sigma^0$  values than the Cameroon wet samples.

### Cameroon vs DRC wet season samples: backscatter coefficients

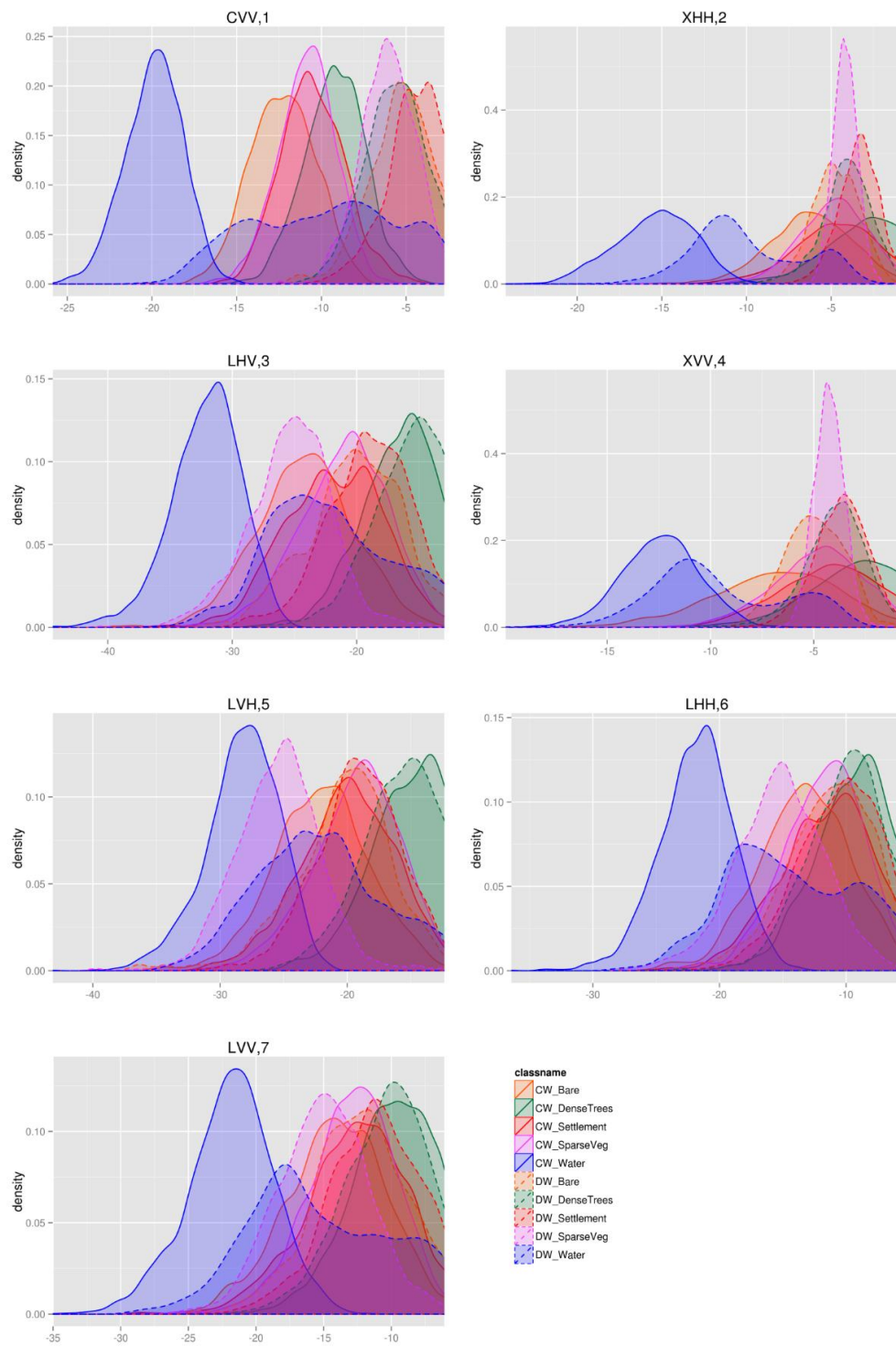


Figure 7.13 Distribution of backscatter coefficients for the Cameroon vs DRC wet season samples. Order based on MDG variable importance of a random forest model of the backscatter coefficients.



# Cameroon vs DRC wet season samples: interchannel ratios

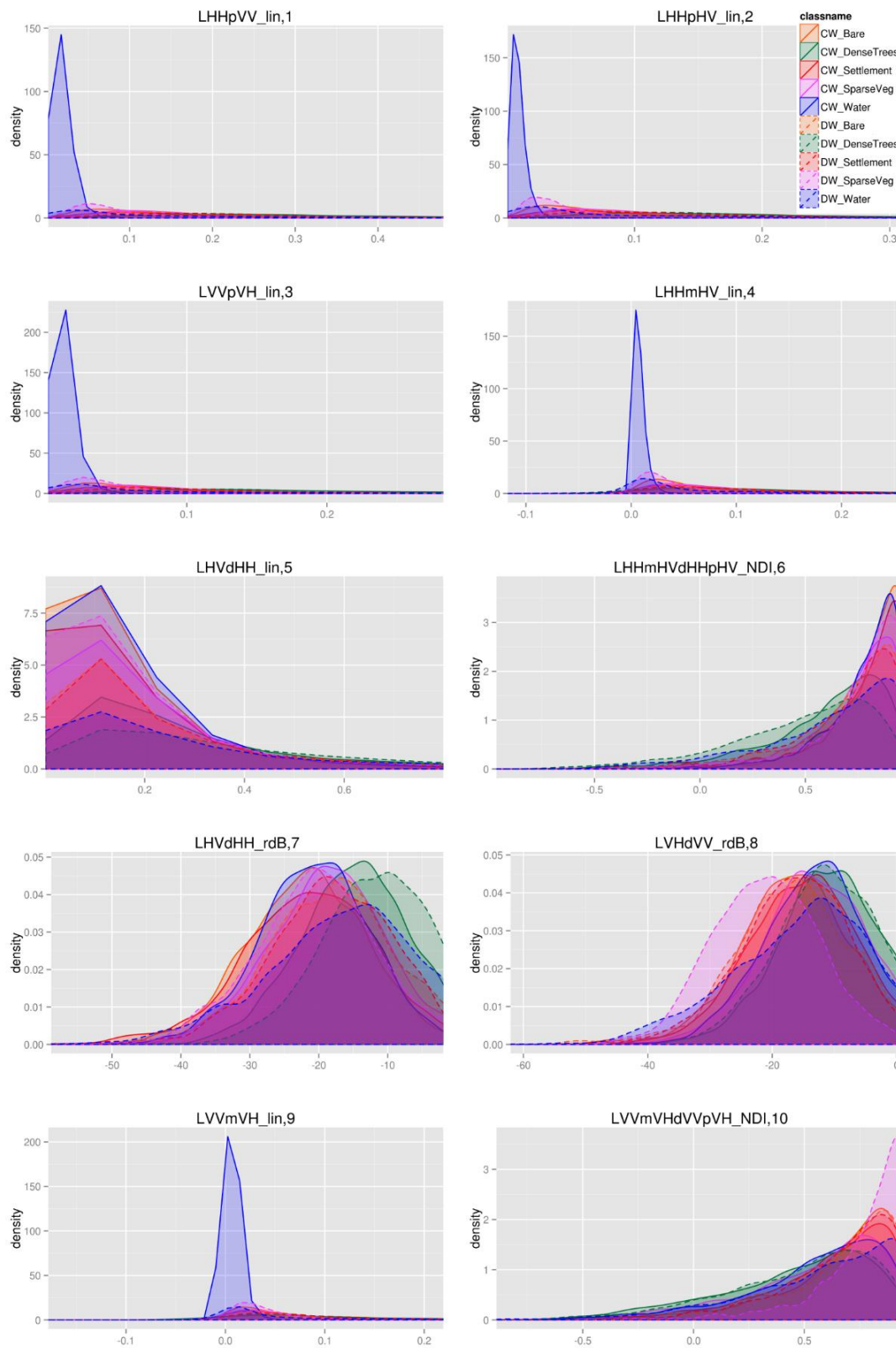


Figure 7.14 Distribution of the top 10 interchannel ratios for the Cameroon vs DRC wet season samples. Order based on MDG variable importance of a random forest model of the interchannel ratios.

### Cameroon vs DRC wet season samples: texture measures

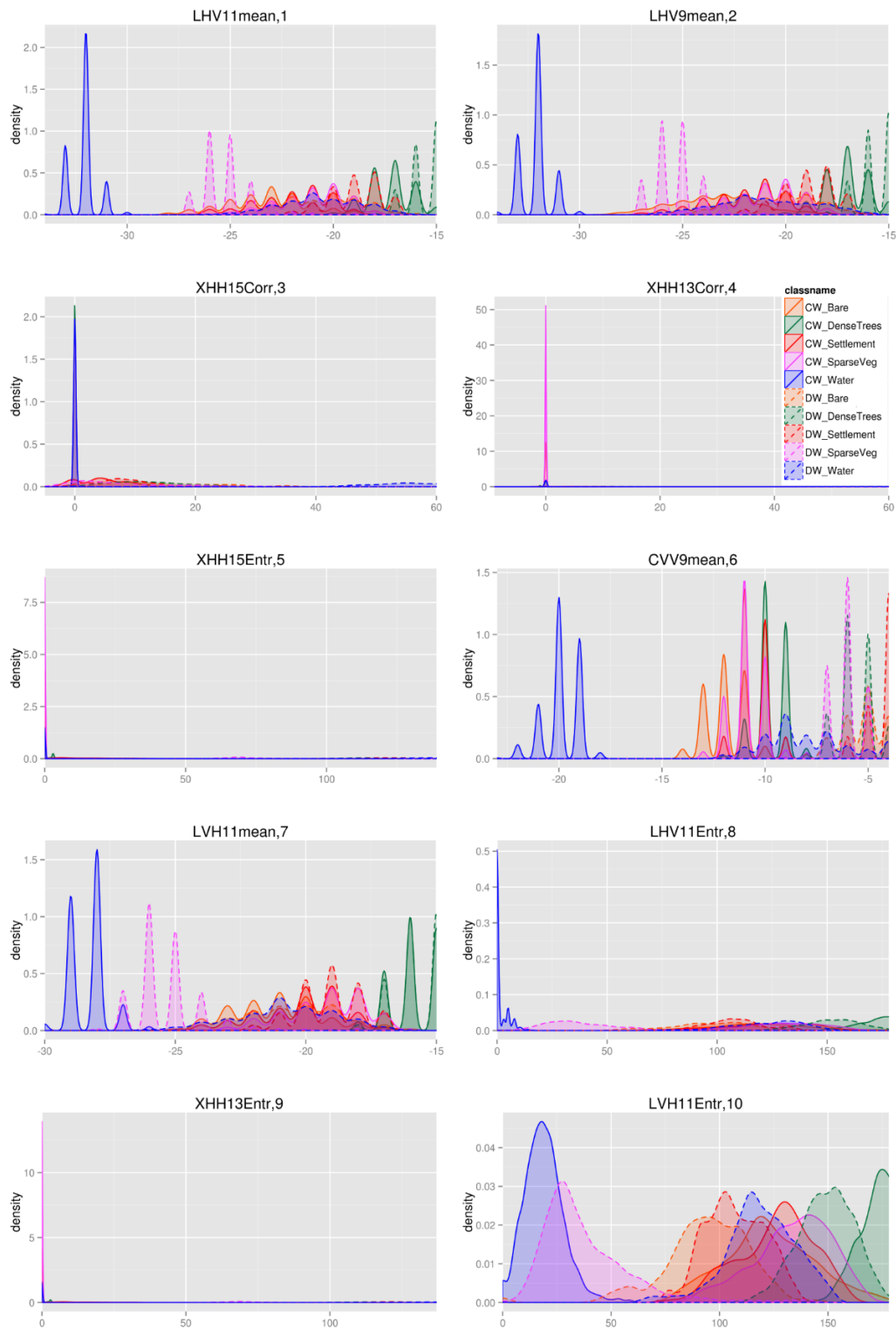


Figure 7.15 Distribution of the top 10 texture measures for the Cameroon vs DRC wet season samples. Order based on MDG variable importance of a random forest model of the texture measures.

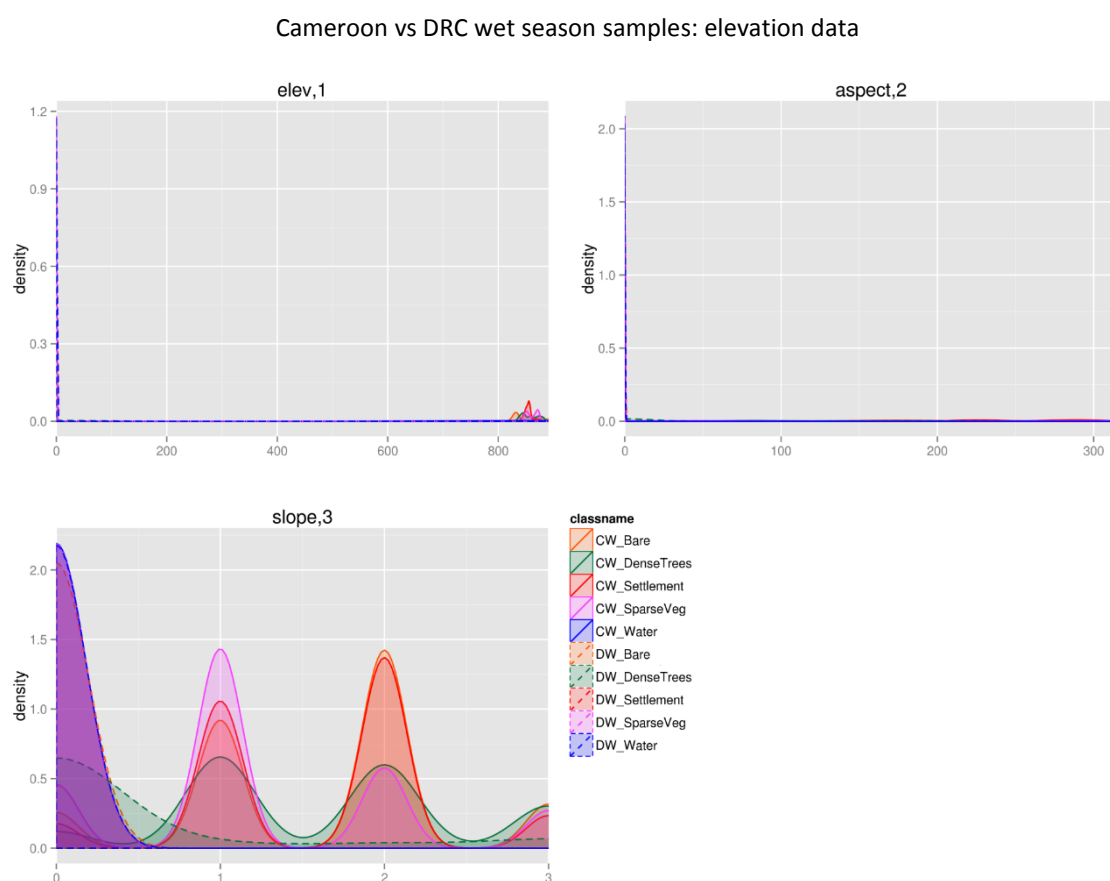


Figure 7.16 Distribution of the elevation, slope and aspect for the Cameroon vs DRC wet season samples. Order based on MDG variable importance of a random forest model of the elevation data.

### 7.3.2 Visual comparison of SAR signatures for the semi-arid sites (Tanzania and Chad)

The visual comparisons of the statistical distributions of the SAR data for the semi-arid sites are given in this section, similar to the forested sites in Section 7.3.1. More specifically, the land cover classes of the Tanzania dry training samples are compared to the Tanzania dry test samples, Tanzania wet training samples and the Chad dry training samples. Table 7.4 shows the figure numbers of the visual comparisons of the statistical distributions of the land cover classes for the Tanzania dry season training vs test samples, Tanzania dry vs wet season samples and Tanzania vs Chad dry season samples. The comparisons of the different land cover classes were undertaken for the backscatter coefficients, interchannel ratios, texture measures and elevations data.



Table 7.4 The site comparisons for the semi-arid sites with corresponding figure numbers.

Site 1 samples	Site 2 samples	Backscatter coefficients figure nr	Interchannel ratios figure nr	Texture measures figure nr	Elevation data figure nr
TD-training	TW-training	7.17	7.18	7.19	7.20
TD-training	ChD-training	7.21	7.22	7.23	7.24

### 7.3.2.1 Tanzania dry vs wet season samples

Figures 7.17, 7.18, 7.19 and 7.20 show the comparison of the statistical distributions of the land cover classes: *agriculture*, *bare*, *dense trees*, *sparse vegetation* and *settlement* for the Tanzania dry season vs wet season samples, for the backscatter coefficients, interchannel ratios, texture measures and elevation data respectively. In Figure 7.17 it is noted that there was a substantial difference between the dry and wet season especially for the *bare* land cover class, for the L-HV, C-VV and C-VH backscatter coefficient channels, and the *agriculture* class, in the L-HV and C-VV backscatter coefficient channels. Figure 7.18 shows that the interchannel ratios which has a large overlap for all the land cover classes. Figure 7.19 shows the texture measures have a significant difference between the wet and dry season land cover classes. The elevation data in Figure 7.20 shows that the distributions between the wet and dry seasons are exactly the same, since the elevation, slope and aspect remain the same between the wet and dry seasons.

### Tanzania dry vs wet season samples: backscatter coefficients

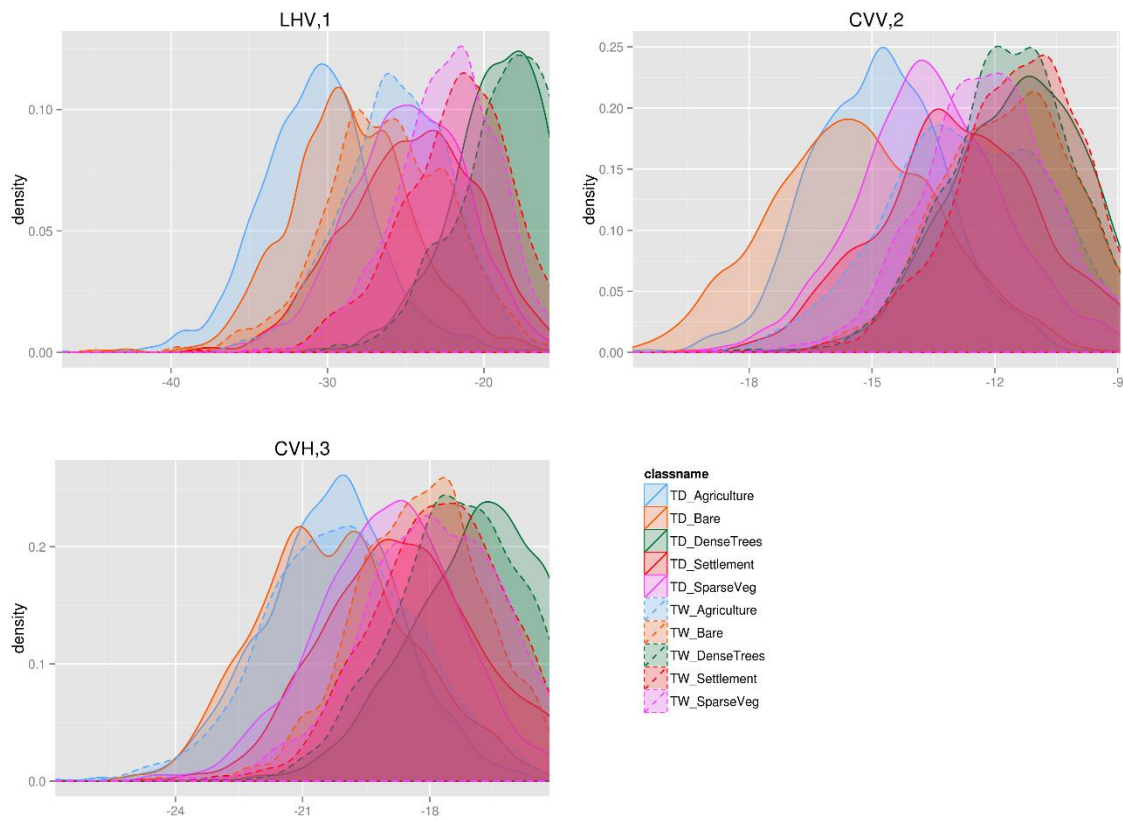


Figure 7.17 Distribution of backscatter coefficients for the Tanzania dry vs wet season samples. Order based on MDG variable importance of a random forest model of the backscatter coefficients.

### Tanzania dry vs wet season samples: interchannel ratios

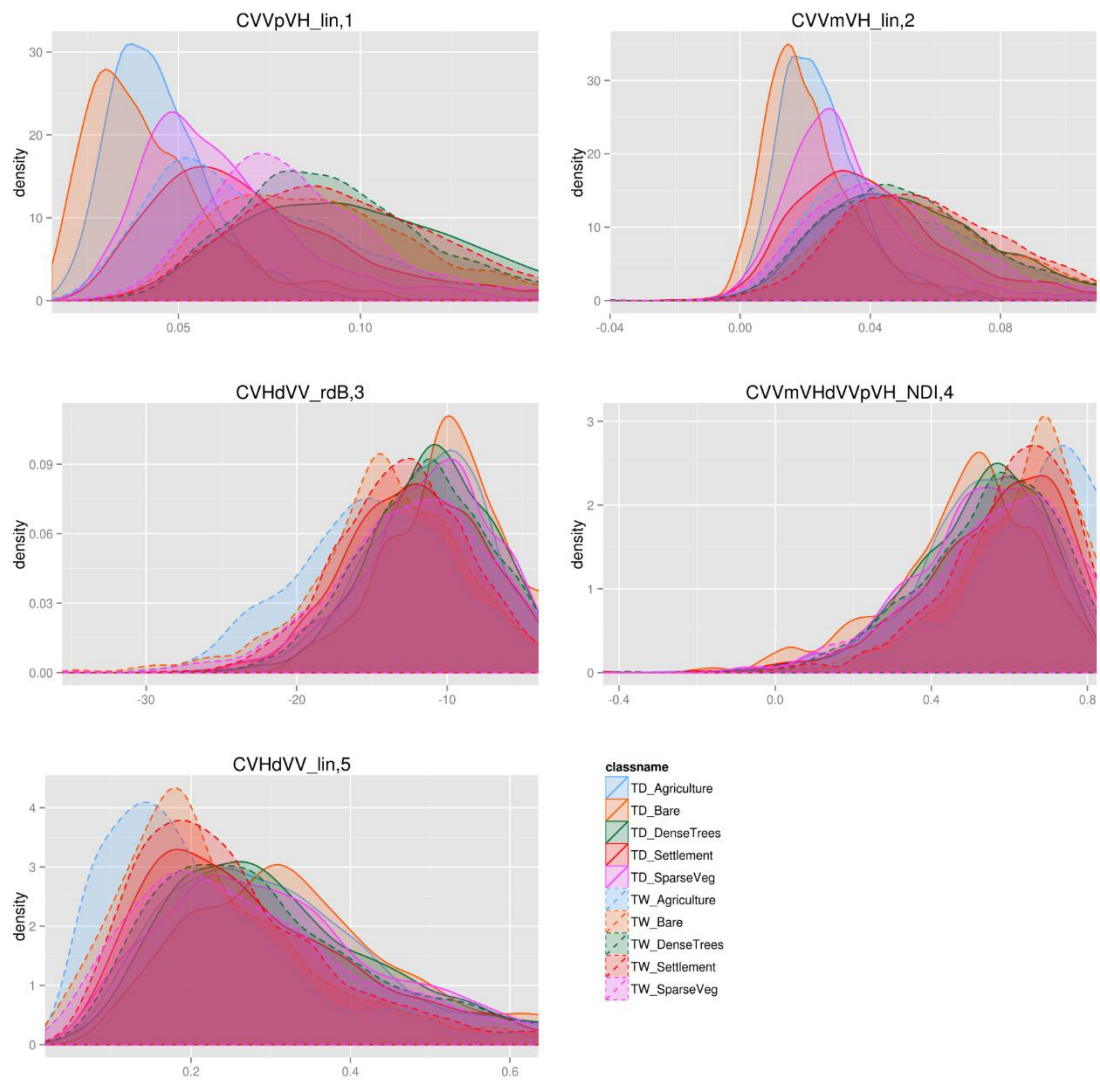


Figure 7.18 Distribution of the interchannel ratios for the Tanzania dry vs wet season samples. Order based on MDG variable importance of a random forest model of the interchannel ratios.

### Tanzania dry vs wet season samples: texture measures

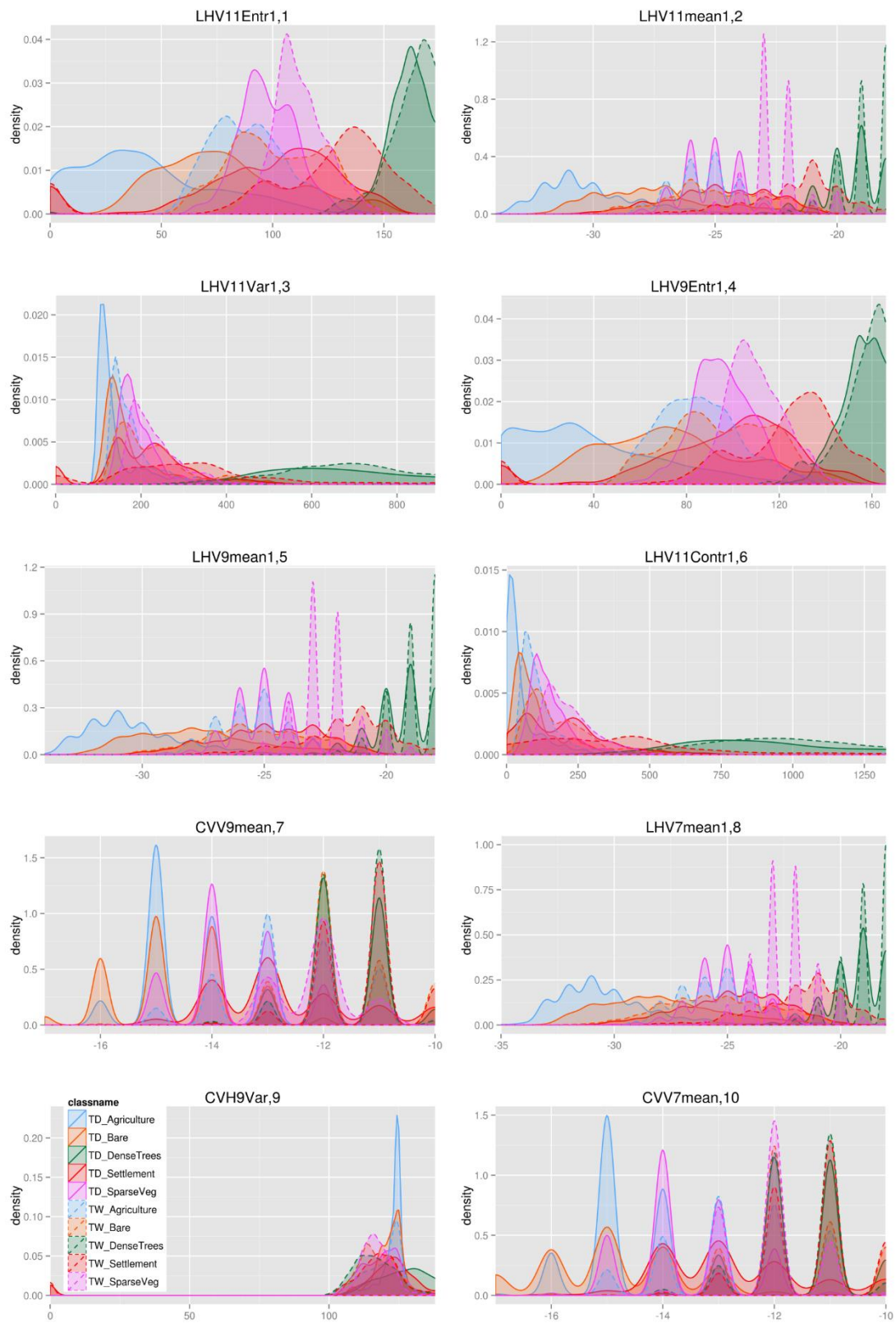


Figure 7.19 Distribution of the top 10 texture measures for the Tanzania dry vs wet season samples. Order based on MDG variable importance of a random forest model of the texture measures.

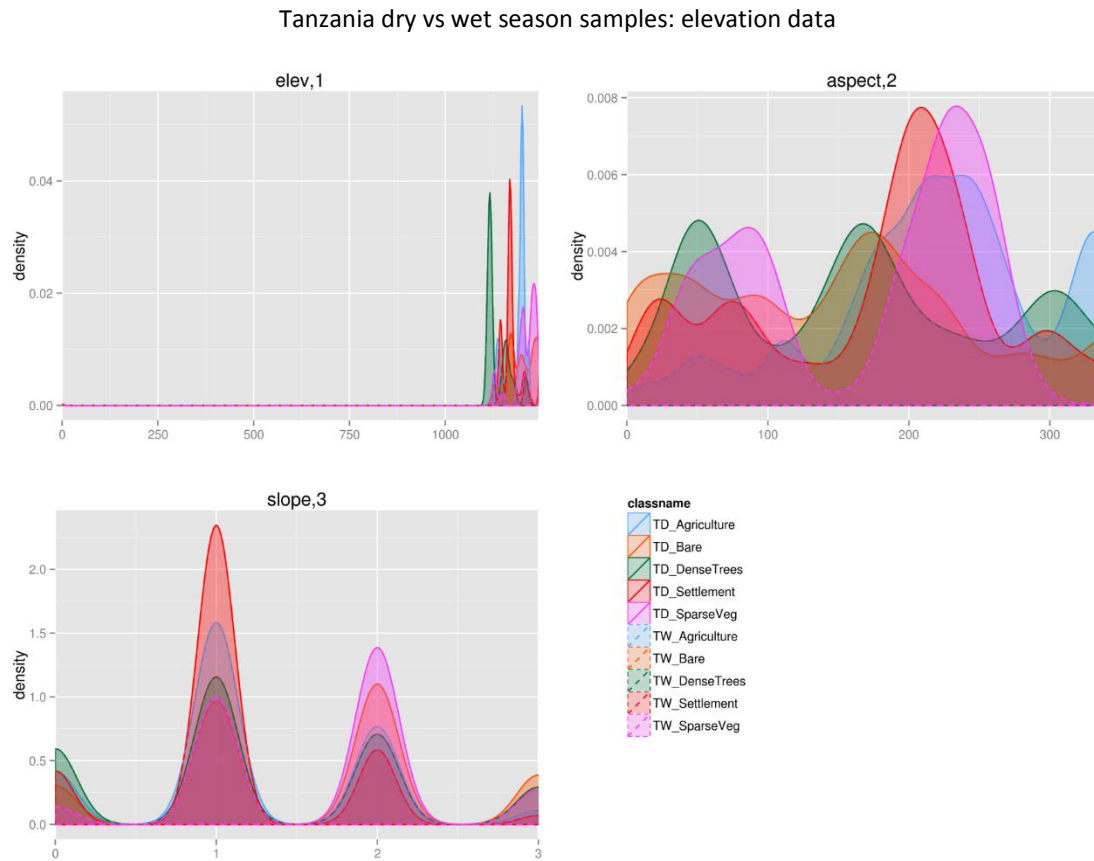


Figure 7.20 Distribution of the elevation, slope and aspect for the Tanzania dry vs wet season samples. Order based on MDG variable importance of a random forest model of the elevation data.

### 7.3.2.2 Tanzania vs Chad dry season samples

Figures 7.21, 7.22, 7.23 and 7.24 show the comparison of the statistical distributions of the land cover classes: *agriculture*, *bare*, *dense trees*, *sparse vegetation* and *settlement* for the Tanzania vs Chad dry season samples, for the backscatter coefficients, interchannel ratios, texture measures and elevation data respectively. There was a substantial difference for all the land cover classes for the backscatter coefficient channels in Figure 7.21 and the additionally calculated information layers in Figures 7.22 and 7.23. There was also a substantial difference in the elevation, slope and aspect data between the Tanzania and Chad sites as shown in Figure 7.24.

### Tanzania vs Chad dry season samples: backscatter coefficients

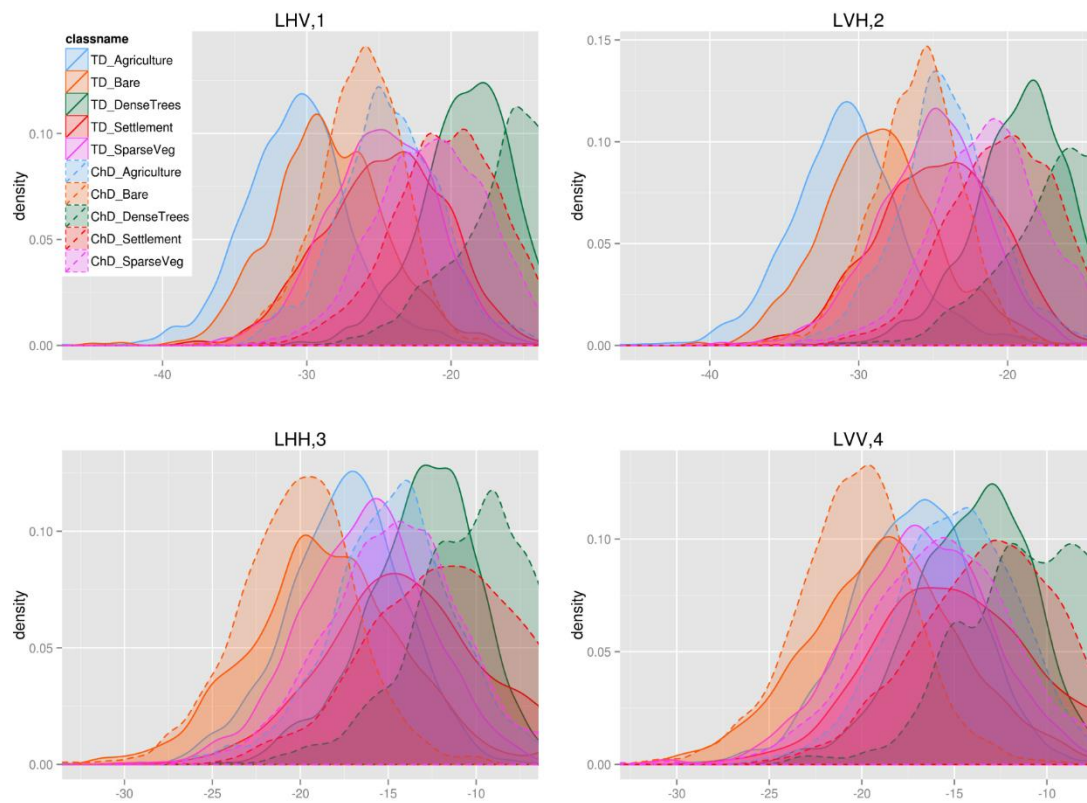


Figure 7.21 Distribution of backscatter coefficients for the Tanzania vs Chad dry season samples. Order based on MDG variable importance of a random forest model of the backscatter coefficients.

# Tanzania vs Chad dry season samples: interchannel ratios

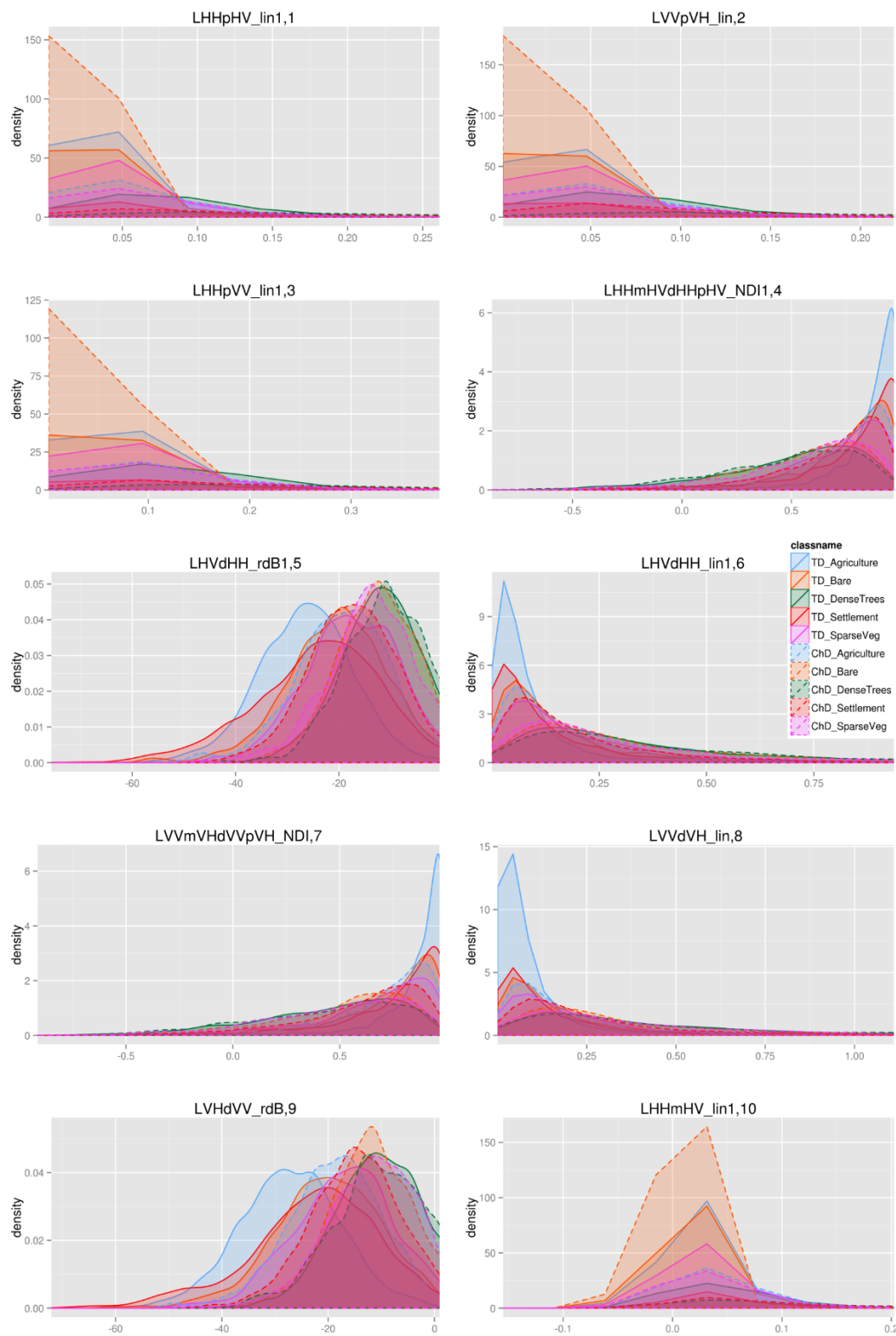


Figure 7.22 Distribution of the top 10 interchannel ratios for the Tanzania vs Chad dry season samples. Order based on MDG variable importance of a random forest model of the interchannel ratios.



## Tanzania vs Chad dry season samples: texture measures

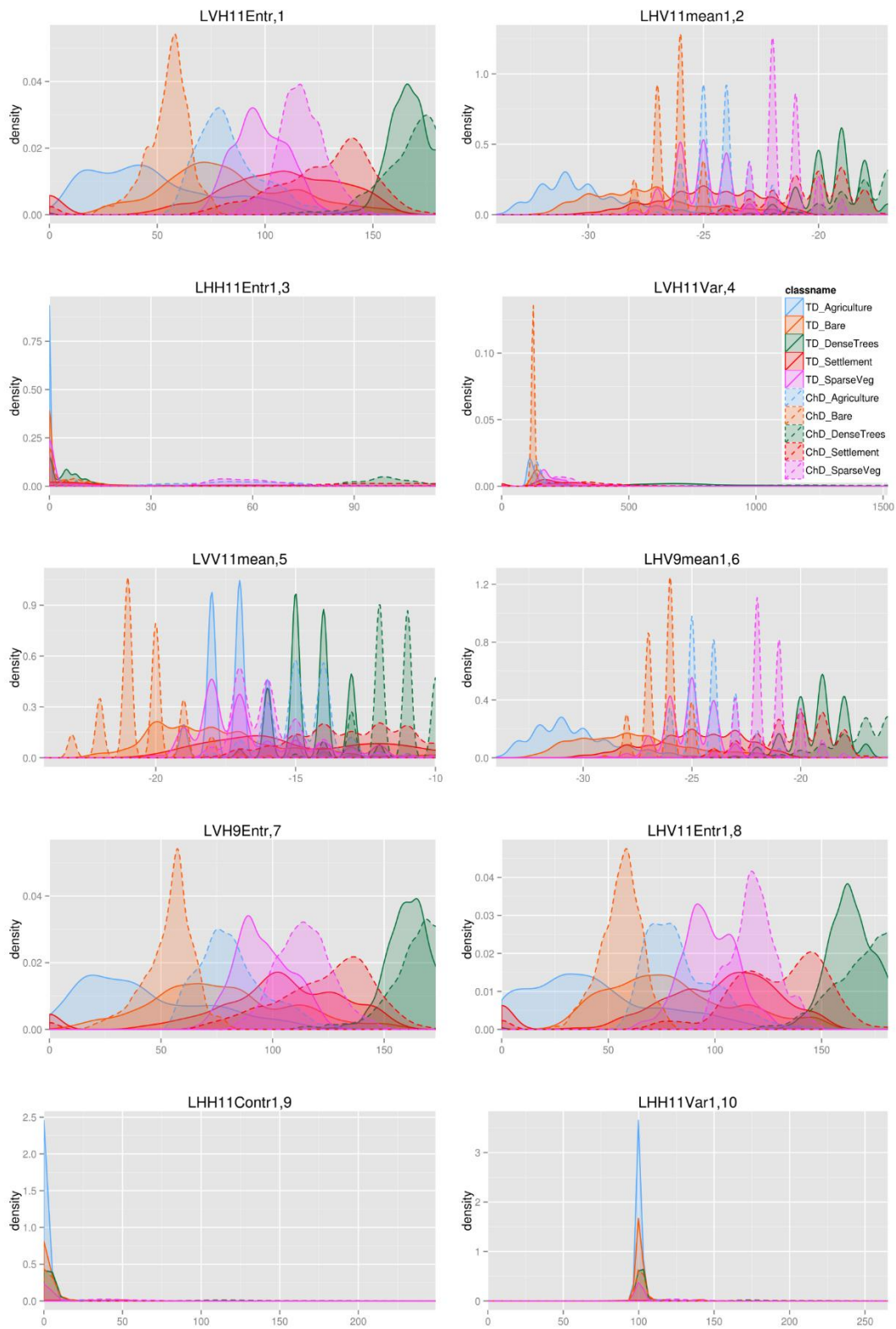


Figure 7.23 Distribution of the top 10 texture measures for the Tanzania vs Chad dry season samples. Order based on MDG variable importance of a random forest model of the texture measures.



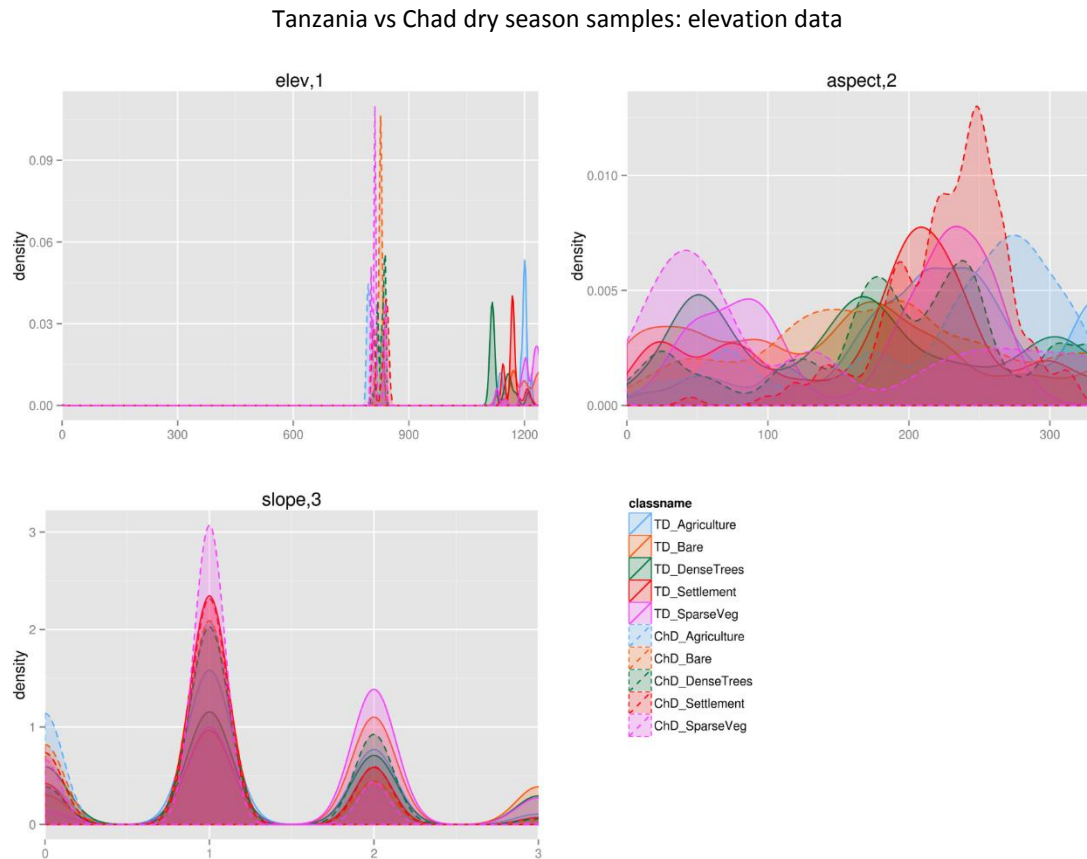


Figure 7.24 Distribution of the elevation, slope and aspect for the Tanzania vs Chad dry season samples. Order based on MDG variable importance of a random forest model of the elevation data.

### 7.3.3 KLD for land cover classes of the forested ecoregion (Cameroon and DRC)

The KLD measures were compared for the land cover classes of the forested sites. The comparison was performed between the Cameroon wet season training vs test samples, the Cameroon wet vs dry season samples and between the Cameroon vs DRC wet season samples. The types of variables considered are the backscatter coefficients and the top 8 texture measures with corresponding figure numbers given in Table 7.5. The order of the texture measures is given by the MDG variable importance measures during random forest model development in Chapter 6.

Table 7.5 The site comparison using KLD of the forested sites with corresponding figure numbers.

Site comparison using KLD	Type of variables	Figure number
Cameroon wet season training vs test	Backscatter coefficients	7.26
	Texture measures (top 8)	7.27
Cameroon wet vs dry season	Backscatter coefficients	7.28
	Texture measures (top 8)	7.29
Cameroon wet season vs DRC wet season	Backscatter coefficients	7.30
	Texture measures (top 8)	7.31

In the KLD plots that follow, the ideal comparison would require similar land cover classes between the two sites in question (e.g. the statistical distribution of the *dense trees* of the Cameroon wet season samples would be similar to the statistical distribution of the *dense trees* class of the DRC wet season samples, implying a small KLD). At the same time, to be able to distinguish the classes successfully, the statistical distribution of the *dense trees* class would be different to the other classes, resulting in a larger KLD. This ideal scenario is illustrated in Figure 7.25. A KLD of more than 1.0 would mean that the statistical distributions are completely separate. To show that two distributions are similar, to exactly the same, a value of less than 0.2 was chosen by observing the KLD of the Cameroon wet training vs test site backscatter coefficient comparison in Figure 7.26. However, this ideal scenario was not found within the data used in this study, and if it was, advanced machine learning algorithms such as random forests and SVMs would not be required to achieve accurate land cover classifications.

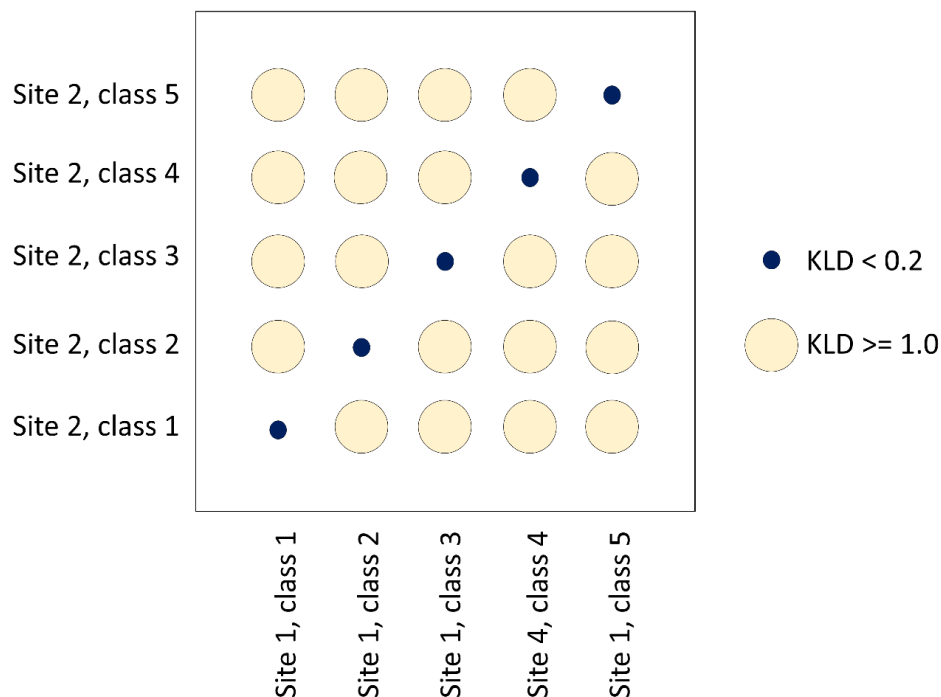


Figure 7.25 The ideal scenario of KLD between two sites that would allow models to be completely transferable between site 1 and site 2. The classes are similar between the two sites, whereas the differences between classes are completely separable.

### 7.3.3.1 KLD for Cameroon wet training vs test samples

Figure 7.26 shows then KLD measures of the backscatter coefficients for the five land cover classes (*bare, dense trees, settlement, sparse vegetation* and *water*) for the Cameroon wet training vs test samples. Figure 7.27 shows the KLD measures of the textures measures, for the Cameroon wet training vs test samples. The KLD measures of the training vs test samples are done as a baseline measurement. First the backscatter coefficients, and then the top 8 texture measures, which were the variables that were highlighted as being the most predictive in the random forest model. Both the backscatter coefficients (Figure 7.26) and the texture measures (Figure 7.27) of the Cameroon wet training vs test samples were relatively close to the ideal scenario of Figure 7.25. The diagonal elements are small values in most cases, apart from densetrees class for the Cameroon training vs test sample texture measures in (Figure 7.27). Some of the off-diagonal elements, which need to be larger values for a good class separation, were small, which indicates therefore that the classes are not

completely separable, e.g. settlements vs sparse vegetation for both the backscatter coefficient (Figure 7.26) and texture measure scenarios (Figure 7.27).

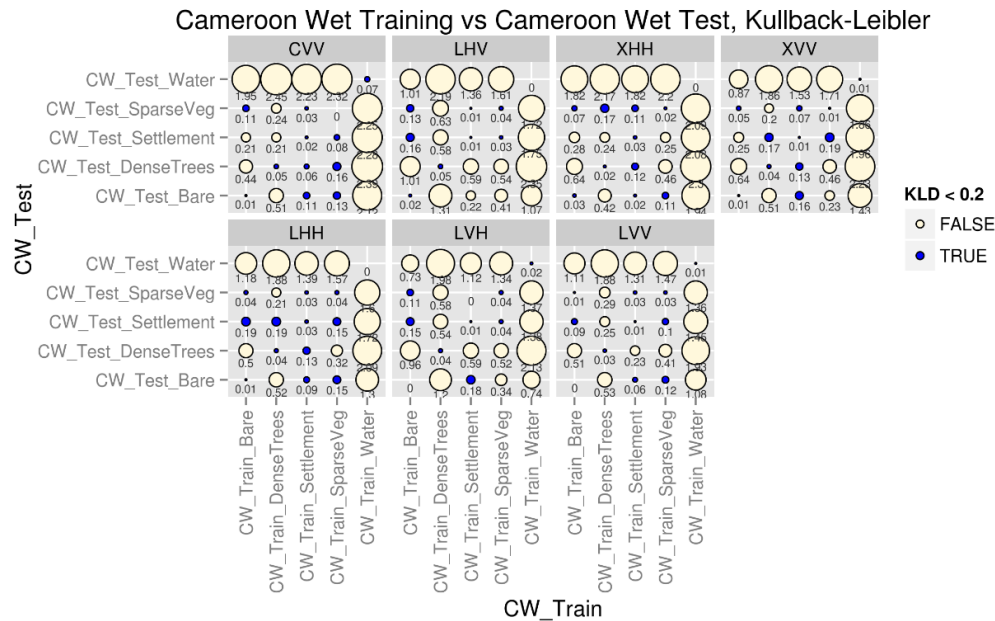


Figure 7.26 The KLD of the backscatter coefficients for the land cover classes of the Cameroon wet training vs test samples.

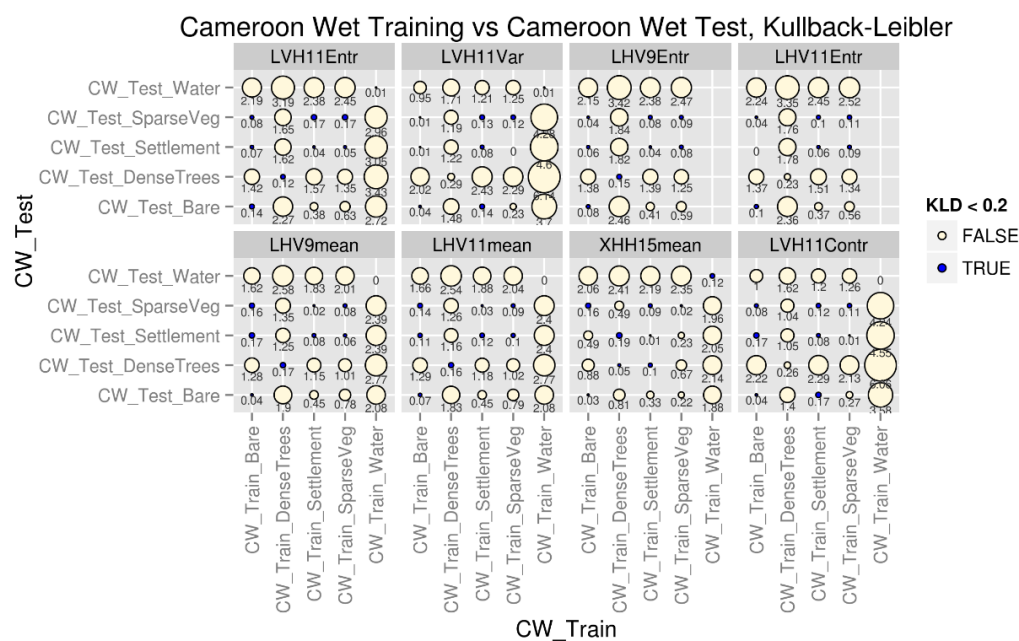


Figure 7.27 The KLD of the top 8 texture measures for the land cover classes of the Cameroon wet training vs test samples. The order of the top 8 variables is based on the MDG variable importance measure from the random forest model.

### 7.3.3.2 KLD for Cameroon wet vs dry season samples

Figure 7.28 shows the KLD measures of the backscatter coefficients for the five land cover classes (*bare, dense trees, settlement, sparse vegetation* and *water*) for the Cameroon wet vs dry season samples. Figure 7.29 shows the KLD measures of the top 8 texture measures for the Cameroon wet vs dry land cover samples. In both the backscatter coefficient (Figure 7.28) and texture measure (Figure 7.29) KLD measures plots, there is more of a deviation from the ideal scenario of Figure 7.25, compared to the training vs test samples KLD plots in Figures 7.26 and 7.27. In these plots, the diagonal values (see Figure 7.25) shows the overlap of the statistical distributions each land cover class between the wet and dry season data. The off-diagonal values shows the difference between the land cover classes. It is noted that there is a larger difference between the wet and dry season data, especially for the bare and water classes. For some of the KLD measure, some of the calculations returned NULL values. This is especially noted for the X-HH Variance (15 window size) texture measure and in the L-HH Entropy (11 window size). Investigating the data distributions in Figure 7.7, it is noted that the distributions were less clear than the other texture measures, which is an indication of why the NULL values were returned.

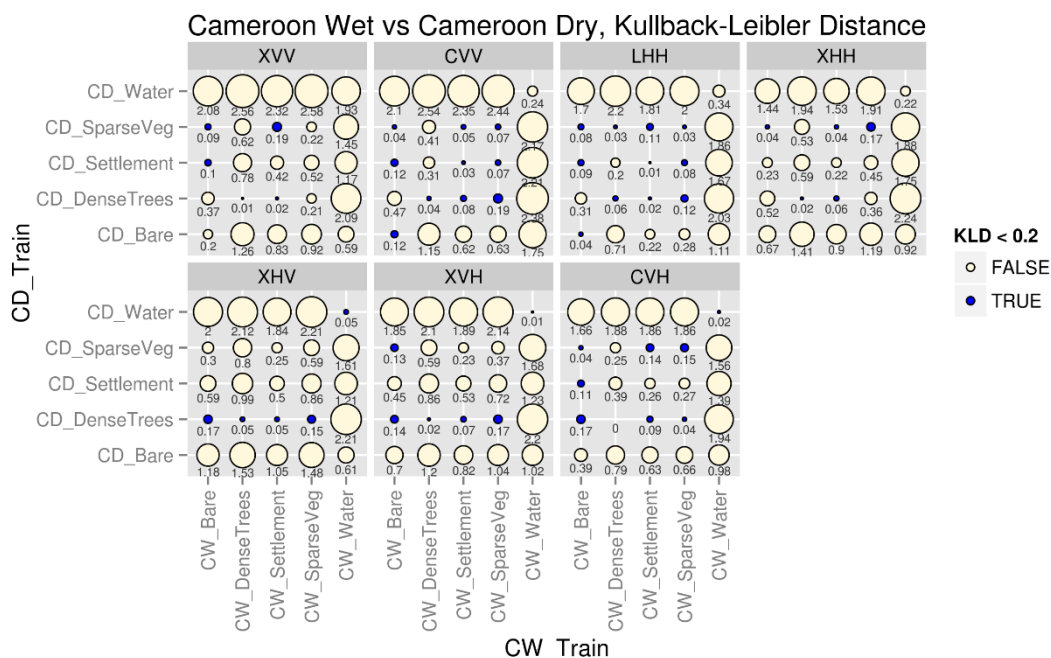


Figure 7.28 The KLD of the backscatter coefficients for the land cover classes of the Cameroon wet vs dry season samples.

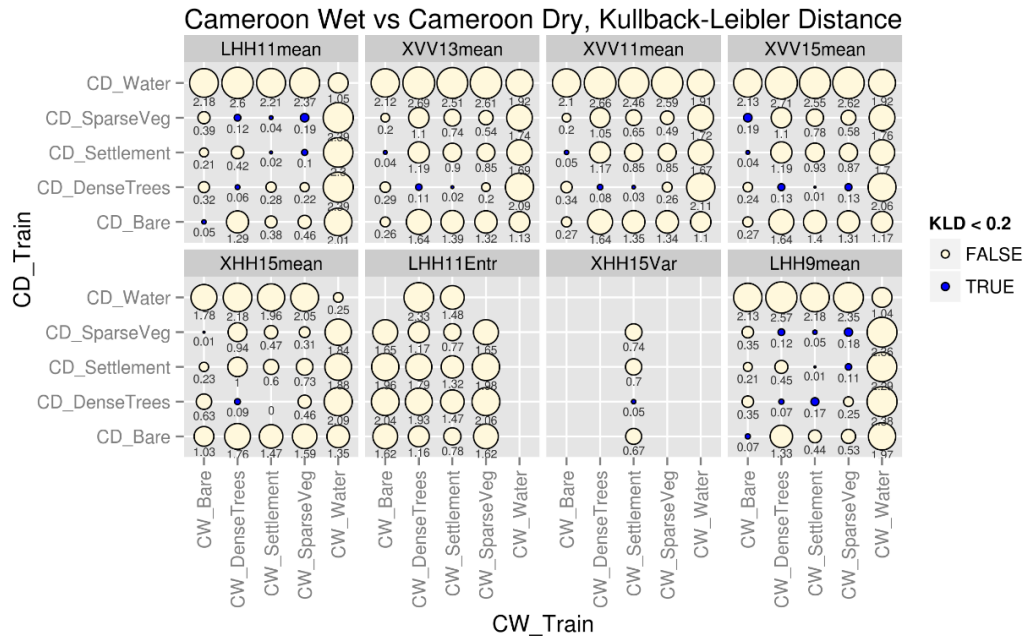


Figure 7.29 The KLD of the top 8 texture measures for the land cover classes of the Cameroon wet vs dry season samples. The order of the top 8 variables is based on the MDG variable importance measure from the random forest model.

### 7.3.3.3 KLD for Cameroon vs DRC wet season samples

Figure 7.30 shows the KLD measures for backscatter coefficients of the five land cover classes (*bare, dense trees, settlement, sparse vegetation and water*) of the Cameroon vs DRC wet season samples. Figure 7.31 shows the KLD measures for the texture measures of the land cover classes of the Cameroon vs DRC wet season samples. There is more of a variation here, with most of the backscatter coefficient and texture measure KLD measures diverging from the ideal scenario in Figure 7.25. For the X-HH Correlation (15 and 13 window sizes) and the X-HH Entropy (15 window size), several NULL values were calculated. The distribution plots in Figure 7.11 shows the statistical distributions of the land cover classes, and gives an indication as to why NULL values were returned for the KLD measures.

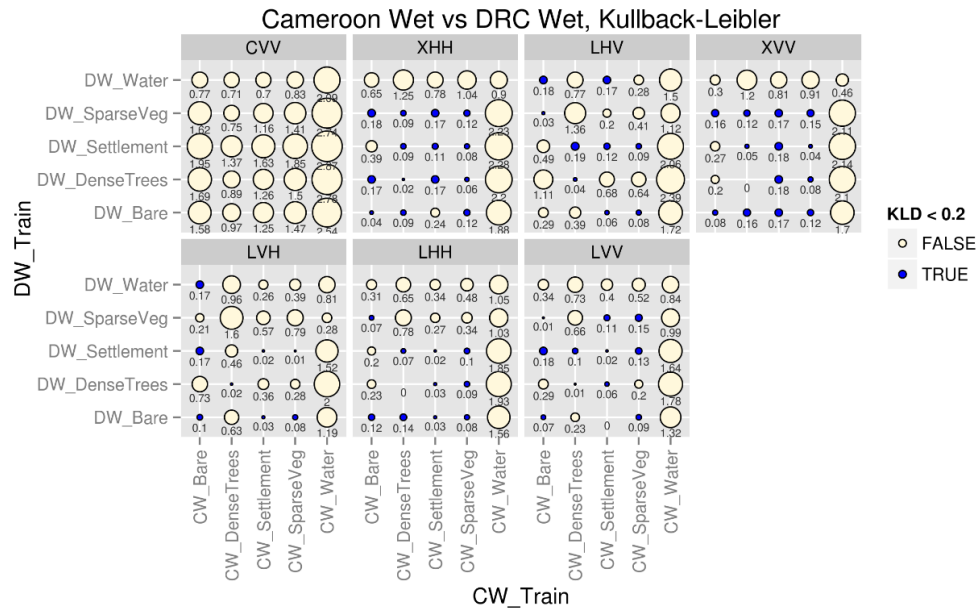


Figure 7.30 The KLD of the backscatter coefficients for the land cover classes of the Cameroon vs DRC wet season samples.

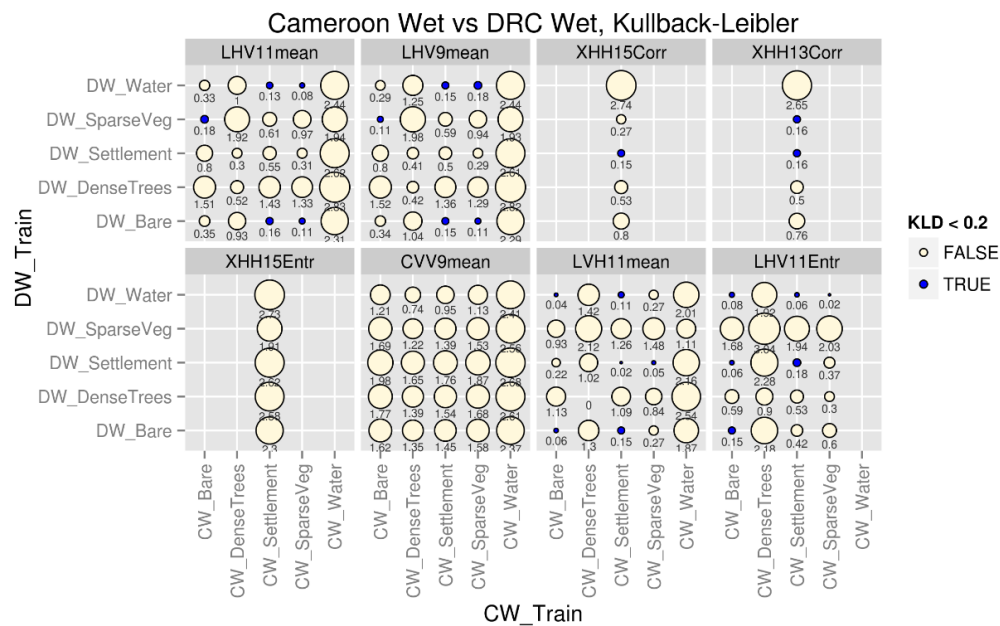


Figure 7.31 The KLD of the top 8 texture measures for the land cover classes of the Cameroon vs DRC wet season samples. The order of the top 8 variables is based on the MDG variable importance measure from the random forest model.

### 7.3.4 KLD for land cover classes of the semi-arid ecoregion (Tanzania and Chad)

In this section the KLD measures are compared for the land cover classes of the semi-arid sites. The comparison is between the Tanzania dry season training vs test samples, the Tanzania dry vs wet season samples and between the Tanzania vs Chad dry season samples. Similar to the forested sites, the types of variables considered are the backscatter coefficients and the top 8 texture measures with corresponding figure numbers given in Table 7.6. The order of the texture measures in this table is given by the MDG variable importance measures during random forest model development in Chapter 6.

*Table 7.6 The site comparison using KLD for the semi-arid sites with corresponding figure numbers.*

Site comparison using KLD	Type of variables	Figure number
Tanzania dry season training vs test	Backscatter coefficients	7.32
	Texture measures (top 8)	7.33
Tanzania dry vs wet season	Backscatter coefficients	7.34
	Texture measures (top 8)	7.35
Tanzania dry season vs Chad dry season	Backscatter coefficients	7.36
	Texture measures (top 8)	7.37



### 7.3.4.1 KLD for Tanzania dry season training vs test samples

Figure 7.32 shows the KLD measures of the backscatter coefficients of the five land cover classes (*agriculture, bare, dense trees, settlement and sparse vegetation*) for the Tanzania dry training vs test samples. Figure 7.33 shows the KLD measures of the texture measures for the Tanzania dry training vs test samples. Although the diagonal elements show a large overall between training and test data, the off-diagonal elements also show low KLD values, indicating a small difference between the statistical distributions of the various land cover classes. The KLD measures of the training vs test samples are shown as a baseline measurement, to compare the inter-season (Figures 7.34 and 7.35) and inter-site (Figures 7.36 and 7.37) comparisons to.

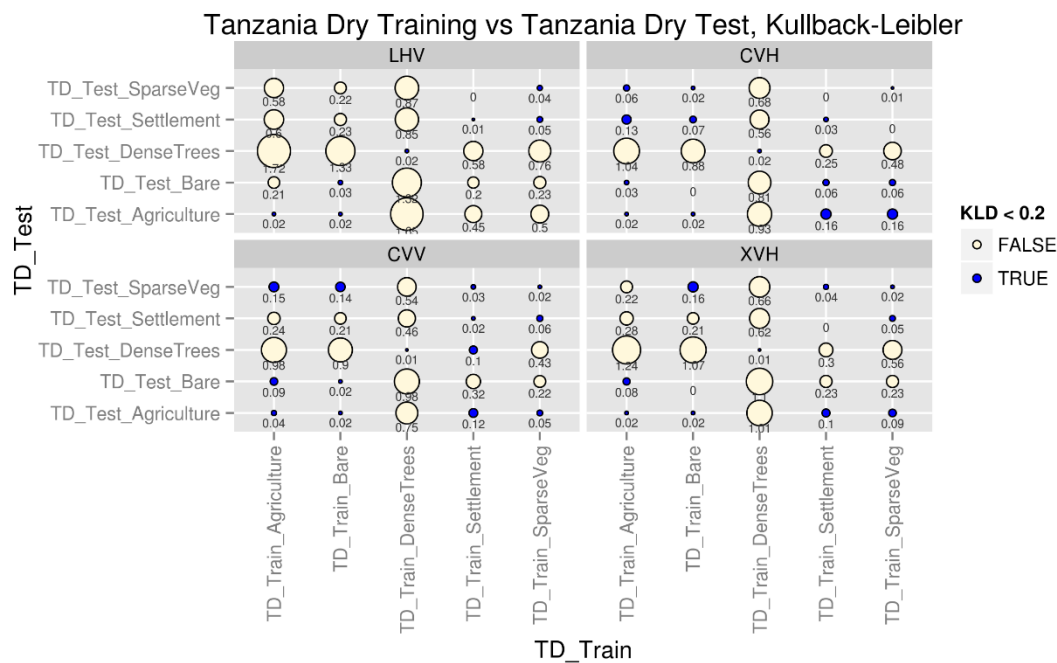


Figure 7.32 The KLD of the backscatter coefficients for the land cover classes of the Tanzania dry season training vs test samples.

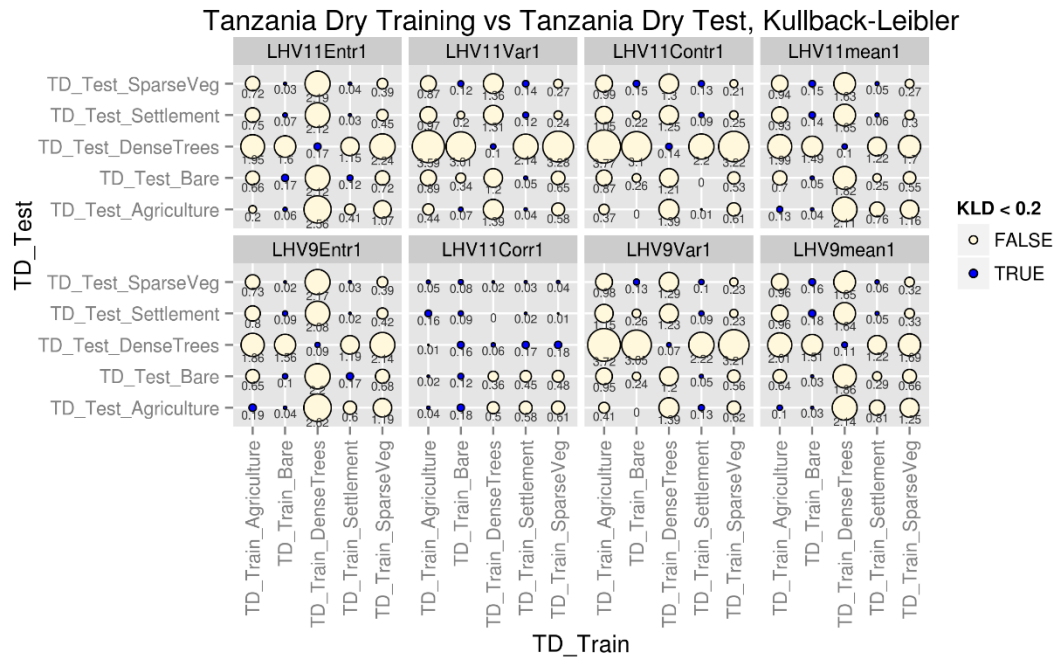


Figure 7.33 The KLD of the top 8 texture measures for the land cover classes of the Tanzania dry season training vs test samples. The order of the top 8 variables is based on the MDG variable importance measure from the random forest model.

#### 7.3.4.2 KLD for Tanzania dry vs wet season samples

Figure 7.34 shows the KLD measures of the backscatter coefficients of the five land cover classes (*agriculture, bare, dense trees, settlement and sparse vegetation*) for the Tanzania dry vs wet season samples. Figure 7.35 shows the KLD measures of the texture measures of the five land cover classes for the Tanzania dry vs wet season samples. The diagonal elements in Figures 7.34 and 7.35 shows that there is a deviation between the statistical distributions of the land cover classes of the dry and wet season data. The small values of the off-diagonal elements, indicate a large overlap between the different land cover classes, which will diminish the ability to classify the land cover classes correctly.

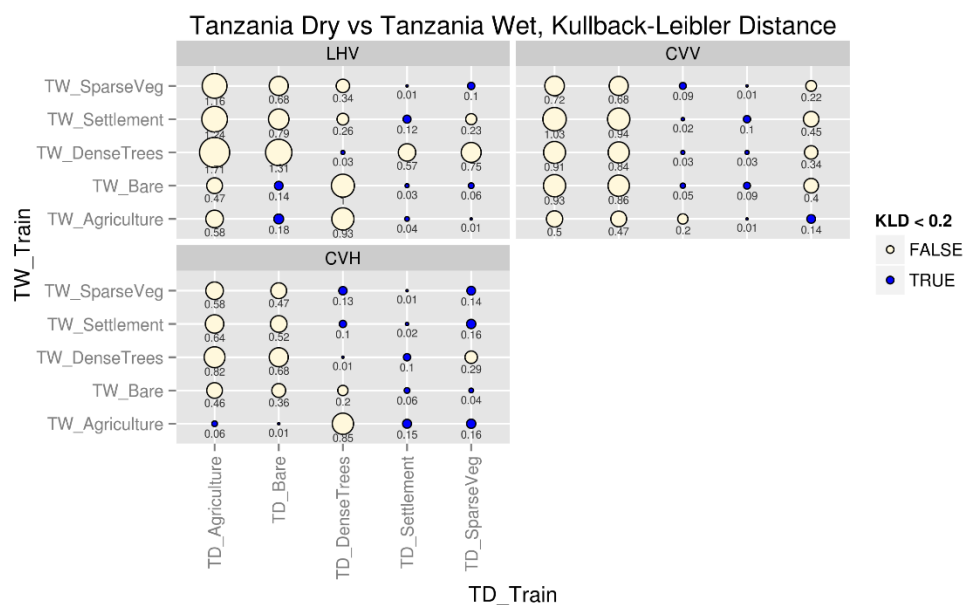


Figure 7.34 The KLD of the backscatter coefficients for the land cover classes of the Tanzania dry vs wet season samples.

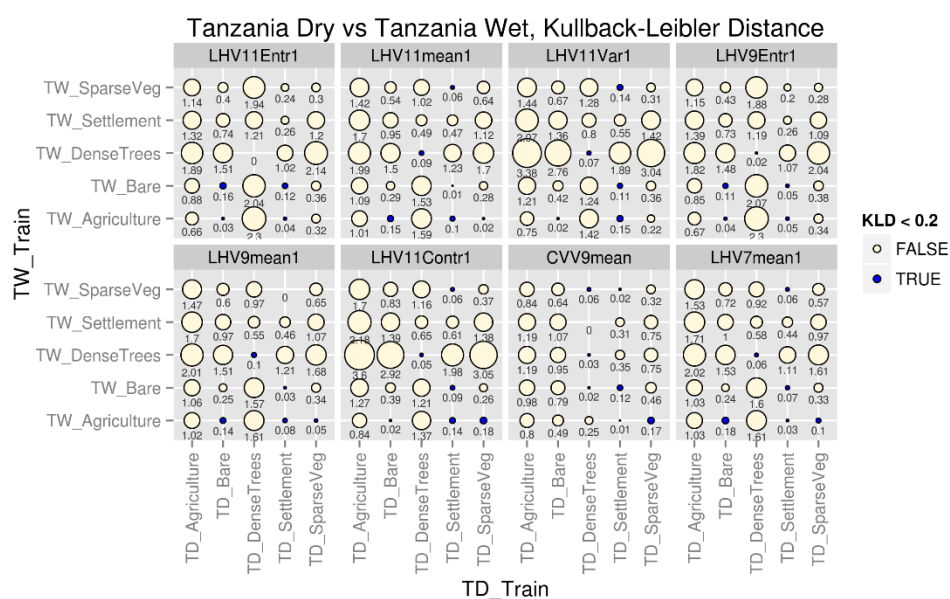


Figure 7.35 The KLD of the top 8 texture measures for the land cover classes of the Tanzania dry vs wet season samples. The order of the top 8 variables is based on the MDG variable importance measure from the random forest model.

### 7.3.4.3 KLD for Tanzania vs Chad dry season samples

Figure 7.36 shows the KLD of the backscatter coefficients of the five land cover classes (*agriculture, bare, dense trees, settlement* and *sparse vegetation*) of the Tanzania vs Chad dry season samples. Figure 7.37 shows the KLD measures of the texture measures of the land cover classes for the Tanzania vs Chad dry season samples. The larger values of the diagonal elements indicate a deviation of the statistical distributions of the land cover classes between the Tanzania and Chad dry season data. Most of the land cover classes seem to be quite different between the two sites, with the *bare* class and *settlement* classes being the most similar. The KLD measures of the L-HH Entropy (11 window size) texture measures produced NULL values, and the distribution of the values in Figure 7.23 shows why it could be difficult to give an indication of the separation between the different statistical distributions.

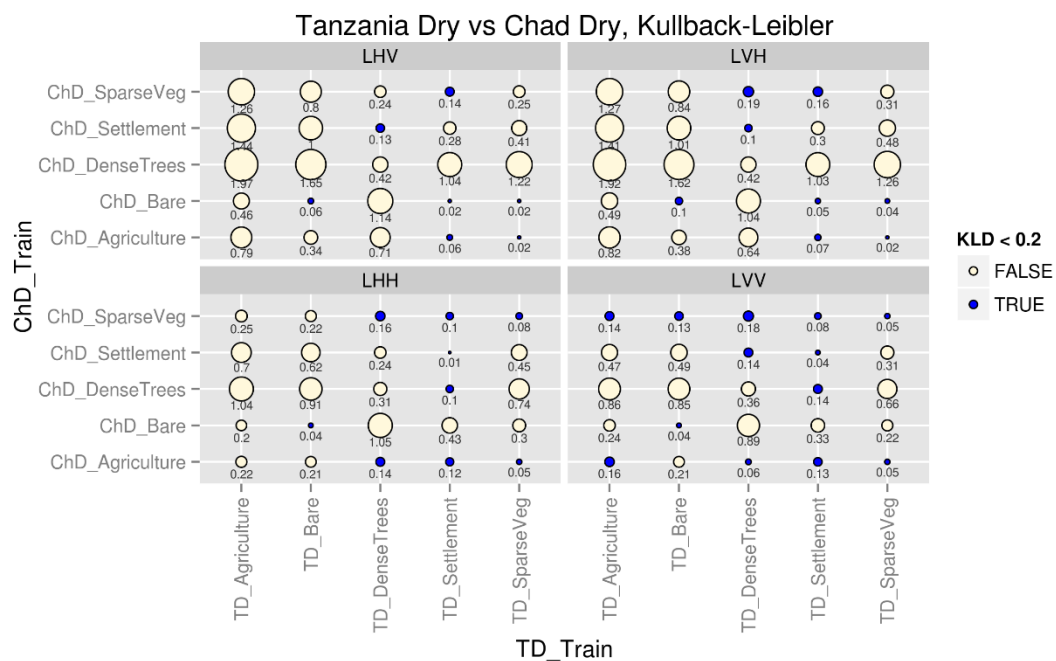


Figure 7.36 The KLD of the backscatter coefficients for the land cover classes of the Tanzania vs Chad dry season samples.

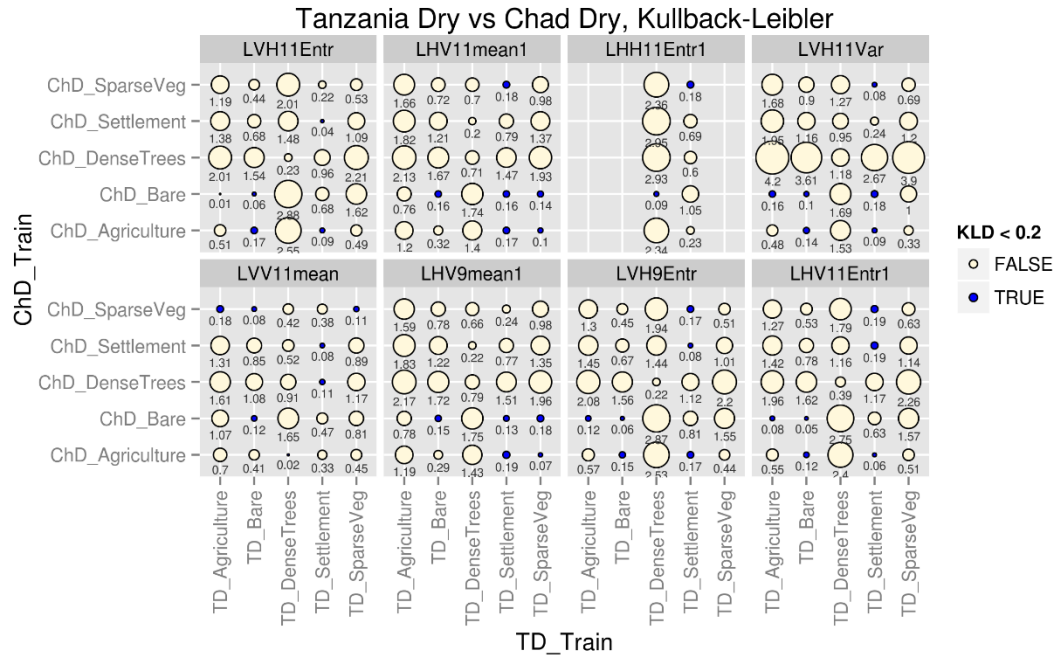


Figure 7.37 The KLD of the top 8 texture measures for the land cover classes of the Tanzania vs Chad dry season samples. The order of the top 8 variables is based on the MDG variable importance measure from the random forest model.

### 7.3.5 KLD class difference vs random forest classification user's accuracy

To assess whether the KLD measures can be used to indicate the expected classification accuracy, the KLD class difference is calculated as follows:

First, the averages of the KLD matrices in the graphs in Sections 7.3.3 and 7.3.4 are calculated for each of the site comparison scenarios leaving, one matrix of KLD averages. In the site-comparisons in Sections 7.3.3 and 7.3.4, the number of matrices varied between 3 and 8. The average of the matrices  $X_i$  with  $i = 1$  to  $N$  and  $N$  the number of matrices present, are calculated as in Equation 7-2:

$$\frac{\sum_{i=1}^N X_i}{N} \quad \text{Equation 7-2}$$

For this calculated matrix, the difference between the diagonal elements and the average of the off-diagonal elements is calculated for each row, resulting in the KLD class differences. The matrix of average KLD measures are represented in Figure 7.38.

Secondly, for each row element, the average of the off-diagonal elements is calculated. The diagonal element are then subtracted from the average off-diagonal elements, for each row, resulting in the KLD class difference for each land cover class.

The results are represented in a confusion matrix between the actual values and the predicted values, as in Table 7.7 (Fielding and Bell, 1997).

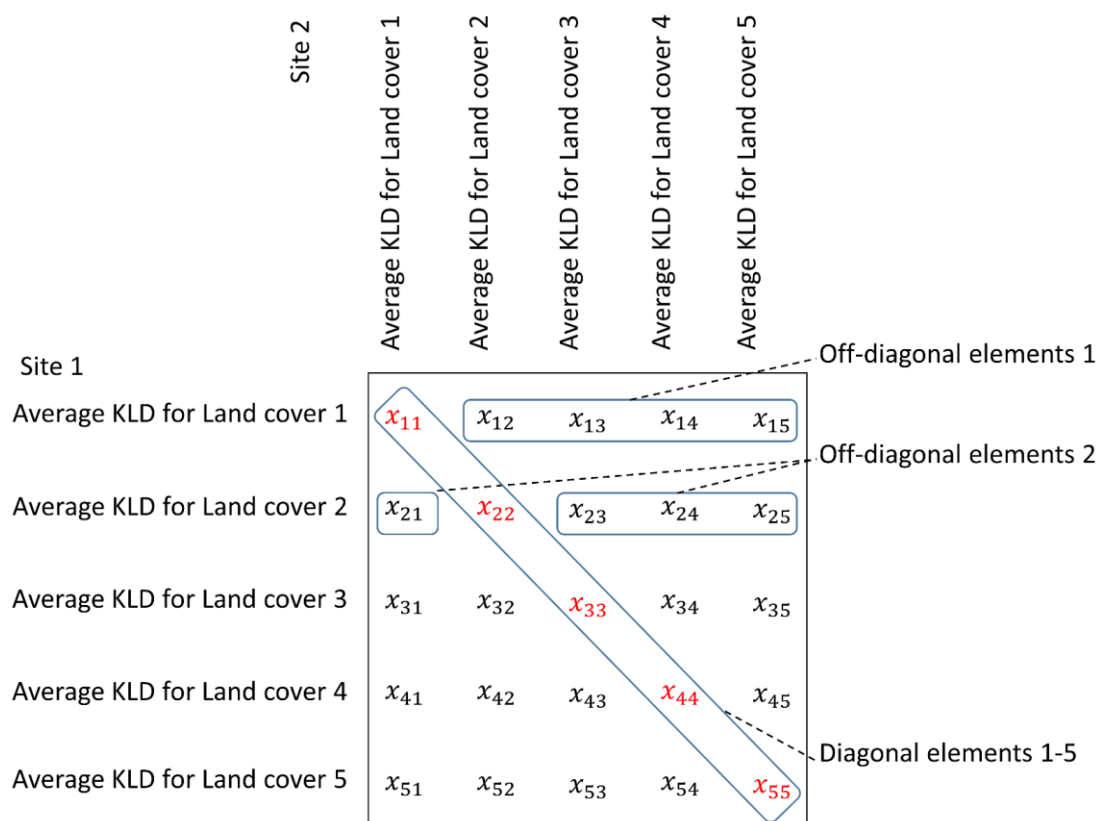


Figure 7.38 An outline of the diagonal and off-diagonal elements of the matrix of the KLD averages.

Table 7.7 A confusion matrix between reference values and predicted values.

Predicted values	Reference values		
		+	-
	+	a	b
	-	c	d

where

- (a) is the number of times the predicted classification matched the actual reference class.
- (b) is the number of times the samples were predicted to be X and were not observed to be X in the reference values.
- (c) is the number of times the samples were predicted to be something other than class X when the reference class was class X.
- (d) is the number of times the samples were predicted not to be X, when the reference class was also observed not to be X.

There are several measures of accuracy from the confusion matrix results. Two of the most commonly used accuracy measures are the producer's accuracy and user's accuracy. The producer's accuracy is also referred to as the positive predictive power and is given in Equation 7-3. This is the accuracy measure that is used in Chapter Five and Six. The user's accuracy is also known as the sensitivity and is given in Equation 7-4.

$$\text{Producer's accuracy} = \frac{a}{a + c} \quad \text{Equation 7-3}$$

$$\text{User's accuracy} = \frac{a}{a + b} \quad \text{Equation 7-4}$$

It can therefore be seen that the calculation of the KLD class difference is similar to the calculation of the user's accuracy from the confusion matrix of the classified results. A regression line is fitted to the KLD class differences for each site comparison, to determine whether the KLD class difference can be used as an indication of the expected user's accuracy.

This expected user's accuracy would be the results from transferring a random forest model developed on one site to an additional site.

The KLD class difference compared to the user's accuracy of the (up to) top 8 variables for the backscatter coefficients (dB) and texture measures, with a regression line fitted for each scenario are given in Figure 7.39 for the forested sites and in Figure 7.40 for the semi-arid sites.

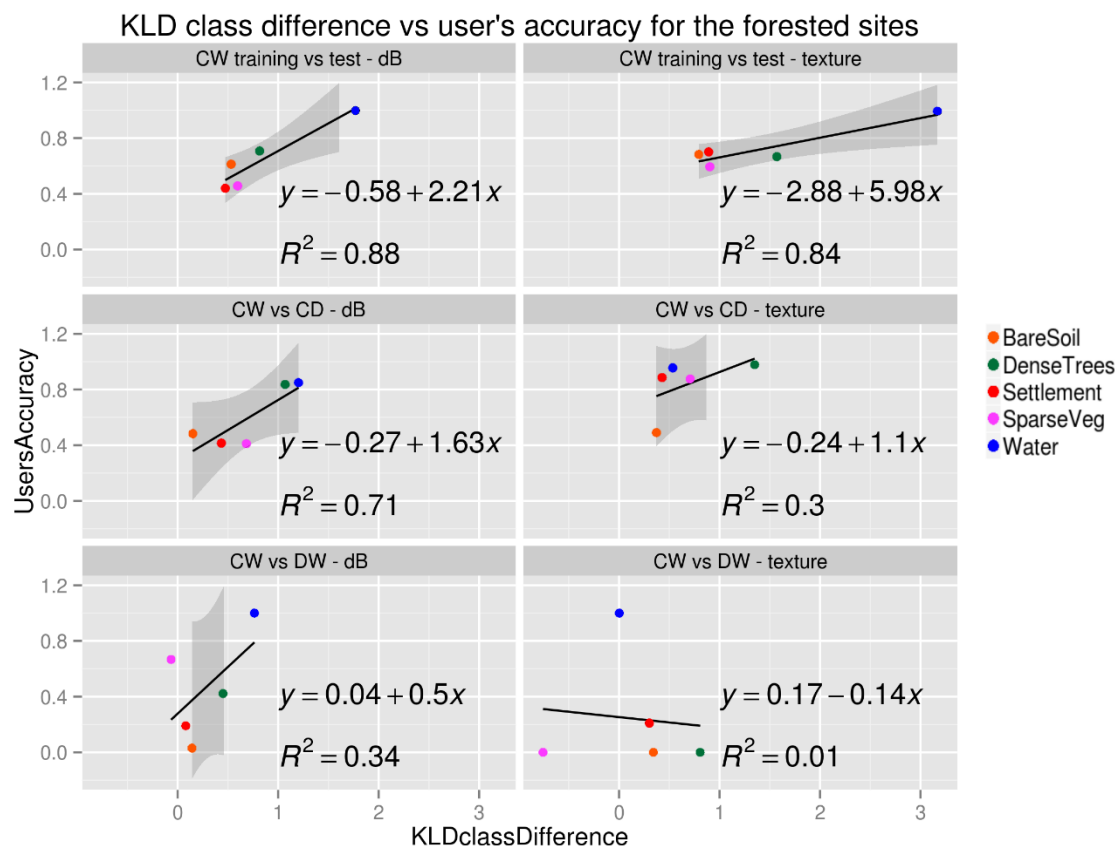


Figure 7.39 The KLD class difference compared to the user's accuracy of the random forest models using at most the top 8 backscatter coefficient channels and the top 8 texture measures for the forested sites.



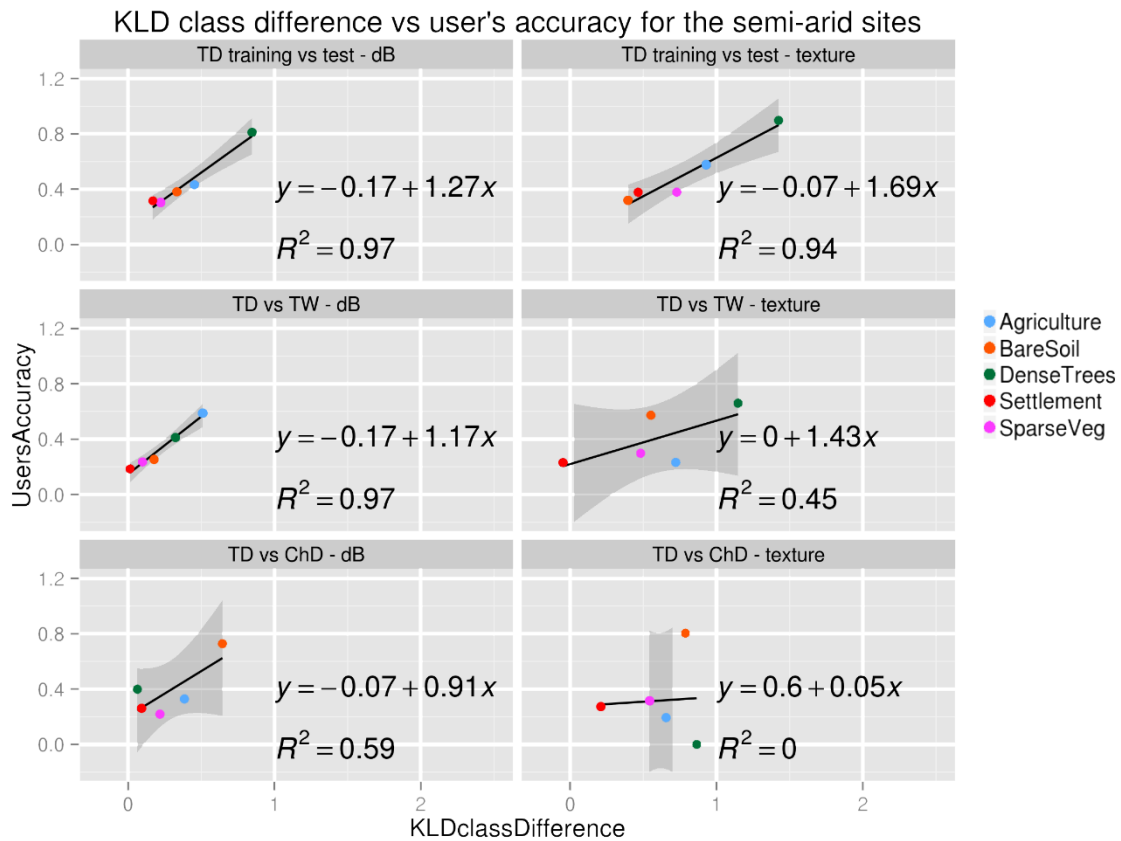


Figure 7.40 The KLD class difference compared to the user's accuracy of the random forest models using at most the top 8 backscatter coefficient channels and top 8 texture measures for the semi-arid sites.

For the Cameroon wet season training vs test sample backscatter coefficient and texture groups, the linear relationship between the KLD class difference and the user's accuracy has an  $R^2$  value of 0.88 and 0.84 respectively. For the Cameroon wet vs dry season backscatter coefficient group of variables, the  $R^2$  value is 0.71, which is still relatively high. However, for the Cameroon wet vs dry season texture measure variables, the  $R^2$  value is only 0.3, and for the Cameroon wet season vs DRC wet season backscatter coefficient and texture measure variables, the  $R^2$  values are 0.34 and 0.01 respectively.

For the Tanzania dry training vs test season backscatter coefficient and texture groups, the linear relationship between the KLD class difference and the user's accuracy has an  $R^2$  value of 0.97 and 0.94 respectively. For the Tanzania dry vs wet season backscatter coefficient variables, the  $R^2$  value is 0.97, which is close to perfectly linear. However, similar to the Cameroon site, the  $R^2$  value for the Tanzania dry vs wet season texture measure variables is much lower, at 0.45. The Tanzania dry season vs Chad dry season backscatter coefficients

scenario has a reasonably good  $R^2$  value of 0.59, with no linear relationship between the KLD class difference and user's accuracy for the Tanzania dry vs Chad dry texture measures scenario. This measure proved to be reliable for inter-seasonal comparison, but less reliable for inter-site comparison. This might indicate that the land cover classes are more different between the different sites, than the initial assessment. This method was however chosen as a promising separability index measure, and its use as part of the MaxEnt classifier.

## 7.4 Discussion

Plotting the statistical distributions of the backscatter coefficients, interchannel ratios, texture measures and elevation data are useful for getting a better insight into the data. This was undertaken for the Cameroon wet season training vs test data (Section 7.3.1.1), Cameroon wet vs dry season data (Section 7.3.1.2) and Cameroon vs DRC wet season data (Section 7.3.1.3). Similarly, for the semi-arid ecoregion, the statistical distributions were compared for the Tanzania dry season training vs test samples (Section 7.3.2.1), Tanzania dry vs wet season data (Section 7.3.2.2) and Tanzania vs Chad dry season data (Section 7.3.2.3).

Based on the SAR distribution plots in Section 7.3.1 and Section 7.3.2, most of the classes appear not to be very separable. In reality, however, the classification algorithm (be it a random forest model or an SVM) are classifying the land cover classes in an  $n$ -dimensional space (where  $n$  is the number of input bands), and better separability is therefore achievable.

Considering the Cameroon wet season training vs test sample graphs (Section 7.3.1.1), it is noted that the statistical distributions of the different land cover classes are very similar for all the groups of variables, namely, backscatter coefficients, interchannel ratios, texture measures and elevation data. For the Cameroon wet vs dry season comparison (Section 7.3.1.2), there is a shift of the statistical distributions of the land cover classes between the wet season and dry season samples. This difference between the wet and dry season is even more prominent for the texture measures. This leads to asking the question of whether there is enough similarity between the wet and dry season data distributions to be able to develop a model on the wet season data and apply it to the dry season data? As was seen in Chapter

6, this does seem to be the case, and resulted in a classification accuracy of 74% for the Cameroon wet season model applied to the Cameroon dry season data, compared to 88% for the Cameroon wet season training vs test data (see Table 6.3).

Taking it one step further, the statistical distributions of the land cover classes for the Cameroon wet season were compared to the DRC wet season samples in Section 7.3.1.3. Here it was noted that there is substantial difference between the land cover classes of the Cameroon wet season and DRC wet season samples. This leads to the question of whether a measure between the land cover classes can give an indication of the potential classification accuracy of a model developed on the Cameroon wet season data, and applied to the DRC wet season data? To achieve this, the KLD was calculated for all Cameroon wet training vs test samples (Section 7.3.3.1), Cameroon wet vs dry season samples (Section 7.3.3.2) and Cameroon wet season vs DRC wet season samples (Section 7.3.3.3). From the KLD measures, the KLD class difference was calculated and correlated with user's accuracy measure. The results were that the KLD class difference measure proves to work well for the training vs test backscatter coefficient and texture measure datasets ( $R^2$  of 0.88 and 0.84 respectively). The Cameroon wet vs dry dataset also had a high correlation ( $R^2$  of 0.71). However, the correlation using the Cameroon wet vs dry texture measures ( $R^2$  of 0.3), inter-site comparison of Cameroon wet vs DRC wet backscatter coefficients ( $R^2$  of 0.34) and texture measures ( $R^2$  of 0.01), did not yield satisfactory results.

The relationship between KLD class difference and the user's accuracy works well for the backscatter coefficient scenarios between the different seasons, but less well for the backscatter coefficient inter-site comparison. Other separability indexes will therefore need to be tested, to find a good measure to assess whether the statistical distributions between one site and another are too far apart for a model to be developed on the one site, and applied to the other site. One finding is that the KLD class difference performs better on the backscatter coefficients than on the texture measures. An explanation for this could be that the KLD measure works well for single modal distributions between two Gaussian distributions, such as the backscatter coefficients (Figure 7.10), but does not work well for multi-modal distributions, such as the texture measure distributions (Figure 7.11). From the

texture measures it appears that the mean texture measure was calculated with an integer instead of a numeric (with decimal values) output. It is noted that this might have contributed to the KLD class difference not working successfully on the texture measures, and will have to be corrected and re-assessed in future calculations.

Upon visual interpretation of the KLD measures, it is seen that the Cameroon wet season training vs test samples have low values for the diagonal elements ( $<0.2$ ) and in most cases higher KLD values for the off-diagonal elements (Section 7.3.3.1). However for the Cameroon wet season vs dry season, the KLD plots diverges further from the ideal scenario (Figure 7.25), for the backscatter coefficients, and especially for the texture measures (Section 7.3.3.2). The KLD measures for the Cameroon wet season vs DRC wet season diverges even further from the ideal scenario (Figure 7.25) for both the backscatter coefficients and texture measures (Section 7.3.3.3). In some instances the KLD calculation resulted in NULL-values.

The semi-arid site follows a similar pattern to the forested site, with the KLD measures low (similar distributions) for the diagonal elements, and higher (dissimilar distributions) for the off-diagonal elements for the Tanzania dry training vs test samples, for both the backscatter coefficients and texture measures (Section 7.3.4.1). For the Tanzania dry vs wet season KLD measures, some of the diagonal elements have higher KLD measures, showing that the statistical distributions of some of the land cover classes are dissimilar between the Tanzania dry and wet seasons. Similarly, for the off-diagonal elements, some of the KLD measures are low, showing that the distributions of the land cover classes are similar, which should ideally be dissimilar, to enable identification of the different land cover classes, for both the backscattering coefficient and texture measures (Section 7.3.4.2). It is also noted again that no X-band data were available for analysis of the Tanzania wet season data. Therefore, in the comparison between the Tanzania dry and wet season data, no X-band data were assessed. The KLD measures between the Tanzania dry season and Chad dry season samples show some diagonal elements with larger KLD measures, showing dissimilar distributions, and some off-diagonal elements with lower KLD values, showing similar statistical distributions between the different land cover classes (Section 7.3.4.3). It is noted that in this comparison, only L-

band data were available for overlap between the Tanzania dry and Chad dry season data (see Table 3.4).

Drawing linear regression lines between the KLD class difference measures and the user's accuracies for the semi-arid sites resulted in higher  $R^2$  values than for the forested sites. The KLD class difference for the Tanzania dry season training vs test samples yielded an  $R^2$  value of 0.97 for the backscatter coefficients and 0.94 for the texture measures. For the Tanzania dry season vs wet season, the backscatter coefficients also yielded an  $R^2$  of 0.97, with the texture measures giving an  $R^2$  value of 0.45. For the Tanzania dry vs Chad dry season backscatter coefficient comparison, the KLD class difference compared to the user's accuracy gave an  $R^2$  value of 0.59, which is much higher than the  $R^2$  value of 0.34 for the Cameroon wet vs DRC wet season backscatter coefficient relationship. For the Tanzania dry vs Chad dry season texture measures, the linear relationship gave an  $R^2$  value of 0.0, showing no linear relationship. For the semi-arid data, the KLD class difference of backscatter coefficients, proved to be a good indicator of the potential user's accuracy of developing the model on one set of data, and applying it to another.

Assessing the statistical distributions and KLD measurements in such a detailed way, manually and visually, highlighted the effectiveness of machine learning classification algorithms for feature selection and classification. Probably the most significant benefit of the random forest algorithm, as used in this study, is the measures of variable importance, which was used to quickly highlight features which result in the highest classification accuracy results (Chapter 5). Machine learning allows the researcher to quickly work through a multitude of input feature layers (in this study between 100-250 feature layers for each site), in an automated and meaningful way, without having to work through each layer manually and visually. The algorithm therefore identifies the most prominent features, and allows the researcher to select the features most prominent for a specific application: in this case land cover classification.

## 7.5 Conclusion

The KLD class difference gives a good indication of the expected user's accuracy when the backscatter coefficients are used directly for both ecoregions. For the inter-season backscatter coefficient comparison, the KLD class difference gave an  $R^2$  value of 0.71 for the user's accuracy of the Cameroon wet vs dry season, and an  $R^2$  value of 0.97 for the user's accuracy of the Tanzania dry vs wet season comparison. For this application, the KLD class difference measure proved accurate and useful.

When applying the KLD class difference to the backscatter coefficients of a new geographical site, the semi-arid sites performed better, with an  $R^2$  value of 0.59 compared to the forested sites with an  $R^2$  value of 0.34. In this application, the KLD class difference measure proved therefore less accurate, than anticipated. The KLD class difference measures of the texture measures were also less accurate than the backscatter coefficients ( $R^2$  value 0.3 and 0.45 for the inter-season comparison of the Cameroon and Tanzania sites respectively), most likely due to the multi-modal statistical distributions of the texture measures.

Even though the KLD class difference gave a high relationship for the inter-site comparison for the semi-arid sites, the results for the forested site, is quite low. There is, therefore, the opportunity for more research to investigate other separability indices, such as the Fisher criterion and the Bhattacharyya distance. These separability indices can be compared to the KLD results, as indicators to assess whether or not the statistical distributions of the land cover classes are similar enough, for the model to yield high classification accuracies on the new data.

## **Chapter Eight    Overall discussion and conclusion**

The potential of utilising multi-frequency SAR was highlighted with the first experiments from airborne SAR sensors through the last three decades, together with the land cover classification results using multi-frequency SAR from SIR-C / X-SAR images in 1994. Several studies have since looked at the use of SAR for land cover classification from single and multi-frequencies. A few studies using SAR for land cover classification that are related to this research, are Waske and Braun (2009), Turkar et al. (2012) and Naidoo et al. (2015).

Waske and Braun (2009) compared decision trees, MLC, boosted decision trees and random forests to classify land cover using multi-temporal SAR data and found that the random forest classifier outperformed the other classifiers for study sites in Bonn and Jena in Germany. Turkar et al. (2012) compared land cover classification accuracies of a mangrove study site in Mumbai using L-, C-, X-band, LC-, LCX-bands without polarimetric decompositions; and LCX-bands with polarimetric decompositions. Turkar et al. (2012) found that L-band gave the highest classification accuracies, using a single frequency, and that combining all three frequencies improved the classification accuracies for all land cover classes considered, namely water, mangrove, urban, forest, saltpans, sewage plant, wetland and grassland. Naidoo et al. (2015) investigated the use of L-, C- and X-band to classify AGB, CC and TCV of woody trees in an African savannah. Naidoo et al. (2012) concluded that the use of LCX-bands data yielded only marginally higher classification results than using L-band data on its own. The results therefore does not justify the acquisition of all three frequencies to monitor tree structure.

This research looked at using multi-frequency SAR for land cover classification of forested and semi-arid ecoregions of Africa, and addressed the three hypothesis posed in Section 1.2. The hypothesis stated that multi-frequency SAR provides a higher classification accuracy to single frequency SAR (first hypothesis), that the models are transferable between different sites and seasons (second hypothesis), and that land cover classes are similar between seasons and similar between different sites of the same ecoregions (third hypothesis).

The overall discussion covering the combined results from Chapters Five, Six and Seven, which tested the first, second and third hypotheses, is presented in Section 8.1.1, Section 8.1.2 and Section 8.1.3 respectively. Practical implications of a multi-frequency SAR approach are discussed in Section 8.2. The final conclusion of the thesis, together with further work identified and a future view are presented in Section 8.3.

## 8.1 Overall discussion

### 8.1.1 Findings from the random forest and SVM classification of multi-frequency SAR over Cameroon and Tanzania study sites

The discussion in this section expands on that in Chapter Five, in which random forest models of different input variable scenarios are compared. A summary of the overall classification accuracies, using all available variables, is given in Table 5.2. From this summary, several conclusions can be made, as discussed in the following sections.

#### 8.1.1.1 Single vs dual vs three frequencies

With the inclusion of texture measures in the random forest model, single frequency SAR gives overall classification accuracy results very similar to dual-frequency and all three SAR frequencies. The question is rather which single frequency to use? From the results of this study, as a single frequency, L-band provided the highest overall classification accuracy, when considering both forested and semi-arid eco-regions simultaneously, with results comparable to combining two or three frequencies. Should L-band images not be available, X-band images would be the second choice, with C-band the third choice for using single frequency SAR for land cover classification. When the forested and semi-arid sites are considered separately, X-band provides the highest overall classification accuracies for the forested ecoregion, when both wet and dry seasons are considered. L-band provides the highest overall classification accuracies for the semi-arid ecoregion, for both dry and wet seasons.

The case for utilising dual-frequency SAR, is that the overall classification results are generally higher, even though only slightly so, than the single frequency scenarios. The best dual-



frequency combination uses L- and X-bands (LX) for both ecoregions. With the current data available, this can only be stated for the Cameroon wet and dry season sites, and the Tanzania dry season site. For the Tanzania wet season site, no X-band data were available and LC-bands are recommended, based on the available data.

When comparing the highest single frequency to the dual frequency LX-bands, the overall classification accuracy for the Cameroon dry season decreases slightly, from 92% to 90% (X vs LX), and for the Cameroon wet season increases, from 78% to 88% (L vs LX). For the semi-arid ecoregion, there is only a slight increase, from 63% to 65% (L vs LX) for the Tanzania dry season. For the Tanzania wet season, there is only a marginal increase from 51% using the L-band single frequency model, to 52% using the LC-bands model. The benefit of using dual / three frequencies for the semi-arid models is therefore very small, and L-band only models lead to overall classification accuracies that are highly comparable to dual / three frequencies. This is similar to the conclusion from Naidoo et al. (2015), who found that L-band data yields sufficiently high accuracy results, and that additional frequencies do not present a large increase in overall classification accuracy for the semi-arid regions. Naidoo et al. (2015) investigated tree structure classification, contrasting to the land cover classification in this study. Turkar et al. (2012) also concluded that L-band is the recommended frequency for land cover classification, which is confirmed in this study. Turkar et al. (2012) concluded that the combination of multi-frequency data (from L-, C- and X-band) improves classification accuracy. This study found that this is the case when only backscatter coefficients are considered. However, the increase in overall classification accuracy when using dual or three frequencies, compared to single frequency, becomes marginal when texture layers are included in the models.

This study presents not only whether or not to use more than one SAR frequency for land cover classification, but which SAR frequencies to use for specific ecoregions. The findings are summarized in Table 5.2 and Table 5.3. The reference figures from Chapter Five are cross-referenced for each section (from Figures 5.1 to 5.9) in Table 5.2, to investigate individually if required. Table 5.2 presents a detailed guideline of the classification accuracies to expect for each ecoregion, using different combinations of frequencies.

The question of whether it would be beneficial to acquire data from L-, C- and X-band SAR satellites will depend on the application at hand and on the classification accuracies that are expected. The combination of LC-bands is a good alternative to LX-bands for both ecoregions, should X-band data not be available, although L-band on its own yields accuracies close to the LC-band combination.

The recommended frequency combination applicable to both ecoregions, and each of the forested and semi-arid ecoregions separately, along with the respective wet and dry seasons are listed in Table 5.2. For the forested ecoregion the overall recommendation is LX-bands, and for the semi-arid ecoregion L-band. Several alternative recommendations are given in Table 5.3, for each of the site / season scenarios.

#### 8.1.1.2 Backscatter coefficients only

Using only the backscatter coefficients for land cover classification is not recommended, as the texture measures and elevation data add in the order of 15-25% increase in overall classification accuracies. Using data from all three frequencies (LCX-bands), but without using additional calculated texture measures, gives lower overall classification accuracies than using single frequency L-band (all four site-season scenarios) or X-band data (all scenarios, excluding Tanzania wet season) that includes texture measures.

#### 8.1.1.3 Interchannel ratios

Interchannel ratios do not add much value in addition to using only the backscatter coefficients. The overall classification accuracies either stay the same, or increase/decrease by 1% when interchannel ratios are combined with backscatter coefficients. For LCX, the addition of interchannel ratios to the backscatter coefficients and texture measures did not influence the overall classification accuracies for any of the study sites. This finding is in contrast to Lönnqvist et al. (2010), who found that the addition of the interchannel ratio layers gave an increase of at least 2.2% on using only the basic backscatter coefficient

channels. It should be noted that Lönnqvist et al. (2010) included polarimetric decompositions from fully polarimetric data, which was excluded in this study.

#### 8.1.1.4 Texture measures

One of the main findings of this study is that texture measures add the most value of all the additionally calculated feature layers. The increase of 15-25% in overall classification accuracies using texture measures, compared to using only backscatter coefficients. Using all three frequencies, classification accuracies increased from 66% to 91% for Cameroon dry; from 59% to 85% for Cameroon wet; from 49% to 64% for Tanzania dry and from 32% to 49% for Tanzania wet. This confirms the findings of Peng et al. (2005), who showed an increase of 30% when texture measures were included for land cover classification, for a study site of a mountainous region in southern Patagonia. The overall classification accuracies of using only texture measures, for the best single frequency (e.g. Cameroon dry: X-band, 91%), dual frequency (e.g. Cameroon dry: CX, 92%) and three frequency scenarios (Cameroon dry: LCX, 91%), are very close to the overall classification accuracies of using all available variables (backscatter coefficients, interchannel ratios, texture measures and elevation data) (e.g. Cameroon dry: LCX using all available feature layers). Texture measures improve the overall classification accuracy of the single frequency models, more than adding additional frequencies, without using texture measures. For the top 30 variable models, for both the MDA and MDG variable importance measures, all top 30 variables are texture measures, as shown in Section 5.3.3 (Note: elevation data were excluded at this point).

#### 8.1.1.5 Elevation data

Elevation data increases the overall classification accuracy for both ecoregions and for both seasons for each ecoregion. The main increase in overall classification accuracy is when elevation data are added to the backscatter coefficient only models. This leads to an increase in the region of 5-20% for the single frequency models and an increase of 1-10% for the dual frequency models. The addition of the elevation data to the models of backscatter coefficients, interchannel ratios and texture measures, still adds up to 5% increase in the

overall classification accuracies. This confirms the findings from Peng et al. (2005) who found that using elevation, slope and aspect data from a DEM increases the overall land cover classification significantly. The transferability of the elevation data does not work when there are elevation differences between two sites, as explained further in section 8.1.2.

#### 8.1.1.6 Dry season vs wet season

The dry season images gave higher overall classification accuracies (Cameroon: 92% for the dry season and 88% for the wet season; Tanzania: 65% for the dry season and 52% for the wet season) than the wet season images for both ecoregions. The forested dry season study site had a maximum overall classification accuracy of 92%, using X-band only (90% using LX-bands) with all feature layers. In comparison, the forested wet season study site had a maximum overall classification accuracy of 88%, using LX-bands with all feature layers. The semi-arid dry season study site had a maximum overall classification accuracy of 65%, using LX-bands with all feature layers, compared to the semi-arid wet study site which only had an overall maximum classification accuracy of 52% using LC-bands.

Wet conditions, including rainfall events, correspond to a higher SAR backscatter return as a result of higher effective vegetation water content and higher soil moisture (Lucas et al., 2010). In bare soil, higher soil moisture relates to an increased observed backscatter coefficient, due to increased surface scattering (Hobbs et al., 1998). This is however the case, until there are pools of water, when the backscatter coefficient will decrease, due to the full signal being reflected away from the SAR sensor. With vegetation, wetter conditions influence different scattering mechanisms. It is a combination of surface scattering, from the surface, and volume scattering from the stems and canopy, that are affected by wet conditions. The result is often still an increase in backscatter, but less drastic than for bare soils (Hobbs et al., 1998).

As a result, the different types of vegetation are better distinguished within the dry season images. Additionally the land cover classes are more similar to each other, as viewed by the SAR signal, in the wet season, due to higher water content throughout the landscape in this

season. The vegetation is also denser during the wet season in all the land cover classes, i.e. more dense vegetation in the *agriculture areas*, *bare soil areas*, *sparse vegetation*, *open woodland* and *dense trees*, and even within the *settlement* areas. It is therefore harder to separate the land cover classes with the SAR signal, since all the land cover classes are more alike during the wet season, than in the dry season. It has been shown that SAR backscatter ratio, called the Radar Vegetation Index (RVI), is correlated to NDVI, and can be used similar to NDVI to monitor vegetation growth (Kumar et al., 2015). Kumar et al. (2015) showed a strong correlation between NDVI and RVI, calculated from C-band HH/HV Radarsat-2 data. Towards the end of the growing season, NDVI saturates before the RVI, however, RVI can not replace NDVI completely since it can not represent greenness directly as NDVI does.

#### 8.1.1.7 Forested vs semi-arid

For all scenarios, the forested site (Cameroon) has higher overall classification accuracies than the semi-arid site (Tanzania). One way to improve the results of the semi-arid region could be to use object-based image analysis (OBIA), such as in Nascimento et al. (2012) and Evans and Costa (2013) for wetland regions, instead of pixel-based classification as used in this study. The benefits of OBIA are that it uses non-spectral features such as shape, size, smoothness, and the spatial relation between objects (Blaschke, 2010; Blaschke et al., 2014; Dronova, 2015). Some aspects such as texture analysis, which is often used as part of OBIA, was used in this study, although overall more standard pixel-based approach was followed. This allows for a results comparison, should the same data be used as part of an OBIA study in the future (Tehrany et al., 2013). OBIA should also address smoothing any speckle that are prevalent in SAR imagery.

The polygons chosen for sample areas of the land cover classes are smaller for the semi-arid regions than for the forested sites, which could affect the classification accuracies. This could lead to a further study to compare the influence of the geometric size of land cover sample areas to the classification accuracies. The semi-arid regions of central Africa pose a challenge in this regard, as the *settlement* areas and *bare* areas are quite small, and scattered between the areas with *sparse vegetation* and some areas with *dense trees*.

#### 8.1.1.8 MDA vs MDG

From all the MDA vs MDG variable comparisons, there does not appear to be a substantial difference between the overall classification accuracies of the models chosen by the MDA and MDG variable importance measures. For the class-specific accuracies of the land cover classes for the top five variables, the MDA vs MDG variable comparison seem somewhat random, but stabilize with more than five variables. The list of the MDG order of variables chosen in Chapter Six, could therefore equally have been from the MDA list of variables.

#### 8.1.1.9 Random Forest vs SVM

In this study SVMs were built using the top 30 variables from the MDA and MDG variable importance measures of the random forest models. In general the SVM models gave very similar overall classification accuracies to the random forest models. The only exceptions are the SVM models built on the Tanzania dry season data, which show an increase for the top 30 MDA variables (from 61% to 68%), and for the top 30 MDG variables (from 63% to 68%). In assessing the class-specific accuracies, the main difference is that the SVM models give a higher value for the *bare soil* land cover class (MDA and MDG variables) and for the *agriculture* class (MDA variables) than the random forest models.

### 8.1.2 Findings from assessing the transferability of the random forest classification models to additional test sites

The main findings of testing the transferability of the models, as discussed in Chapter Six, is expanded on below. The transferability of models from the wet to the dry season for the forested site, and from the dry to the wet season for the semi-arid site proved to work reasonably well. The loss in overall classification accuracy was from 88% to 74%, for the Cameroon wet model applied on the Cameroon dry data, and from 65% to 51%, for the Tanzania dry model applied on Tanzania dry data. This is a useful finding, since most studies only focus on models as applied to the season for which that model is developed. Another approach would be to use images from both seasons during the model development, as in Rodriguez-Galiano et al. (2012a).

The inter-site transferability of the models did produce satisfactory results when the same land cover classes were used. The highest inter-site overall classification accuracy for the Cameroon wet model applied to the DRC wet data is 33% (using X-band with backscatter coefficient data only), compared to the baseline model of 88% (using LCX-bands with all available features). The highest overall classification accuracy for the Tanzania dry model applied to the Chad dry data, was 40% (using L-band with only texture measures), compared to 63% for the baseline model (with L-band and all available features). There is therefore still more research to be done, to develop models that are readily transferable to additional geographical sites.

One suggestion, as investigated in this study, is to select land cover samples from both the main development site and the additional site, which are here referred to as site-specific land cover classes. The model is then trained on the site-specific land cover classes, which essentially expands the list of land cover classes to a finer level. The land cover classes can then be refined to a finer level of detail within the LCCS framework. The transferability of the models using site-specific land cover classes, in this study, had high overall classification accuracies. The overall classification accuracies of the Cameroon wet model applied to DRC wet data, using class-specific accuracies, was 87% (for the LCX-model with all available variables), compared to the 88% overall classification accuracy of the baseline model (LCX-bands). The overall accuracy of the Tanzania dry model applied to the Chad dry data was 68% (using L-band with all available features), compared to 63% for the baseline model (LCX-bands). The high overall classification accuracies of the site-specific land cover class models show that the land cover classes are sufficiently different between the two sites, to be able to distinguish them as separate classes.

The transferability of elevation as a feature to classify land cover are not recommended, unless the elevation between the two sites are very similar. The elevation difference between the Cameroon (850m) and DRC (950m) sites were ~100m, and lead to a decrease in classification accuracy from 58% to 21%. The elevation difference between the Tanzania (1200m) and Chad (850m) sites were ~350m lower, and the classification accuracy decreased

from 36% to 20%. This is the case when looking only at elevation data, but the effect is much less pronounced when the texture measure features are included in the model.

### 8.1.3 Findings from the SAR signatures and KLD measures of multi-frequency SAR images

This section summarizes and expands on the discussion in Chapter Seven. The KLD class difference measure that is calculated in this study, proves to be an effective measure for indicating the expected user's accuracies in most cases. The KLD class difference measure is more aligned to the user's accuracy when the backscatter coefficients are used in the calculation, instead of the texture measures. A high linear relationship is found between the KLD class difference and the expected user's accuracies for the inter-season transfer for both the forested and semi-arid ecoregions. The inter-season transfer gave less satisfactory results with an  $R^2$  value of 0.34, for the Cameroon wet model applied on the DRC wet season data, and an  $R^2$  value of 0.59, for the Tanzania dry model applied to the Chad dry season data.

A more suitable measure for indicating whether the difference in the SAR distributions of the land cover classes between two sites, can show how transferable a model will be, still remains to be found in research.

To determine the limit of SAR distributions differing, so that a model trained on one set of data can still be applied on another dataset with different statistical distributions, is therefore still to be determined. It will be useful to derive a measure of acceptable deviation between two distributions, to give an indication that the model will still give reliable results.

The visual display of the statistical distributions of the different SAR frequencies and polarisations is useful for obtaining a deeper insight into the data. This is related to Ulaby and Dobson (1989), where the statistical distributions of various land cover classes were presented for different frequencies and polarisations, but from a ground-based radar instrument. In comparison, the data used in this study are all acquired from satellites.



## 8.2 Practical implications of a multi-frequency SAR approach

There are many practical implications that need to be considered when the use of multi-frequency SAR is considered. Currently the different SAR sensors are operated by separate governments and space agencies (e.g. TerraSAR-X by the DLR, ALOS-2 PALSAR-2 by JAXA, RADARSAT-2 by the CSA and Sentinel1- by ESA). The mission objectives of each satellite are not necessarily aligned for use in an operational multi-frequency set-up. There are inter-governmental collaborations such as the International Charter 'Space and Major Disasters', where satellite acquisitions can be commissioned to coincide closely in time and geographical area (The International Charter, 2000; Mahmood et al., 2005). For an operational multi-frequency SAR system to work, a similar agreement will be necessary, to ensure that acquisitions are coordinated.

The other possibility would be a multi-frequency SAR constellation of satellites. One such agreement is SIASGE (Sistema Italo-Argentino di Satelliti per la Gestione delle Emergenze) (Pierdicca et al., 2011) which seeks to combine imagery from the Italian Cosmo-Skymed X-band satellite and the Argentinian SAOCOM L-band satellite for disaster response applications (Giudici et al., 2010; CONAE, 2011). The SIASGE agreement is in line with the findings from this study, that LX-bands provide high overall classification accuracies for both the forested and semi-arid ecoregions.

An important aspect to note from this study is the overlap size between satellite images, when images are acquired from more than one sensor. When choosing images from multiple satellites, the overlapping area is likely to become smaller with each additional satellite image. If a larger area needs to be classified, it might be of more benefit to use two frequencies instead of three, as a larger area can be classified from the overlapping data. The area of overlap can be maximized with planned acquisitions, compared to with archive data, as used in this study.

One aspect of SAR that was not covered in this research is the impact of ascending and descending SAR acquisitions. This will most likely have a greater impact on classification results for areas of varying topography. Since the study sites in this research were relatively flat, the impact of ascending and descending acquisitions was omitted from this research.

The benefit of additional classification accuracy needs to outweigh the additional cost of acquiring images from more satellites, to motivate for the use of multi-frequency SAR. Several satellite image providers are making their images available free of charge, or at a low cost per scene. The main provider of freely distributed SAR data is ESA with Sentinel 1A data (ESA, 2012b). Additionally, ALOS-2 archive data are available at low cost (since November 2014) (RESTEC, 2014). There is, however, still the processing cost of orthorectifying the images, and calculating additional texture measure layers that need to be taken into account for a multi-frequency SAR application.

One of the main challenges in this study was to obtain accurate orthorectification for the images from all three SAR sensors. There were several images that did not orthorectify correctly, and in particular the sub-optimal orthorectification results from the X-band images for the Tanzania wet season site, resulted in an incomplete set of images for this site. One suggestion for future research is to request both the detected and SLC versions of the images, to have a relevant reference image for each SAR image.

### 8.3 Conclusion, further work identified and future view

The final conclusion, further work and future view are presented in this section.

#### 8.3.1 Final conclusion

- (1) L-band is the best single frequency for land cover classification, when considering both forested and semi-arid ecoregions.
- (2) X-band gives the highest overall classification accuracy when only forested ecoregions are considered.
- (3) L-band gives the highest overall classification accuracy when only semi-arid ecoregions are considered.
- (4) The recommended dual-frequency combination is LX-bands, and alternatively LC-bands, if no X-band images are available.

- (5) The addition of texture measures increases the overall classification accuracies between 15-25%, and it is recommended to include texture measures to obtain high classification accuracies for all land cover classification projects that use SAR data.
- (6) The inter-season transfer of models works well, with a relatively small drop in overall classification accuracy. The overall classification accuracies decreased from 88% to 74%, when the Cameroon wet season model was applied to the Cameroon dry season data, and from 65% to 51% when the Tanzania dry season model was applied to the Tanzania wet season data.
- (7) The inter-site transfer of the models did not give satisfactory results, when considering the same land cover classes.
- (8) The high classification accuracies for the Cameroon wet model applied to the DRC wet season data, showed that the land cover classes were different enough to enable their classification as site-specific land cover classes.
- (9) The KLD class difference measure provides a good linear relationship for the inter-season land cover comparison, with an  $R^2$  value of 0.71 for the Cameroon wet season vs Cameroon dry season backscatter coefficient data and an  $R^2$  value of 0.97 for the Tanzania dry season vs Tanzania wet season backscatter coefficient data.
- (10) The KLD class difference measure did not provide satisfactory results for the inter-site land cover comparison, with an  $R^2$  value of 0.34 for the Cameroon wet season vs DRC wet season backscatter coefficient data and an  $R^2$  value of 0.59 for the Tanzania dry season vs Chad dry season backscatter coefficient data.
- (11) The inclusion of all three frequencies of L-, C- and X-band for large area monitoring is not necessary, as the increase of classification accuracy does not warrant the use of all three frequencies simultaneously. The consideration would therefore be project specific as to the final choice of frequencies, depending on the ecoregion, and on class-specific classification accuracies required.

### 8.3.2 Further work identified

Further research would include:

- 1) Comparing the number of images per frequency vs classification accuracy?
- 2) Comparing the spatial resolution of images vs classification accuracy? In the data used in this study, C-band had the lowest spatial resolution of 15m, compared to the spatial resolution of 10m for the L-band data and 5m for the X-band data. How did this influence the classification accuracies?
- 3) What is the relationship between the geometric size of the land cover polygons and the classification accuracies? In this study the land cover polygons from the semi-arid Tanzania site were much smaller than the land cover polygons from the forested Cameroon site. How does this influence the overall classification accuracies?
- 4) Compare the random forest and SVM results using either different or more sample points, to assess whether the SVM model still outperforms random forest model for the Tanzania wet season site.
- 5) Investigating the use, and comparison, of OBIA compared to the pixel-based classification used in this research.
- 6) First identify biomass from the images, using different combinations of images; compare the accuracy from different frequencies / polarisations and texture measures and use this to classify the underlying land cover classes.
- 7) What are the overall and class-specific classification accuracies when the roles of the main site and additional sites are reversed, to that used in this study? I.e. train the model on the DRC wet season data and apply the model to the Cameroon wet season data; and train the model on the Chad dry season data and apply it to the Tanzania dry season data.
- 8) What are the overall and class-specific classification accuracies when the roles of the inter-season comparison sites are reversed? I.e. train the model on the Cameroon dry season data and apply it to the Cameroon wet season data; and train the model on the Tanzania wet season data and apply it to the Tanzania dry season data.
- 9) Deriving a cross-ecoregion modelling approach: either training the model on one ecoregion and applying it to another ecoregion, or training the model on both

ecoregions simultaneously, similar to the site-specific land cover class approach used in this study.

- 10) Testing additional separability indices such as the Fisher criterion and the Bhattacharyya distance, to assess if a better measure than the KLD class difference measures can be found. This is especially for the multi-modal statistical distributions of the texture measures.

### 8.3.3 Future view

The future view is that there will be an increase in multi-frequency SAR applications, from all the available SAR satellites in the future (see Table 2.2 for a list of current SAR satellites and those planned for the future). With ALOS-2 PALSAR-2 in orbit, along with Sentinel 1A, an LC-band application is feasible. An LX-band combination is also possible between ALOS-2 PALSAR-2 and TerraSAR-X. If the planned TerraSAR-L satellites coincide with current TerraSAR-X satellites, this could create the ideal LX-band constellation for land cover classification.

The cost of SAR data has decreased substantially during the last few years, with open access to Sentinel 1 data, and from 24 November 2014, ALOS PALSAR data available at a fraction of the cost per scene than before then (RESTEC, 2014). It is the author's view that the lower cost and open access to SAR data will increase its use by a much wider user community. The lowering in cost of data processing and data storage will make the processing and storage of the large data requirements for SAR satellite images more feasible to a much wider user community of SAR data.

A universal model to classify SAR data for multiple ecoregions is still lacking. The training of a model using site-specific land cover classes, as in this study, shows one way of doing this. If enough samples are collected from multiple sites, from multiple ecoregions, around the world, this could be done. This could be one of the next challenges to utilise the increasing number of SAR images available, with big data and distributed processing approaches, in the future.

## Appendices

### Appendix One: GAMMA orthorectification scripts

Appendix One covers the orthorectification scripts using the GAMMA SAR and interferometric software (Werner et al., 2012). The script was originally written by Waldram (2014), and was modified for this research. To run the GAMMA orthorectification scripts, this research used the ALICE High Performance Computing Facility at the University of Leicester.

#### A1.1 Overview of GAMMA orthorectification scripts

The broad outline of the orthorectification steps are shown in Figure 4.2. The script requires the following input:

1. An SLC SAR image: TerraSAR, ENVISAT ASAR or ALOS PALSAR.
2. A Landsat Tile ( or mosaic of tiles) covering the full extent of the SAR image, or
3. An SRTM DEM Tile (or mosaic of tiles) covering the full extent of the SAR image

The main script used for this processing, is the automate.csh script which is shown in Script A1.1. The directory structure that needs to be created before the script is run, is shown in Figure A1.1, and the /DEM directory structure with file locations in Figure A1.2. The location of Scripts A1.1, A1.2, A1.4 and A1.5 should be placed in the /phd folder, as shown in Figure A1.1. The base directory in this case is /scratch/gionet/bfs4/ but can be any other reference directory in which the analysis is done. The script provides the ability to use either an SRTM DEM or an alternative DEM from a panchromatic Landsat image. Even though panchromatic Landsat images were used in this research, the alternative SRTM scripts are presented as reference. The steps to download, mosaic and prepare a panchromatic Landsat image are presented in Section A1.2. The steps to download, mosaic and prepare an SRTM DEM for the GAMMA scripts are presented in Section A1.3. Some of the file and folder names are in lower case, e.g. /srtm. It is important to note that the linux operating system is case sensitive, and the folder and file names should therefore be treated as lower or upper case as appropriate.

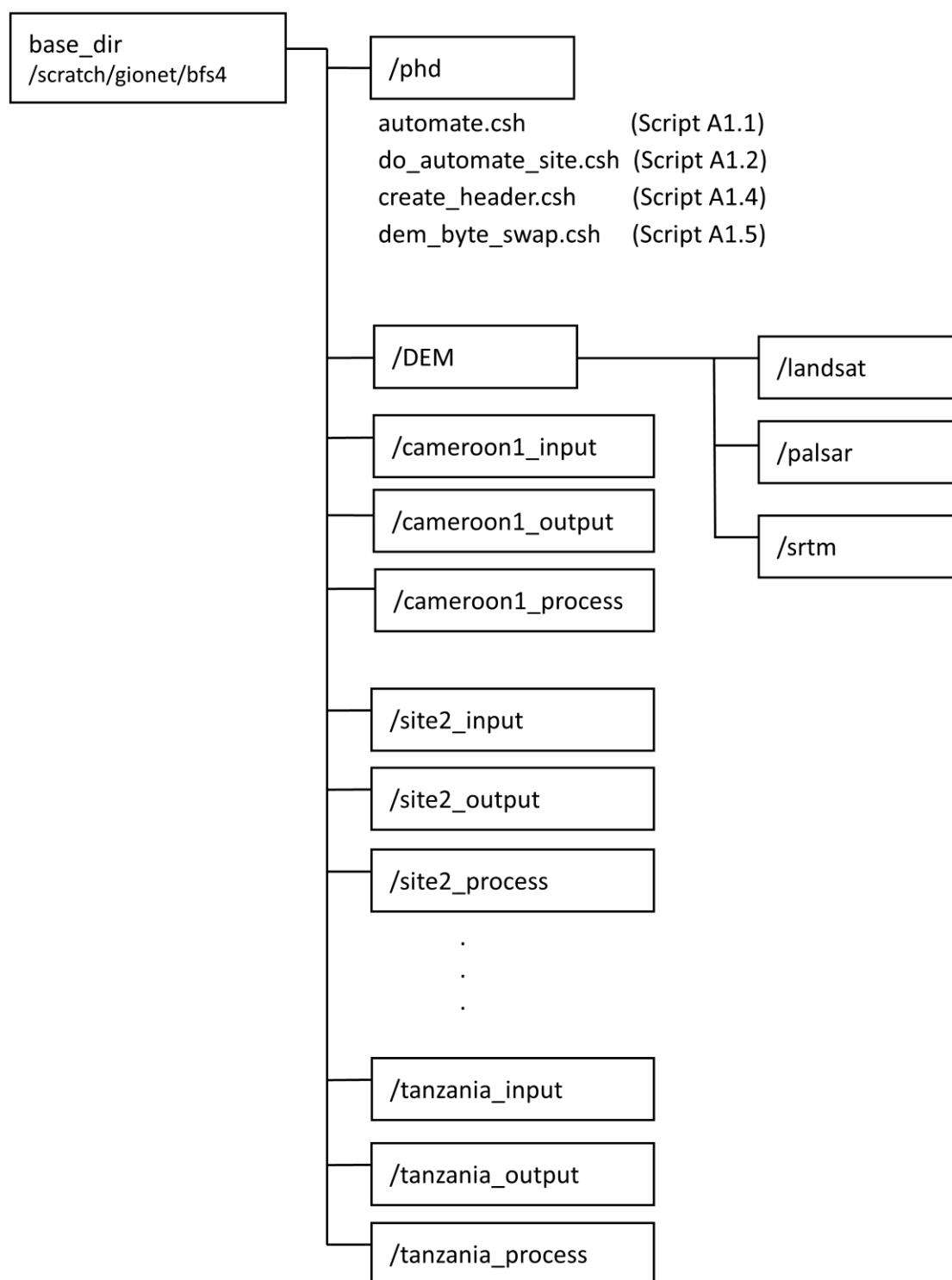


Figure A1.1 Directory structure to be created for the GAMMA orthorectification scripts.

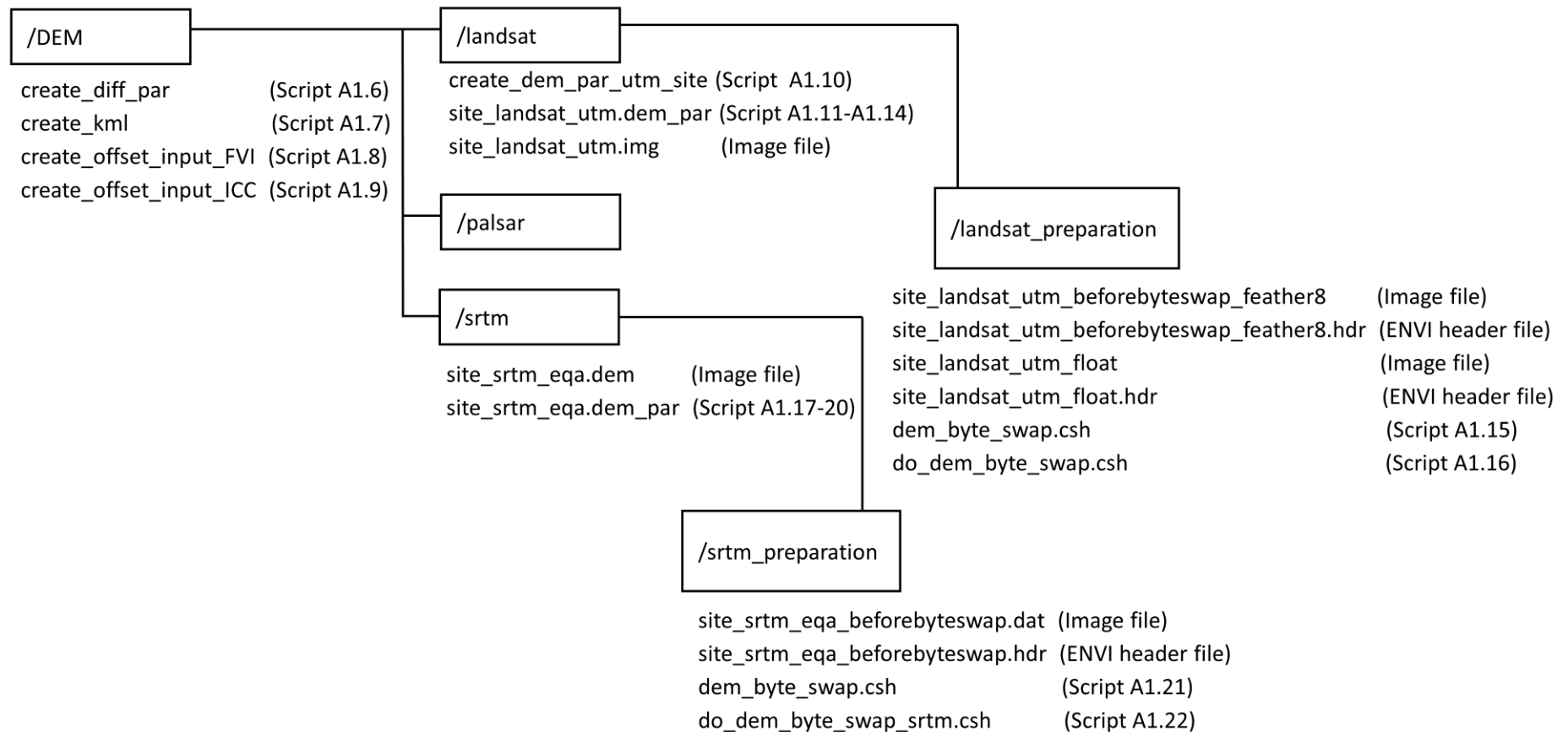


Figure A1.2 The directory structure of the /DEM folder, with locations of scripts



## A1.2 DEM preparation steps – Landsat

The steps to download, mosaic and prepare a Landsat panchromatic image to be used as a DEM in the GAMMA orthorectification scripts are given here:

- 1) Download landsat images from <http://earthexplorer.usgs.gov/>
- 2) Mosaic in ENVI (using the feather 8 method):
  - a. Load the images in ENVI. Select the tool to 'georeferenced mosaicking' from the tool menu.
  - b. Select Import > 'Import Files and Edit Properties'.
  - c. Select Feathering Distance of 8 pixels, No Colour Balancing.
  - d. Select File > Apply to create the mosaic.
- 3) Convert to float using float(b1) in bandmath in ENVI.
- 4) Copy to /DEM/landsat/landsat\_preparation on ALICE.
- 5) Run the do\_dem\_byteswap.csh script (Script A1.16) located in the /DEM/landsat/landsat\_preparation folder.

## A1.3 DEM preparation – SRTM

The steps to download, mosaic and prepare an SRTM DEM to be used in the GAMMA orthorectification scripts are given here:

- 1) Download from <http://srtm.csi.cgiar.org/> (void filled, version 4 SRTM, recommended by GAMMA) or from <http://dwtkns.com/srtm/> which is an intuitive link to the <http://srtm.csi.cgiar.org/> website.
  - a. Open file in ENVI and save as ENVI format (to get header file info, to be used later).
  - b. To work with the SRTM's on your local computer, the images need to be re-projected into UTM projection, to be able to overlap the SAR imagery, which will be geo-terrain corrected into UTM.
- 2) Mosaic in ENVI (using the feather 8 method), and do not convert to float format as with the Landsat image.

- a. Load the images in ENVI. Select the tool to 'georeferenced mosaicking' from the tool menu.
  - b. Select Import > 'Import Files and Edit Properties'
  - c. Select Feathering Distance of 8 pixels, No Colour Balancing
  - d. Select File > Apply to create the mosaic.
- 3) Copy the mosaic or single srtm.dat and srtm.hdr files to ALICE into the /DEM/srtm/srtm\_preparation folder
- 4) Run the do\_dem\_byte\_swap\_srtm.csh (Script A1.22) program
- 5) Manually create site\_srtm\_eqa.dem\_par (Scripts A1.17- A1.20) files using the details from the .hdr file from the ENVI mosaic (this could be automated later if necessary).
  - a. In the header file, the samples becomes the width in the dem\_par file, and the lines becomes the nlines in the dem\_par file.
  - b. Check the latitude and longitude for the top left corner. The .dem\_par files are represented in Scripts A1.17-A1.20.

## A1.4 GAMMA orthorectification scripts

The GAMMA orthorectification scripts and supporting files as modified from Waldram (2014), are represented in this section. The main GAMMA script, called automate.csh are presented in Script A1.1.

### A1.4.1 /phd folder scripts

*Script A1.1 GAMMA script, automate.csh*

```
1  #!/bin/csh -fe
2  if ($#argv < 13) then
3    echo " "
4    echo "**** Process ALOS PALSAR, ENVISAT ASAR and TerraSAR-X Data ****"
5    echo "**** Copyright 2004, Gamma Remote Sensing, v1.4 18-May-2004 ts/uw/clw"
6    echo "****"
7    echo "**** Modified from the Emma Tebbs/MSW version ****"
8    echo "**** Edited by Bernard Spies, bfspies@gmail.com - version 6 June 2013 ****"
9    echo " "
10   echo "automate.csh : To calibrate, multilook and geo-terrain correct images from
    ALOS PALSAR, ENVISAT ASAR and TerraSAR-X SLC images"
11   echo " "
12   echo "usage: automate.csh"
13   echo "  1. working_dir    Base working directory e.g. /alice/scratch/gionet/bfs4"
14   echo "  2. script_dir     Directory in which the script is kept, e.g. phd"
15   echo "  3. region         Region name e.g. cameroon1, cameroon2, drc, tanzania,
16   sudan, chad"
17   echo "  4. sensor         PALSAR, ASAR, or TSX"
18   echo "  5. slc1_dir1      Data directory for slc1, e.g.
19   Cameroon1_Dry_PALSAR_20100318_SLC_0_HH"
20   echo "  6. slc1_dir2      Deeper directory for TSX, if not TSX use -"
21   echo "  7. slc1           SLC 1 level 1.1 IMG file name e.g. IMG-HH-
22   ALPSRP075937190-H1.1__A"
23   echo "  8. slc2_dir1      Data directory for slc2 for interferometry, e.g.
24   Cameroon1_Dry_PALSAR_20100318_SLC_0_HH"
25   echo "  9. slc2_dir2      Deeper directory for TSX, if not TSX use -"
26   echo "  10. slc2          SLC 2 level 1.1 IMG file name e.g. IMG-HH-
27   ALPSRP075937190-H1.1__A"
28   echo "  11. mli_range_looks Multi-look intensity range looks (for PALSAR from
29   Matt: FBS = 2; FBD, PLR = 1) "
30   echo "  12. mli_azimuth_looks Multi-look intensity azimuth looks (for PALSAR
31   from Matt: FBS, FBD = 6; PLR = 9)"
32   echo "  13. polarisation   Polarisation (HH, HV, VH or VV)"
```

```

32 echo " 14. grd_pixel_spacing Ground pixel spacing, output resolution in m for
    utm (e.g. 25, assume pixel is square)"
33 echo " 15. create_offset_method Offset_algorithm (for interferometry) 1:
34 intensity cross-correlation (default), 2: fringe visibility"
35 echo " 16. direction Ascending/Descending, Asc,flipping = 1,
36 Desc,flipping = -1"
37 echo " 17. interp_mode Interpolation Mode (0: nearest neighbor (default), 1:
38 spline, 2: spline log)"
39 echo " 18. dem_source DEM source, e.g. srtm"
40 echo " 19. img_source image source, e.g. landsat or palsar"
41 echo " 20. init_thres Initial Threshold for SNR correlation between SAR
42 images and simulated SAR from DEM (default 7.000)"
43 echo " 21. second_thres Secondary Threshold for SNR correlation between SAR
44 images and simulated SAR from DEM (default 7.0)"
45 echo " 22. hemisphere Hemisphere of the image, for the ENVI .hdr file
46 creation"
47 echo " 23. imgtype Image type: SLC or MLI"
48 echo " 24. geocode_path GTC (directly from Gamma) vs geocode (version by
49 Matt Waldram)"
50 echo " "
51 exit
52 endif
53
54 #set the variables from the do_automate_site.csh command
55 set working_dir = $1
56 set script_dir = $2
57 set region = $3
58 set sensor = $4
59 set slc1_dir1 = $5
60 set slc1_dir2 = $6
61 set slc1 = $7
62 set slc2_dir1 = $8
63 set slc2_dir2 = $9
64 set slc2 = $10
65 set range_looks = $11
66 set azimuth_looks = $12
67 set pol = $13
68 set grd_pixel_spacing = $14
69 set create_offset_method = $15
70 set direction = $16
71 set interp_mode = $17
72 set dem_source = $18
73 set img_source = $19
74 set output_name = $5_$13
75 set init_thres = $20
76 set second_thres = $21
77 set hemisphere = $22

```

```

78 set imgtype          = $23
79 set geocode_path     = $24
80
81 echo "input parameters:"
82 echo "working_dir(1): $working_dir"
83 echo "script_dir(2): $script_dir"
84 echo "region(3):      $region"
85 echo "sensor(4):      $sensor"
86 echo "slc1_dir1(5):   $slc1_dir1"
87 echo "slc1_dir2(6):   $slc1_dir2"
88 echo "slc1(7):        $slc1"
89 echo "slc2_dir1(8):   $slc2_dir1"
90 echo "slc2_dir2(9):   $slc2_dir2"
91 echo "slc2(10):       $slc2"
92 echo "range_looks(11): $range_looks"
93 echo "azimuth_looks(12): $azimuth_looks"
94 echo "pol(13):        $pol"
95 echo "grd_pixel_spacing(14): $grd_pixel_spacing"
96 echo "create_offset_method(15): $create_offset_method"
97 echo "direction(16)   $direction"
98 echo "interp_mode(17): $interp_mode"
99 echo "dem_source(18):  $dem_source"
100 echo "img_source(19):  $img_source"
101 echo "output_name(5_13): $output_name"
102 echo "init_thres(20):  $init_thres"
103 echo "second_thres(21): $second_thres"
104 echo "hemisphere(22):  $hemisphere"
105 echo "imgtype (23):    $imgtype"
106 echo "geocode_path(24): $geocode_path"
107
108 if ($direction == "Ascending") set flipping = 1
109 if ($direction == "Descending") set flipping = -1
110
111 set grd_rsp = $grd_pixel_spacing
112 set grd_azsp = $grd_pixel_spacing
113 set input_dir = $working_dir/$region\_input
114
115 #these offsetts are for interferometric processing
116 set init_offset_corr_threshold_1 = 4.0
117 set init_offset_corr_threshold_2 = 4.0
118
119 #Option1 1: slc1 only, Preparation - orthorectification only, no interforogram, using
GTC_MLI procedure
120 if ($geocode_path == "GTC") then
121   echo "GTC path"
122   set do_untidy          = 0
123   set do_prepare_slc1    = 1 #1 necessary for slc1

```

```

124 set do_initial_offset1 = 0
125 set do_create_offset1 = 0
126 set do_interferogram1 = 1 #1 necessary for slc1 - multilooking
127 set do_base_est1 = 0
128 set do_flattening1 = 0
129 set do_coherence1 = 0
130 set do_ad_filtering1 = 0
131 set do_calibration1 = 1 #1 necessary for slc1
132 set do_GTC_MLI = 1 #1 GTC_MLI code from GAMMA
133 set do_geocode1 = 0 #0 only do_geocode1 for slc1(not slc1 & slc2) works at
the moment
134 set add_pixel_area = 0 #0 suggestion by Andreas Wiesmann, Oct2013
135 set do_geocode1_next = 0 #0 only do_geocode1_next for slc1 works at the
moment
136 set onemore_init_offsetm = 0 #0 extra option suggested by Andreas Wiesmann
137 set do_header1 = 0 #0 necessary for slc1
138 set do_kml = 0
139 set do_tidy_up = 0 #0 necessary for slc1, to tidy up
140 #keeper variables
141 set do_prepare_slc2 = 0
142 set do_initial_offset2 = 0
143 set do_create_offset2 = 0
144 set do_interferogram2 = 0
145 set do_base_est2 = 0
146 set do_flattening2 = 0
147 set do_coherence2 = 0
148 set do_ad_filtering2 = 0
149 set do_calibration2 = 0
150 set do_header2 = 0
151 set do_tidy_up_Matt = 0
152 endif #if geocode_pat == "GTC"
153
154 #Option 2: slc1 only, Preparation - orthorectification only, no interferogram, using
Matt's geocode procedure
155 if ($geocode_path == "geocode") then
156 echo "geocode path original"
157 set do_untidy = 0
158 set do_prepare_slc1 = 1 #1 necessary for slc1
159 set do_initial_offset1 = 0
160 set do_create_offset1 = 0
161 set do_interferogram1 = 1 #1 necessary for slc1 - multilooking
162 set do_base_est1 = 0
163 set do_flattening1 = 0
164 set do_coherence1 = 0
165 set do_ad_filtering1 = 0
166 set do_calibration1 = 1 #1 necessary for slc1
167 set do_GTC_MLI = 0 #0 GTC_MLI code from GAMMA

```

```

168 set do_geocode1 = 1 #1 only do_geocode1 for slc1(not slc1 & slc2) works at
the moment
169 set add_pixel_area = 0 #0 suggestion by Andreas Wiesmann, Oct2013
170 set do_geocode1_next = 1 #1 only do_geocode1_next for slc1 works at the
moment
171 set onemore_init_offsetm = 0 #0 extra option suggested by Andreas Wiesmann
172 set do_header1 = 1 #1 necessary for slc1
173 set do_kml = 0
174 set do_tidy_up = 1 #1 necessary for slc1, to tidy up
175 #keeper variables
set do_prepare_slc2 = 0
176 set do_initial_offset2 = 0
177 set do_create_offset2 = 0
178 set do_interferogram2 = 0
179 set do_base_est2 = 0
180 set do_flattening2 = 0
181 set do_coherence2 = 0
182 set do_ad_filtering2 = 0
183 set do_calibration2 = 0
184 set do_header2 = 0
185 set do_tidy_up_Matt = 0
186 endif #else if geocode_path == "geocode"
187
188 #Option 3: slc1 only, Preparation - orthorectification only, no interferogram, using
Matt's geocode procedure - with pixel area correction
189 if ($geocode_path == "geocode_with_pixel_area") then
190 echo "geocode path with pixel area"
191 set do_untidy = 0
192 set do_prepare_slc1 = 1 #1 necessary for slc1
193 set do_initial_offset1 = 0
194 set do_create_offset1 = 0
195 set do_interferogram1 = 1 #1 necessary for slc1 - multilooking
196 set do_base_est1 = 0
197 set do_flattening1 = 0
198 set do_coherence1 = 0
199 set do_ad_filtering1 = 0
200 set do_calibration1 = 1 #1 necessary for slc1
201 set do_GTC_MLI = 0 #0 GTC_MLI code from GAMMA
202 set do_geocode1 = 1 #1 only do_geocode1 for slc1(not slc1 & slc2) works at
the moment
203 set add_pixel_area = 1 #1 suggestion by Andreas Wiesmann, Oct2013
204 set do_geocode1_next = 1 #1 only do_geocode1_next for slc1 works at the
moment
205 set onemore_init_offsetm = 1 #1 extra option suggested by Andreas Wiesmann
206 set do_header1 = 1 #1 necessary for slc1
207 set do_kml = 0
208 set do_tidy_up = 1 #1 necessary for slc1, to tidy up

```

```

209 #keeper variables
210 set do_prepare_slc2 = 0
211 set do_initial_offset2 = 0
212 set do_create_offset2 = 0
213 set do_interferogram2 = 0
214 set do_base_est2 = 0
215 set do_flattening2 = 0
216 set do_coherence2 = 0
217 set do_ad_filtering2 = 0
218 set do_calibration2 = 0
219 set do_header2 = 0
220 set do_tidy_up_Matt = 0
221 endif #else if geocode_path == "geocode"
222
223 #if ((! $geocode_path == "geocode") || (! $geocode_path == "GTC")) then
224 # echo "incorrect geocode_path value. Use GTC or geocode as input"
225 #endif #geocode_info
226
227 if (0) then
228 set do_geo_fine_LS = 0
229 set do_geocode_LS = 0
230 endif
231
232 #variables for the unwrapping steps (for interferometry)
233 set do_unwrapping_bcut2 = 0
234 set do_unwrapping_mcf2 = 0
235
236 #=====#
237 # set the hemisphere for the different locations of the images
238 #set hemisphere = 0
239 #if ($region == "cameroon1") set hemisphere = "North"
240 #if ($region == "cameroon2") set hemisphere = "North"
241 #if ($region == "drc") set hemisphere = "North"
242 #if ($region == "tanzania") set hemisphere = "South"
243 #if ($region == "sudan") set hemisphere = "North"
244 #if ($region == "chad") set hemisphere = "North"
245
246 #set process directory, clean for reruns, change to the process directory
247 set process_dir =
248 $region\_process/$output_name\_ $range_looks\ x$azimuth_looks\_ $grd_pixel_spacing\_m\_ $geocode_path
249 if (! (-d $working_dir/$process_dir) ) mkdir $working_dir/$process_dir
250
251 echo "cd to process directory, to clean before new processing"
252 cd $working_dir/$process_dir
253 if (-e create_header.csh) rm *
254 echo "after cleaning the directory"

```



```

255
256 #copy create_header script and GTC_MLI script from the script folder to the process
257 folder
258 cp $script_dir/create_header.csh $working_dir/$process_dir
259 cp $script_dir/GTC_MLI_edit.csh $working_dir/$process_dir
260 cd $working_dir/$process_dir
261
262 #=====#
263 #do untidy (remove files from processed_files folder created in do_tidy and places
264 them in insar_directory)#
265 if ($do_untidy) then
266 # if (!(-d processing_files)) then
267 cd $working_dir/$process_dir/processing_files/
268 mv * ..
269 cd ..
270 # endif
271 endif
272
273 #=====#
274 #Prepare slc data and processing parameter files#
275 # for 1 slc image, for geocoding only, not interferogram#
276 if ($do_prepare_slc1) then
277 echo ""
278 echo "input directory to prepare the slc:"
279 echo $input_dir
280 echo ""
281 echo "slc1.slc"
282 echo $slc1.slc
283 echo ""
284 echo "slc1.slc.par"
285 echo $slc1.slc.par
286
287 if (-e $slc1.slc) rm -f $slc1.slc
288 if (-e $slc1.slc.par) rm -f $slc1.slc.par
289 if (($sensor == "PALSAR") && ($imgtype == "SLC")) par_EORC_PALSAR
290 $input_dir/$slc1_dir1/LED* $slc1.slc.par $input_dir/$slc1_dir1/$slc1 $slc1.slc
291
292 if (($sensor == "PALSAR") && ($imgtype == "MLI")) par_EORC_PALSAR_geo
293 $input_dir/$slc1_dir1/LED* $slc1.mli.par $slc1.dem_par
294 $input_dir/$slc1_dir1/$slc1 $slc1.mli
295
296 if ($sensor == "ASAR" && $imgtype == "SLC") par_ASAR
297 $input_dir/$slc1_dir1/$slc1 $slc1.slc -
298 if ($sensor == "ASAR" && $imgtype == "SLC") mv *.$pol.SLC $slc1.slc
299 if ($sensor == "ASAR" && $imgtype == "SLC") mv *.$pol.SLC.par $slc1.slc.par
300 if ($sensor == "ASAR" && $imgtype == "MLI") par_ASAR
301 $input_dir/$slc1_dir1/$slc1 $slc1.mli -

```

```

295 if ($sensor == "ASAR" && $imgtype == "MLI") mv *.$pol.MLI $slc1.mli
296 if ($sensor == "ASAR" && $imgtype == "MLI") mv *.$pol.MLI.par $slc1.mli.par
297 if ($sensor == "RS2" && $imgtype == "MLI") par_RSAT2_SG
    $input_dir/$slc1_dir1/product.xml $input_dir/$slc1_dir1/lutSigma.xml
    $input_dir/$slc1_dir1/imagery\_ $pol\ $tif $pol $slc1.grd_par $slc1.grd
298
299 #if ($sensor == "ASAR") mv $slc1.slc.$pol.SLC $slc1.slc
300 #if ($sensor == "ASAR") mv *.$pol.SLC.par $slc1.slc.par
301 #mv *.HH.SLC Chad_Dry_ASAR_20071106_VV.N1.slc
302 #mv *.HH.SLC.par Chad_Dry_ASAR_20071106_VV.N1.slc.par
303
304 echo ""
305 echo "test Sudan TSX SLC"
306 echo "dir1: $input_dir/$slc1_dir1/$slc1_dir2/*.xml "
307 echo "dir2: $input_dir/$slc1_dir1/$slc1_dir2/IMAGEDATA/$slc1 "
308 echo "slc1: $slc1"
309 echo "pol: $pol"
310 echo ""
311
312 if ($sensor == "TSX" && $imgtype == "SLC") par_TX_SLC
    $input_dir/$slc1_dir1/$slc1_dir2/*.xml
    $input_dir/$slc1_dir1/$slc1_dir2/IMAGEDATA/$slc1 $slc1.slc.par $slc1.slc $pol
313
314 if ($sensor == "TSX" && $imgtype == "MLI") par_TX_SLC
    $input_dir/$slc1_dir1/$slc1_dir2/*.xml
    $input_dir/$slc1_dir1/$slc1_dir2/IMAGEDATA/$slc1 $slc1.mli.par $slc1.dem_par
    $slc1.cml $pol
315
316 endif #do_prepare_slc1
317
318 #=====#
319 #Prepare slc data and processing parameter files#
320 # for 2 slc images, for geocoding AND interferogram#
321 if ($do_prepare_slc2) then
322     echo ""
323     echo "input directory to prepare the slc:"
324     echo $input_dir
325     #echo $working_dir/$region\_input/$slc1_dir1/LED*
326     #echo $working_dir/$region\_input/$slc1_dir1/$slc1
327     #echo $working_dir/$region\_input/$slc2_dir1/LED*
328     #echo $working_dir/$region\_input/$slc2_dir1/$slc2
329     echo ""
330     if (-e $slc1.slc) rm -f $slc1.slc
331     if (-e $slc1.slc.par) rm -f $slc1.slc.par
332     if ($sensor == "PALSAR") par_EORC_PALSAR $input_dir/$slc1_dir1/LED*
        $slc1.slc.par $input_dir/$slc1_dir1/$slc1 $slc1.slc
333     if (-e $slc2.slc) rm -f $slc2.slc

```

```

334 if (-e $slc2.slc.par) rm -f $slc2.slc.par
335 if ($sensor == "PALSAR") par_EORC_PALSAR $input_dir/$slc2_dir1/LED*
$slc2.slc.par $input_dir/$slc2_dir1/$slc2 $slc2.slc
336 endif
337
#####
338 ### Offset estimation of the slc images (for interferometry)
339 ###
340 #####
341 if ($do_initial_offset1) then
342   echo ""
343   echo "do_intial_offset1:"
344   echo "no initial offset necessary, since no interferogram is calculated using this
option"
345   echo ""
346   endif
347
348 if ($do_initial_offset2) then
349   if (-e $slc1\_ $slc2.off) rm -f $slc1\_ $slc2.off
350   if ($create_offset_method == 1) then
351     create_offset $slc1.slc.par $slc2.slc.par $slc1\_ $slc2.off $create_offset_method
$range_looks $azimuth_looks < $working_dir/DEM/create_offset_input_ICC
352   else
353     endif
354   if ($create_offset_method == 2) then
355     create_offset $slc1.slc.par $slc2.slc.par $slc1\_ $slc2.off $create_offset_method
$range_looks $azimuth_looks < $working_dir/DEM/create_offset_input_FVI
356   else
357     endif
358   init_offset $slc1.slc $slc2.slc $slc1.slc.par $slc2.slc.par $slc1\_ $slc2.off
$range_looks $azimuth_looks - - 0 0 $init_offset_corr_threshold_1
359   init_offset $slc1.slc $slc2.slc $slc1.slc.par $slc2.slc.par $slc1\_ $slc2.off 1 1 - - 0 0
$init_offset_corr_threshold_2
360   endif
361
362 if ($do_create_offset1) then
363   echo ""
364   echo "do_create_offset1:"
365   echo "create offset is not necessary, since no interferogram is calculated using this
option"
366   echo ""
367   endif
368
369 if ($do_create_offset2) then
370   if ($create_offset_method == 1) then
371     if (-e $slc1\_ $slc2.off) rm -f $slc1\_ $slc2.off
372     if (-e $slc1\_ $slc2.snr) rm -f $slc1\_ $slc2.snr

```

```

373 if (-e $slc1\_slc2.offsets) rm -f $slc1\_slc2.offsets
374 if (-e $slc1\_slc2.coffs) rm -f $slc1\_slc2.coffs
375 if (-e $slc1\_slc2.coffsets) rm -f $slc1\_slc2.coffsets
376 offset_pwr $slc1.slc $slc2.slc $slc1.slc.par $slc2.slc.par $slc1\_slc2.off
$slc1\_slc2.off $slc1\_slc2.snr - - $slc1\_slc2.offsets 1 - - $second_thres
377 offset_fit $slc1\_slc2.off $slc1\_slc2.snr $slc1\_slc2.off $slc1\_slc2.coff
$slc1\_slc2.coffsets - 4 0
378 else
379 endif
380
381 if ($create_offset_method == 2) then
382 if (-e $slc1\_slc2.off) rm -f $slc1\_slc2.off
383 if (-e $slc1\_slc2.snr) rm -f $slc1\_slc2.snr
384 if (-e $slc1\_slc2.offsets) rm -f $slc1\_slc2.offsets
385 if (-e $slc1\_slc2.coffs) rm -f $slc1\_slc2.coffs
386 if (-e $slc1\_slc2.coffsets) rm -f $slc1\_slc2.coffsets
387 offset_SLC $slc1.slc $slc2.slc $slc1.slc.par $slc2.slc.par $slc1\_slc2.off
$slc1\_slc2.off $slc1\_slc2.snr - - $slc1\_slc2.offsets 2 - - 3.0
388 offset_fit $slc1\_slc2.off $slc1\_slc2.snr $slc1\_slc2.off $slc1\_slc2.coff
$slc1\_slc2.coffsets - 4 0
389 else
390 endif
391 endif
392
393 #####
394 ### Compute interferogram (interferometry and multi-looking calculation)
395 ###
396 #####
397 #this step used to be $do_interferogram1, but has been renamed to do_multilook1
as a better description
398 if ($do_interferogram1) then
399 #basically only does multilooking of the slc1 image, no interforegram
400 if (-e $slc1.mli) rm -f $slc1.mli
401 if (-e $slc1.mli.par) rm -f $slc1.mli.par
402 multi_look $slc1.slc $slc1.slc.par $slc1.mli $slc1.mli.par $range_looks
$azimuth_looks
403 set image_width = `awk '{if($1 == "range_samples:" ) printf("%d", $2)}' <
$slc1.mli.par`
404 echo ""
405 echo "image_width:"
406 echo $image_width
407 echo ""
408 raspwr $slc1.mli $image_width 1 0 1 1 1.0 0.35 $flipping $slc1.mli.bmp
409 if (-e $slc1.mli_flip.bmp) rm -f $slc1.mli_flip.bmp
410 #convert -flip $slc1.mli.bmp $slc1.mli_flip.bmp
411 rm -f $slc1.mli.bmp
412 endif

```

```

413
414 #multi-looking and interferogram calculation
415 if ($do_interferogram2) then
416   if (-e $slc1\_$_slc2.int) rm -f $slc1\_$_slc2.int
417   if (-e $slc2.rslc) rm -f $slc2.rslc
418   if (-e $slc2.rslc.par) rm -f $slc2.rslc.par
419   SLC_interp $slc2.slc $slc1.slc.par $slc2.slc.par $slc1\_$_slc2.off $slc2.rslc
    $slc2.rslc.par
420
421   SLC_intf $slc1.slc $slc2.rslc $slc1.slc.par $slc2.rslc.par $slc1\_$_slc2.off
    $slc1\_$_slc2.int $range_looks $azimuth_looks
422   if (-e $slc1.mli) rm -f $slc1.mli
423   if (-e $slc1.mli.par) rm -f $slc1.mli.par
424   if (-e $slc2.mli) rm -f $slc2.mli
425   if (-e $slc2.mli.par) rm -f $slc2.mli.par
426   if (-e $slc2.rmli) rm -f $slc2.rmli
427   if (-e $slc2.rmli.par) rm -f $slc2.rmli.par
428   multi_look $slc1.slc $slc1.slc.par $slc1.mli $slc1.mli.par $range_looks
    $azimuth_looks
429   multi_look $slc2.slc $slc2.slc.par $slc2.mli $slc2.mli.par $range_looks
    $azimuth_looks
430   multi_look $slc2.rslc $slc2.rslc.par $slc2.rmli $slc2.rmli.par $range_looks
    $azimuth_looks
431   set image_width = `awk '{if($1 == "range_samples:") printf("%d", $2)}' <
    $slc1.mli.par`
432   echo ""
433   echo "image_width:"
434   echo $image_width
435   echo ""
436   raspwr $slc1.mli $image_width 1 0 1 1 1.0 0.35 $flipping $slc1.mli.bmp
437   raspwr $slc2.mli $image_width 1 0 1 1 1.0 0.35 $flipping $slc2.mli.bmp
438   raspwr $slc2.rmli $image_width 1 0 1 1 1.0 0.35 $flipping $slc2.rmli.bmp
439   if (-e $slc1.mli_flip.bmp) rm -f $slc1.mli_flip.bmp
440   if (-e $slc2.mli_flip.bmp) rm -f $slc2.mli_flip.bmp
441   if (-e $slc2.rmli_flip.bmp) rm -f $slc2.rmli_flip.bmp
442   #convert -flip $slc1.mli.bmp $slc1.mli_flip.bmp
443   #convert -flip $slc2.mli.bmp $slc2.mli_flip.bmp
444   #convert -flip $slc2.rmli.bmp $slc2.rmli_flip.bmp
445   rm -f $slc1.mli.bmp $slc2.mli.bmp $slc2.rmli.bmp
446   rasmph $slc1\_$_slc2.int $image_width 1 0 1 1 1.0 0.35 $flipping
    $slc1\_$_slc2.mag_phase.bmp
447   if (-e $slc1\_$_slc2.mag_phase_flip.bmp) rm -f $slc1\_$_slc2.mag_phase_flip.bmp
448   #convert -flip $slc1\_$_slc2.mag_phase.bmp $slc1\_$_slc2.mag_phase_flip.bmp
449   rasmph_pwr $slc1\_$_slc2.int $slc1.mli $image_width 1 1 0 1 1 1.0 0.35 $flipping
    $slc1\_$_slc2.mag_phase_pwr.bmp
450   if (-e $slc1\_$_slc2.mag_phase_pwr_flip.bmp) rm -f
    $slc1\_$_slc2.mag_phase_pwr_flip.bmp

```

```

451 #convert -flip $slc1\_$_slc2.mag_phase_pwr.bmp
    $slc1\_$_slc2.mag_phase_pwr_flip.bmp
452 #rm -f $slc1\_$_slc2.mag_phase_pwr.bmp $slc1\_$_slc2.mag_phase.bmp
453
454 set image_width = `awk 'if($1 == "range_samples:" ) printf("%d", $2)}' <
    $slc1.mli.par`
455 endif
456
457 #command for all methods to access - set image_width
458 #uncomment to run all
459 #set image_width = `awk 'if($1 == "range_samples:" ) printf("%d", $2)}' <
    $slc1.mli.par`
460
461 #####
462 ### Generate baseline file (parallel component from the orbit parameters,
463 ###
464 ### perpendicular component from the fringe rate), remove the Earth phase
    trend, ###
465 ### filter (running adf several times with a smaller coefficient and      ###
466 ### decreasing window size (e.g. 128,64,32) will lead to a better filtering but   ###
467 ### is time consuming) and estimate the phase noise (obs.: 05721_25394.smcc is
468 ###
469 ### not the degree of coherence but rather the phase noise because it is      ###
470 ### estimated from the filtered interferogram without use of the intensities).
471 ###
472 #####
473 if ($do_base_est1) then # generate baseline file (first remove an eventual first
    estimate)
474     echo ""
475     echo "do_base_est1:"
476     echo "baseline generation is not necessary, since no interferogram is calculated
    with 1 slc image"
477     echo ""
478     endif
479
480 if ($do_base_est2) then # generate baseline file (first remove an eventual first
    estimate)
481     if (-e $slc1\_$_slc2.base) rm -f $slc1\_$_slc2.base
482     if (-e $slc1\_$_slc2.base.txt) rm -f $slc1\_$_slc2.base.txt
483     touch $slc1\_$_slc2.base.txt
484     base_init $slc1.slc.par $slc2.rslc.par $slc1\_$_slc2.off $slc1\_$_slc2.int
    $slc1\_$_slc2.base 0 1024 1024 > $slc1\_$_slc2.base.txt
485     endif
486
487 #=====#
488 if ($do_flattening1) then # curved Earth phase trend removal (flattening) and filter
489     echo ""

```

```

490 echo "do_flattening1:"
491 echo "do_flattening is not necessary, since no interferogram is calculated with 1
slc image"
492 echo ""
493 endif
494
495 if ($do_flattening2) then # curved Earth phase trend removal (flattening) and filter
496 if (-e $slc1\_slc2.flt) rm -f $slc1\_slc2.flt
497 ph_slope_base $slc1\_slc2.int $slc1.slc.par $slc1\_slc2.off $slc1\_slc2.base
$slc1\_slc2.flt
498 if (-e $slc1\_slc2.flt_mag_phase_flip.bmp) rm -f
$slc1\_slc2.flt_mag_phase_flip.bmp
499 rasmph $slc1\_slc2.flt $image_width 1 0 1 1 1. 0.35 $flipping
$slc1\_slc2.flt_mag_phase.bmp
500 #convert -flip $slc1\_slc2.flt_mag_phase.bmp
$slc1\_slc2.flt_mag_phase_flip.bmp
501 if (-e $slc1\_slc2.flt_mag_phase_pwr_flip.bmp) rm -f
$slc1\_slc2.flt_mag_phase_pwr_flip.bmp
502 rasmph_pwr $slc1\_slc2.flt $slc1.mli $image_width 1 1 0 1 1 1.0 0.35 $flipping
$slc1\_slc2.flt_mag_phase_pwr.bmp
503 #convert -flip $slc1\_slc2.flt_mag_phase_pwr.bmp
$slc1\_slc2.flt_mag_phase_pwr_flip.bmp
504 #rm -f $slc1\_slc2.flt_mag_phase.bmp $slc1\_slc2.flt_mag_phase_pwr.bmp
endif
505
506 if ($do_coherence1) then # coherence estimation and generation of Bitmaps
507 echo ""
508 echo "do_coherence1:"
509 echo "do_coherence is not necessary, since no interferogram is calculated with 1
slc image"
510 echo ""
511 endif
512
513 #=====#
514 if ($do_coherence2) then # coherence estimation and generation of Bitmaps
515 if (-e $slc1\_slc2.cc) rm -f $slc1\_slc2.cc
516 cc_wave $slc1\_slc2.flt $slc1.mli $slc2.rmli $slc1\_slc2.cc $image_width
517 if (-e $slc1\_slc2.cc_flip.bmp) rm -f $slc1\_slc2.cc_flip.bmp
518 ras_linear $slc1\_slc2.cc $image_width 1 0 1 1 0.0 1.0 $flipping
$slc1\_slc2.cc.bmp
519 #convert -flip $slc1\_slc2.cc.bmp $slc1\_slc2.cc_flip.bmp
520 if (-e $slc1\_slc2.cc_pwr_flip.bmp) rm -f $slc1\_slc2.cc_pwr_flip.bmp
521 endif
522
523 #=====#
524 if ($do_ad_filtering1) then # adaptive interferogram filtering and generation of
Bitmaps

```

```

525 echo ""
526 echo "do_ad_filtering1:"
527 echo "do_ad_filtering is not necessary, since no interferogram is calculated with 1
slc image"
528 echo ""
529 endif
530
531 #=====#
532 if ($do_ad_filtering2) then # adaptive interferogram filtering and generation of
Bitmaps
533   if (-e $slc1\_slc2.cc_filt) rm -f $slc1\_slc2.cc_filt
534   if (-e $slc1\_slc2.flt_filt) rm -f $slc1\_slc2.flt_filt
535   adf $slc1\_slc2.flt $slc1\_slc2.flt_filt $slc1\_slc2.cc_filt $image_width .5
536   if (-e $slc1\_slc2.flt_filt_mag_phase_pwr_flip.bmp) rm -f
$slc1\_slc2.flt_filt_mag_phase_pwr_flip.bmp
537   rasmph_pwr $slc1\_slc2.flt_filt $slc1.mli $image_width 1 1 0 1 1 1.0 0.35
$flipping $slc1\_slc2.flt_filt_mag_phase_pwr.bmp
538   #convert -flip $slc1\_slc2.cc.bmp $slc1\_slc2.cc_flip.bmp
539   if (-e $slc1\_slc2.cc_pwr_flip.bmp) rm -f $slc1\_slc2.cc_pwr_flip.bmp
540   rascc $slc1\_slc2.cc $slc1.mli $image_width 1 1 0 1 1 0.1 0.9 1.0 0.35 $flipping
$slc1\_slc2.cc_pwr.bmp
541   #convert -flip $slc1\_slc2.cc_pwr.bmp $slc1\_slc2.cc_pwr_flip.bmp
542   rm -f $slc1\_slc2.cc.bmp $slc1\_slc2.cc_pwr.bmp
543 endif
544
545 #####
546 ### Phase unwrapping ###
547 #####
548 if ($do_unwrapping_bcut2) then # phase unwrapping (first remove an eventual
549 first estimate)
550   if (-e $slc1\_slc2.flag) rm -f $slc1\_slc2.flag
551   if (-e $slc1\_slc2.unw) rm -f $slc1\_slc2.unw
552   corr_flag $slc1\_slc2.cc_filt $slc1\_slc2.flag $image_width 0.25
553   neutron $slc1.mli $slc1\_slc2.flag $image_width
554   residue $slc1\_slc2.flt_filt $slc1\_slc2.flag $image_width
555   tree_cc $slc1\_slc2.flag $image_width 64
556   grasses $slc1\_slc2.flt_filt $slc1\_slc2.flag $slc1\_slc2.unw $image_width
557   if (-e $slc1\_slc2.unw.bmp) rm -f $slc1\_slc2.unw.bmp
558   if (-e $slc1\_slc2.unw_flip.bmp) rm -f $slc1\_slc2.unw_flip.bmp
559   rasrmg $slc1\_slc2.unw $slc1.mli $image_width 1 1 0 1 1 .5 1. .35 0.0 $flipping
$slc1\_slc2.unw.bmp
560   #convert -flip $slc1\_slc2.unw.bmp $slc1\_slc2.unw_flip.bmp
561   endif
562
563 if ($do_unwrapping_mcf2) then # phase unwrapping (Minimum Cost Flow)
564   if (-e $slc1\_slc2.cc_mask.bmp) rm -f $slc1\_slc2.cc_mask.bmp
565

```



```

566  rasccl_mask $slc1\_slc2.cc $slc1.mli $image_width 1 1 0 1 1 0.2 - - - -
    $slc1\_slc2.cc_mask.bmp
567  if (-e $slc1\_slc2.unw_mcf0) rm -f $slc1\_slc2.unw_mcf0
    mcf $slc1\_slc2.flt_filt $slc1\_slc2.cc $slc1\_slc2.cc_mask.bmp
568  $slc1\_slc2.unw_mcf0 $image_width 1 0 0 - - 1 1 - - -
    rasrmg $slc1\_slc2.unw_mcf0 $slc1.mli $image_width 1 1 0 1 1 .5 1. .35 0.0
569  $flipping $slc1\_slc2.unw_mcf0.bmp
570  #convert -flip $slc1\_slc2.unw_mcf0.bmp $slc1\_slc2.unw_mcf0_flip.bmp
571  if (-e $slc1\_slc2.unw_mcf0_interp) rm -f $slc1\_slc2.unw_mcf0_interp
572  interp_ad $slc1\_slc2.unw_mcf0 $slc1\_slc2.unw_mcf0_interp $image_width
    rasrmg $slc1\_slc2.unw_mcf0_interp $slc1.mli $image_width 1 1 0 1 1 .5 1. .35
573  0.0 $flipping $slc1\_slc2.unw_mcf0_interp.bmp
    #convert -flip $slc1\_slc2.unw_mcf0_interp.bmp
574  $slc1\_slc2.unw_mcf0_interp_flip.bmp
575  if (-e $slc1\_slc2.unw_mcf) rm -f $slc1\_slc2.unw_mcf
    unw_model $slc1\_slc2.flt_filt $slc1\_slc2.unw_mcf0_interp
576  $slc1\_slc2.unw_mcf $image_width - - -
577  if (-e $slc1\_slc2.unw_mcf.bmp) rm -f $slc1\_slc2.unw_mcf.bmp
    rasrmg $slc1\_slc2.unw_mcf $slc1.mli $image_width 1 1 0 1 1 .5 1. .35 0.0
578  $flipping $slc1\_slc2.unw_mcf.bmp
579  #convert -flip $slc1\_slc2.unw_mcf.bmp $slc1\_slc2.unw_mcf_flip.bmp
580  rm -f $slc1\_slc2.unw_mcf0_interp.bmp $slc1\_slc2.unw_mcf0.bmp
581  endif
582
583  #=====#
584  #do_calibration1
585  if ($do_calibration1) then    # Perform a calibration of the SLC data
586  if (-e $slc1.cslc.par) rm -f $slc1.cslc.par
587  if (-e $slc1.cslc) rm -f $slc1.cslc
588  if (-e $slc1.cml.par) rm -f $slc1.cml.par
589  if (-e $slc1.cml) rm -f $slc1.cml
590  if ($sensor == "PALSAR" && $imgtype == "SLC") then
    #these next two lines works, but for concistentency, the calibration was done
591  after multi-looking
    #radcal_SLC $slc1.slc $slc1.slc.par $slc1.cslc $slc1.cslc.par 1 - 0 0 1 0 -115.0
    radcal_SLC $slc1.slc $slc1.slc.par $slc1.cslc $slc1.cslc.par 1 - 0 0 1 0 -115.0 1
592  #added the 1 to get correct sigma0 values.
    multi_look $slc1.cslc $slc1.cslc.par $slc1.cml $slc1.cml.par $range_looks
593  $azimuth_looks
594  #radcal_SLC $slc1.slc $slc1.slc.par $slc1.cslc $slc1.cslc.par 1 - 0 0 1 0 -
595
596
    #Alternative (first multilook, then calibrate - does not work without the
597  calibration factor of -115. Otherwise the final db values are in the range of 80-100.
    # therefore, for PALSAR, the radcal_SLC and multi_look was done, instead of
598  multilook and radcal_MLI as for ASAR and TSX.
599

```

```

600 #multi_look $slc1.slc $slc1.slc.par $slc1.mli $slc1.mli.par $range_looks
601 $azimuth_looks
602 #radcal_MLI $slc1.mli $slc1.mli.par - $slc1.cml
603 #create a copy of the .mli.par file and name it the .cml.par file for further use
604 #cp $slc1.mli.par $slc1.cml.par
605 #rm -f $pol.cslc $pol.cslc.par $pol.cml
606 endif #ifPALSAR
607 if ($sensor == "ASAR" && $imgtype == "SLC") then
608 #radcal_SLC $slc1.slc $slc1.slc.par $slc1.cslc $slc1.cslc.par 1 - 0 0 1 0 -
609 multi_look $slc1.slc $slc1.slc.par $slc1.mli $slc1.mli.par $range_looks
610 $azimuth_looks
611 #rm -f $pol.cslc $pol.cslc.par $pol.cml
612 radcal_MLI $slc1.mli $slc1.mli.par - $slc1.cml
613 #create a copy of the .mli.par file and name it the .cml.par file for further use
614 cp $slc1.mli.par $slc1.cml.par
615 endif #ifASAR
616 if ($sensor == "TSX" && $imgtype == "SLC") then
617 #radcal_SLC $slc1.slc $slc1.slc.par $slc1.cslc $slc1.cslc.par 1 - 0 0 1 0 -
618 multi_look $slc1.slc $slc1.slc.par $slc1.mli $slc1.mli.par $range_looks
619 $azimuth_looks
620 #rm -f $pol.cslc $pol.cslc.par $pol.cml
621 radcal_MLI $slc1.mli $slc1.mli.par - $slc1.cml
622 #create a copy of the .mli.par file and name it the .cml.par file for further use
623 cp $slc1.mli.par $slc1.cml.par
624 endif #ifTSX
625 #multi-look the .grd image from RS2, already calibrated in prepare_slc step.
626 if ($sensor == "RS2" && $imgtype == "MLI") then
627 multi_look $slc1.grd $slc1.grd_par $slc1.cml $slc1.cml_par $range_looks
628 $azimuth_looks
629 endif #ifRS2
630 #create .bmp file of multi-looked image before and after calibration
631 set image_width = `awk '{if($1 == "range_samples:" ) printf("%d", $2)}' <
632 $slc1.mli.par`
633 echo ""
634 echo "image_width:"
635 echo $image_width
636 echo ""
637 raspwr $slc1.mli $image_width 1 0 1 1 1.0 0.35 $flipping $slc1.mli.bmp
638 if (-e $slc1.mli_flip.bmp) rm -f $slc1.mli_flip.bmp
639 #convert -flip $slc1.mli.bmp $slc1.mli_flip.bmp
640 rm -f $slc1.mli.bmp
641 if (-e $slc1.cml_flip.bmp) rm -f $slc1.cml_flip.bmp
642 raspwr $slc1.cml $image_width 1 0 1 1 1.0 0.35 $flipping $slc1.cml.bmp

```

```

642 #convert -flip $slc1.cmlt.bmp $slc1.cmlt_flip.bmp
643 rm -f $slc1.cmlt.bmp
644 endif
645
646 #=====#
647 if ($do_calibration2) then    # Perform a calibration of the SLC data
648   if (-e $slc1.cslc.par) rm -f $slc1.cslc.par
649   if (-e $slc1.cslc) rm -f $slc1.cslc
650   if (-e $slc2.cslc.par) rm -f $slc2.cslc.par
651   if (-e $slc2.cslc) rm -f $slc2.cslc
652   if (-e $slc1.cmlt.par) rm -f $slc1.cmlt.par
653   if (-e $slc1.cmlt) rm -f $slc1.cmlt
654   if (-e $slc2.cmlt.par) rm -f $slc2.cmlt.par
655   if (-e $slc2.cmlt) rm -f $slc2.cmlt
656   radcal_SLC $slc1.slc $slc1.slc.par $slc1.cslc $slc1.cslc.par 1 - 0 0 1 0 -115.0
657   radcal_SLC $slc2.slc $slc2.slc.par $slc2.cslc $slc2.cslc.par 1 - 0 0 1 0 -115.0
658   multi_look $slc1.cslc $slc1.cslc.par $slc1.cmlt $slc1.cmlt.par $range_looks
659   $azimuth_looks
660   multi_look $slc2.cslc $slc2.cslc.par $slc2.cmlt $slc2.cmlt.par $range_looks
661   $azimuth_looks
662   if (-e $slc1.cmlt_flip.bmp) rm -f $slc1.cmlt_flip.bmp
663   if (-e $slc2.cmlt_flip.bmp) rm -f $slc2.cmlt_flip.bmp
664   raspwr $slc1.cmlt $image_width 1 0 1 1 1.0 0.35 $flipping $slc1.cmlt.bmp
665   #convert -flip $slc1.cmlt.bmp $slc1.cmlt_flip.bmp
666   raspwr $slc2.cmlt $image_width 1 0 1 1 1.0 0.35 $flipping $slc2.cmlt.bmp
667   #convert -flip $slc2.cmlt.bmp $slc2.cmlt_flip.bmp
668   rm -f $slc1.cmlt.bmp $slc2.cmlt.bmp
669   endif
670
671 #do_calibration alternative - 20 March 2013 - Bernard Edit first Multilook, then
672 calibrate
673 #if ($do_calibration) then    # Perform a calibration of the SLC data
674 # if (-e "$slc1.mli") rm -f $slc1.mli
675 # if (-e "$slc2.mli") rm -f $slc2.mli
676 # if (-e "$slc1.mli.par") rm -f $slc1.mli.par
677 # if (-e "$slc2.mli.par") rm -f $slc2.mli.par
678 # if (-e "$slc1.cmlt") rm -f $slc1.cmlt
679 # if (-e "$slc2.cmlt") rm -f $slc2.cmlt
680 # multi_look $slc1.slc $slc1.slc.par $slc1.mli $slc1.mli.par $range_looks
681 $azimuth_looks
682 # multi_look $slc2.slc $slc2.slc.par $slc2.mli $slc2.mli.par $range_looks
683 $azimuth_looks
684 # #rm -f $pol.cslc $pol.cslc.par $pol.cmlt
685 # radcal_MLI $slc1.mli $slc1.mli.par - $slc1.cmlt - 0 0 1 0.0 -115.0
686 # radcal_MLI $slc2.mli $slc2.mli.par - $slc2.cmlt - 0 0 1 0.0 -115.0
687
688 #create a copy of the .mli.par file and name it the .cmlt.par file for further use

```

```

684 # cp $slc1.mli.par $slc1.cml.par
685 # cp $slc2.mli.par $slc2.cml.par
686
687 #bmp image of un-calibrated files
688 # if (-e $slc1.mli_flip.bmp) rm -f $slc1.mli_flip.bmp
689 # if (-e $slc2.mli_flip.bmp) rm -f $slc2.mli_flip.bmp
690 # raspwr $slc1.mli $image_width 1 0 1 1 1.0 0.35 $flipping $slc1.mli.bmp
691 # #convert -flip $slc1.mli.bmp $slc1.mli_flip.bmp
692 # raspwr $slc2.mli $image_width 1 0 1 1 1.0 0.35 $flipping $slc2.mli.bmp
693 # #convert -flip $slc2.mli.bmp $slc2.mli_flip.bmp
694 # rm -f $slc1.mli.bmp $slc2.mli.bmp
695
696 #bmp image of calibrated files
697 # if (-e $slc1.cml_flip.bmp) rm -f $slc1.cml_flip.bmp
698 # if (-e $slc2.cml_flip.bmp) rm -f $slc2.cml_flip.bmp
699 # raspwr $slc1.cml $image_width 1 0 1 1 1.0 0.35 $flipping $slc1.cml.bmp
700 # #convert -flip $slc1.cml.bmp $slc1.cml_flip.bmp
701 # raspwr $slc2.cml $image_width 1 0 1 1 1.0 0.35 $flipping $slc2.cml.bmp
702 # #convert -flip $slc2.cml.bmp $slc2.cml_flip.bmp
703 # rm -f $slc1.cml.bmp $slc2.cml.bmp
704 # endif
705
706 #=====#
707 # Run the GTC_MLI script on the MLI images
708 #
709 # first create the dem_par from the Landsat images to be used in the GTC_MLI
710 # script
711 #=====#
712 if ($do_GTC_MLI) then
713   if (-e $slc1.dem_par) rm -f $slc1.dem_par
714   if (-e $slc1.dem) rm -f $slc1.dem
715   if (-e $slc1.dem_seg) rm -f $slc1.dem_seg
716   if (-e $slc1.img) rm -f $slc1.img
717
718   if ($dem_source == srtm) set dem_projection = eqa
719   if ($img_source == landsat) set img_projection = utm
720   if ($img_source == palsar) set img_projection = utm
721
722   #create_dem_par for image outline within landsat reference
723
724   echo "reach1"
725   # create_dem_par of landsat DEM
726   create_dem_par $slc1\_utm.dem_par $slc1.cml.par - -$grd_rsp $grd_rsp <
727   $working_dir/DEM/$img_source/create_dem_par_utm_$region
728   #cp $working_dir/DEM/$img_source/$region\_landsat_utm.img $slc1\_utm.dem
729   echo "reach2"

```

```

728 #transform the eqa srtm to utm srtm, using the SAR image and landsat dem_par
729 created above
730 # output: $slc1\_utm.dem
731
732 #transform the eqa srtm to utm srtm, using the SAR image and landsat dem_par
733 created above
734 dem_trans $working_dir/DEM/srtm/$region\_srtm_eqa.dem_par
735 $working_dir/DEM/srtm/$region\_srtm_eqa.dem $slc1\_utm.dem_par
736 $slc1\_utm.dem
737 echo "reach3"
738
739 #display the $slc1\_utm.dem file
740 set dem_width_utm = `awk '{if($1 == "width:") printf("%d", $2)}' <
741 $slc1\_utm.dem_par`
742
743 #create header for the DEM (to be able to open the file in ENVI or other GIS/RS
744 software)
745 ./create_header.csh $slc1\_utm $hemisphere
746
747 #create a bitmap .bmp file to easily see the result
748 raspwr $slc1\_utm.dem $dem_width_utm 1 0 1 1 1. .35 1 $slc1\_utm.dem.bmp 0
749 0
750
751 echo ""
752 echo "dem_width_utm"
753 echo $dem_width_utm
754 echo ""
755
756 # run the GTC_MLI script
757 ./GTC_MLI_edit.csh $slc1.mli $slc1.mli.par $slc1\_utm.dem $slc1\_utm.dem_par
758 $slc1\_output.img $slc1\_utm_dem_seg.dem $slc1\_utm_dem_seg.dem_par 1 1 1
759 $init_thres $second_thres
760
761 # create header file for the geocorrected image
762 cp $slc1\_utm_dem_seg.dem_par $slc1.dem_par
763 set pixels = `awk '{if($1 == "width:") printf("%d", $2)}' < $slc1.dem_par`
764 set lines = `awk '{if($1 == "nlines:") printf("%d", $2)}' < $slc1.dem_par`
765 set ul_easting = `awk '{if($1 == "corner_east:") printf("%.3f", $2)}' <
766 $slc1.dem_par`
767 set ul_northing = `awk '{if($1 == "corner_north:") printf("%.3f", $2)}' <
768 $slc1.dem_par`
769 set pixel_size = `awk '{if($1 == "post_east:") printf("%.10f", $2)}' < $slc1.dem_par`
770 set projection = `awk '{if($1 == "projection_name:") printf("%s", $2)}' <
771 $slc1.dem_par`
772 set utm_zone = `awk '{if($1 == "projection_zone:") printf("%d", $2)}' <
773 $slc1.dem_par`

```

```

764 set envi_hdr_file = $slc1\_output.hdr
765 echo "ENVI" > $envi_hdr_file
766 echo "description = {" >> $envi_hdr_file
767 echo " File Imported into ENVI. }" >> $envi_hdr_file
768 echo "samples = $pixels" >> $envi_hdr_file
769 echo "lines = $lines" >> $envi_hdr_file
770 echo "bands = 1" >> $envi_hdr_file
771 echo "header offset = 0" >> $envi_hdr_file
772 echo "file type = ENVI Standard" >> $envi_hdr_file
773 echo "data type = 4" >> $envi_hdr_file
774 echo "interleave = bsq" >> $envi_hdr_file
    echo "sensor type = UNKNOWN" >> $envi_hdr_file
    echo "byte order = 1" >> $envi_hdr_file
775 echo "map info = {$projection, 1.0000, 1.0000, $ul_easting, $ul_northing,
776 $pixel_size, $pixel_size, $utm_zone, $hemisphere, WGS-84, units=Meters}" >>
777 $envi_hdr_file
778 echo "wavelength units = Unknown" >> $envi_hdr_file
779 echo "band names = {" >> $envi_hdr_file
780 echo "}" >> $envi_hdr_file

781 #copy the files to the output folder
782 echo "move the output files from process to output folder, to easily copy for
further steps"
    cd $working_dir
783 set output_folder =
784 $output_name\_ $range_looks\x$azimuth_looks\_ $grd_pixel_spacing\m_ $geocode
785 _path
786 set output_dir = $region\_output/$output_folder
787
788 if (! (-d $working_dir/$output_dir) ) mkdir $working_dir/$output_dir
789 echo ""
790 echo "output_dir"
791 echo $output_dir
792
793 #clean the output directory of any existing files in it.
794 cd $working_dir/$output_dir
795 #if (-e $working_dir/$output_dir/$slc1.*) rm -f $slc1.*
796 #if (-e $working_dir/$output_dir/$output_folder.*) rm -f $output_folder.*
797 cd $working_dir/$process_dir
798
    mv *_output.* $working_dir/$output_dir
799 cp $slc1.slc.par $working_dir/$output_dir
    mv $working_dir/$output_dir/$slc1\.slc.par
800 $working_dir/$output_dir/$output_folder\.slc.par
    mv $working_dir/$output_dir/$slc1\_output.hdr
801 $working_dir/$output_dir/$output_folder\.hdr

```

```

802 mv $working_dir/$output_dir/$slc1\_output.img
803 $working_dir/$output_dir/$output_folder\.img
804 mv $working_dir/$output_dir/$slc1\_output.img.bmp
805 $working_dir/$output_dir/$output_folder\.bmp
806 endif
807
808
809 #####
810 # Geocoding using srtm & Landsat      ##
811 #####
812
813 #####
814 if ($do_geocode1) then
815     if (-e $slc1.dem_par) rm -f $slc1.dem_par
816     if (-e $slc1.dem) rm -f $slc1.dem
817     if (-e $slc1.dem_seg) rm -f $slc1.dem_segset
818     if (-e $slc1.img) rm -f $slc1.img
819     if (-e $slc1.rough.utm_to_rdc) rm -f $slc1.rough.utm_to_rdc
820     if (-e $slc1.utm.sim_sar) rm -f $slc1.utm.sim_sar
821     if (-e $slc1\_utm.dem_par) rm -f $slc1\_utm.dem_par
822     if (-e $slc1\_utm.dem) rm -f $slc1\_utm.dem
823     if (-e $slc1.u) rm -f $slc1.u
824     if (-e $slc1.v) rm -f $slc1.v
825     if (-e $slc1.inc) rm -f $slc1.inc
826     if (-e $slc1.psi) rm -f $slc1.psi
827     if (-e $slc1.pix) rm -f $slc1.pix
828     if (-e $slc1.ls_map) rm -f $slc1.ls_map
829     #Remove the .bmp files
830     if (-e $slc1.dem.bmp) rm -f $slc1.dem.bmp
831     if (-e $slc1.dem_seg.bmp) rm -f $slc1.dem_seg.bmp
832     if (-e $slc1.img.bmp) rm -f $slc1.img.bmp
833     if (-e $slc1.rough.utm_to_rdc.bmp) rm -f $slc1.rough.utm_to_rdc.bmp
834     if (-e $slc1.utm.sim_sar.bmp) rm -f $slc1.utm.sim_sar.bmp
835     if (-e $slc1\_utm.dem.bmp) rm -f $slc1\_utm.dem.bmp
836     if (-e $slc1.u.bmp) rm -f $slc1.u.bmp
837     if (-e $slc1.v.bmp) rm -f $slc1.v.bmp
838     if (-e $slc1.inc.bmp) rm -f $slc1.inc.bmp
839     if (-e $slc1.psi.bmp) rm -f $slc1.psi.bmp
840     if (-e $slc1.pix.bmp) rm -f $slc1.pix.bmp
841     if (-e $slc1.ls_map.bmp) rm -f $slc1.ls_map.bmp
842
843     if ($dem_source == srtm) set dem_projection = eqa
844     if ($img_source == landsat) set img_projection = utm
845     if ($img_source == palsar) set img_projection = utm
846
847     #create_dem_par - eqa

```

```

# create_dem_par $working_dir/DEM/srtm/cameroon1_srtm_eqa.dem_par <
845 $working_dir/DEM/srtm/create_dem_par_eqa_cameroon1
846
# create_dem_par for image outline within landsat reference
847 create_dem_par $slc1\_utm.dem_par $slc1.cml.par - -$grd_rsp $grd_rsp <
848 $working_dir/DEM/$img_source/create_dem_par_utm_$region

# transform the eqa srtm to utm srtm, using the SAR image and landsat dem_par
849 created above
850 # output: $slc1\_utm.dem
851 dem_trans $working_dir/DEM/srtm/$region\_srtm_eqa.dem_par
$working_dir/DEM/srtm/$region\_srtm_eqa.dem $slc1\_utm.dem_par
852 $slc1\_utm.dem
853
# display the $slc1\_utm.dem file
854 set dem_width_utm = `awk '{if($1 == "width:") printf("%d", $2)}' <
855 $slc1\_utm.dem_par`
856 ./create_header.csh $slc1\_utm $hemisphere
857 raspwr $slc1\_utm.dem $dem_width_utm 1 0 1 1 1. .35 1 $slc1\_utm.dem.bmp 0
858 0
859 echo ""
860 echo "dem_width_utm"
861 echo $dem_width_utm
echo ""

862 # derive initial geocoding lookup table and SAR intensity image simulation
863 # original gc_map, before .pix and other options were added
#gc_map $slc1.cml.par - $slc1\_utm.dem_par $slc1\_utm.dem
$slc1\_utm_dem_seg.dem_par $slc1\_utm_dem_seg.dem $slc1.rough.utm_to_rdc
864 - - $slc1.utm.sim_sar
865 # added .pix and other options, by bspies, 28/10/2013
gc_map $slc1.cml.par - $slc1\_utm.dem_par $slc1\_utm.dem
$slc1\_utm_dem_seg.dem_par $slc1\_utm_dem_seg.dem $slc1.rough.utm_to_rdc
- - $slc1.utm.sim_sar - - $slc1.inc - $slc1.pix $slc1.ls_map 8 - -

#gc_map $slc1.mli.par -
866 $scratch_dir/$location/$season/$sensor/Z_SRTM/$image_dir1/$srtm_name\_utm.
867 dem_par
868 $scratch_dir/$location/$season/$sensor/Z_SRTM/$image_dir1/$srtm_name\_utm.
869 dem $slc1.dem_seg_par $slc1.dem_seg $slc1.rough.utm_to_rdc_lut - -
$slc1.utm.sim_sar $slc1.u $slc1.v $slc1.inc $slc1.psi $slc1.pix $slc1.ls_map 8 - -
870
871 ./create_header.csh $slc1\_utm_dem_seg hemisphere
872 cp $slc1\_utm_dem_seg.hdr $slc1.utm.sim_sar.hdr
873 set dem_width = `awk '{if($1 == "width:") printf("%d", $2)}' <
874 $slc1\_utm_dem_seg.dem_par`
875

```



```

876 echo ""
877 echo "dem_width from $slc1\_utm_dem_seg.dem_par"
878 echo $dem_width
879 echo ""

879 if ($add_pixel_area) then
880     # step 1
881     pixel_area $slc1.cml.par $slc1\_utm.dem_par $slc1\_utm.dem
882     $slc1.rough.utm_to_rdc $slc1.ls_map $slc1.inc $slc1.pix_sigma0 $slc1.pix_gamma0
883     10
884     # Calculate offset polynomials
885     echo "dem_diff_par" >> create_diff_par_take3
886     echo "0 0" >> create_diff_par_take3
887     echo "16 16" >> create_diff_par_take3
888     echo "256 256" >> create_diff_par_take3
889     echo $init_thres >> create_diff_par_take3

890     # extra '1' paramater added below, bspies, 20131029
891     create_diff_par $slc1.cml.par - $slc1.diff_par 1 1 < create_diff_par_take3
892     #create_diff_par $slc1.cml.par - $slc1.diff_par 1
893     init_offsetm $slc1.pix_gamma0 $slc1.cml $slc1.diff_par 1 1 - - - $second_thres
894     - 1 > geocode_info2
895     offset_pwr $slc1.pix_gamma0 $slc1.cml $slc1.diff_par $slc1.off $slc1.snr - -
896     $slc1.offsets 2 - - - 1 >> geocode_info2
897     offset_fitm $slc1.off $slc1.snr $slc1.diff_par $slc1.coffs $slc1.coffsets
898     $second_thres - 0 >> geocode_info2
899

900     # Refine the look-up table based on the offset polynomials
901     gc_map_fine $slc1.map_to_rdc $dem_width $slc1.diff_par
902     $slc1.map_to_rdc_fine

903     #Calculate terrain-based sigma/gamma nought normalization area in slant-
904     range geometry
905     # based on the refined geocoding look-up table
906     pixel_area $slc1.cml.par $slc1\_utm_dem_seg.dem_par
907     $$slc1\_utm_dem_seg.dem $slc1.map_to_rdc_fine $slc1.ls_map $slc1.inc
908     $slc1.pix_sigma0_2 $slc1.pix_gamma0_2
909

910     #make bmp to visualise
911     echo "create bmp of pix_sigma0_2 and pix_gamma0_2"
912     raspwr $slc1.pix_sigma0_2 $dem_width 1 0 1 1 1. .35 1 $slc1.pix_sigma0_2.bmp
913     0 0
914     raspwr $slc1.pix_gamma0_2 $dem_width 1 0 1 1 1. .35 1
915     $slc1.pix_gamma0_2.bmp 0 0
916

917     # step 2
918     # separate ASAR and PALSAR and TSX??

```

```

912     echo "step to run radcal_MLI again"
913     radcal_MLI $slc1.mli $slc1.mli.par - $slc1.cml - 0 0 0 0.0 - $slc1.ellip_pix_sigma0
#ratio $slc1.ellip_pix_sigma0 $slc1.pix.sigma0_2 $slc1.sigma0ratio $dem_width
914 1 1
915     #product $slc1.cml $slc1.sigma0ratio $slc1.cml_sigma0norm $dem_width 1 1
#make bmp to visualise
916     raspwr $slc1.ellip_pix_sigma0 $dem_width 1 0 1 1 1. .35 1
$slc1.ellip_pix_sigma0.bmp 0 0

917     # is the geocode_back step still necessary, or is this already achieved with the
918 pixel_area step?
geocode_back $slc1.cml $image_width $slc1.map_to_rdc_fine
919 $slc1_dir1\_cml_utm_$grd_rsp\m_$pol\_ $geocode_path\_new $dem_width 0
$interp_mode 0

920     if (-e $slc1_dir1\_cml_utm_$grd_rsp\m_$pol\_ $geocode_path\_new.bmp) rm -
921 f $slc1_dir1\_cml_utm_$grd_rsp\m_$pol\_ $geocode_path\_new.bmp
922     raspwr $slc1_dir1\_cml_utm_$grd_rsp\m_$pol\_ $geocode_path\_new
$dem_width 1 0 1 1 1.0 0.35 1
923 $slc1_dir1\_cml_utm_$grd_rsp\m_$pol\_ $geocode_path\_new.bmp
924 endif #add_pixel_area
925
926     raspwr $slc1\_utm_dem_seg.dem $dem_width 1 0 1 1 1. .35 1
927 $slc1\_utm_dem_seg.dem.bmp 0 0
928     raspwr $slc1.utm.sim_sar $dem_width 1 0 1 1 1. .35 1 $slc1.utm.sim_sar.bmp 0 0
929 #raspwr $slc1.u $dem_width 1 0 1 1 1. .35 1 $slc1.u.bmp 0 0
930 #raspwr $slc1.v $dem_width 1 0 1 1 1. .35 1 $slc1.v.bmp 0 0
931 #raspwr $slc1.inc $dem_width 1 0 1 1 1. .35 1 $slc1.inc.bmp 0 0
932 #raspwr $slc1.psi $dem_width 1 0 1 1 1. .35 1 $slc1.psi.bmp 0 0
#raspwr $slc1.pix $dem_width 1 0 1 1 1. .35 1 $slc1.pix.bmp 0 0
933 #raspwr $slc1.ls_map $dem_width 1 0 1 1 1. .35 1 $slc1.ls_map.bmp 0 0
934
935     #create_dem_par - landsat utm
#create_dem_par $working_dir/DEM/landsat/cameroon1_landsat_utm.dem_par
< $working_dir/DEM/landsat/create_dem_par_utm_cameroon1
936
937     #create_dem_par for image outline within landsat reference
#create_dem_par $working_dir/DEM/$region\_ $img_source\_utm.dem_par
$slc1.cml.par - -$grd_rsp $grd_rsp <
938 $working_dir/DEM/$img_source/create_dem_par_utm_$region
939
940     #create_dem_par $working_dir/DEM/cameroon1_landsat_utm.dem_par IMG-HH-
ALPSRP220970110-H1.1__A.cml.par - -25 25 <
$working_dir/DEM/landsat/create_dem_par_utm_cameroon1

941     #map transformation from the landsat utm image to the dem_seg outline
942

```

```

943 map_trans
$working_dir/DEM/$img_source/$region\_ $img_source\_utm.dem_par
944 $working_dir/DEM/$img_source/$region\_ $img_source\_utm.img
945 $slc1\_utm_dem_seg.dem_par $slc1.img
946 cp $slc1\_utm_dem_seg.hdr $slc1.hdr
947 raspwr $slc1.utm.sim_sar $dem_width 1 0 1 1 1. .35 1 $slc1.utm.sim_sar.bmp 0 0
948 #set dem_width = `awk '{if($1 == "width:") } printf("%d", $2)}' <
949 $slc1\_utm_dem_seg.dem_par`
raspwr $slc1.img $dem_width 1 0 1 1 1. .35 1 $slc1.img.bmp 0 0
950 endif
951
952 if ($do_geocode1_next) then
#set dem_width = `awk '{if($1 == "width:") } printf("%d", $2)}' < $slc1.dem_par`
953 set dem_width = `awk '{if($1 == "width:") } printf("%d", $2)}' <
954 $slc1\_utm_dem_seg.dem_par`
955
if ($img_source != -) then
956 geocode $slc1.rough.utm_to_rdc $slc1.img $dem_width $slc1.sim_sar
957 $image_width - 1 0
958 endif
959 #if ($img_source == -) then
960 # geocode $slc1.rough.utm_to_rdc $slc1.utm.sim_sar $dem_width $slc1.sim_sar
961 $image_width - 1 0
962 # endif
963
964 if (-e $slc1.diff_par) rm -f $slc1.diff_par
965 if (-e geocode_info) rm -f geocode_info
966 echo "before create_diff_par"
967 # line below from GTC_MLI.csh script
968 #echo 'geo_par 0 0 24 24 128 128 $second_thres' > diff_par.in
969 #echo 'dem_diff_par 0 0 16 16 256 256 $second_thres' > diff_par.in
970
971 #create_diff_par $MLI_par.tmp - $diff_par 1 < diff_par.in
972 #create_diff_par $slc1.cml.par - $slc1.diff_par 1 < diff_par.in
973
974 echo "dem_diff_par" >> create_diff_par_take2
975 echo "0 0" >> create_diff_par_take2
976 echo "16 16" >> create_diff_par_take2
echo "256 256" >> create_diff_par_take2
977 echo $init_thres >> create_diff_par_take2
978
# extra '1' paramater added below, bspies, 20131029
979 create_diff_par $slc1.cml.par - $slc1.diff_par 1 1 < create_diff_par_take2
980 #create_diff_par $slc1.cml.par - $slc1.diff_par 1 <
$working_dir/DEM/create_diff_par
981
982

```

```

983 # threshold was 7.0, change to $second_thres (several places further down, in
984 offset_pwrmm
985 echo "before init_offset"
986 init_offsetm $slc1.cml $slc1.sim_sar $slc1.diff_par - - - - - $second_thres - 0 >
987 geocode_info
988 echo "just after init_offset"
989 if (-e $slc1.gc_offs) rm -f $slc1.gc_offs
990 if (-e $slc1.gc_snr) rm -f $slc1.gc_snr
991 if (-e $slc1.gc_offsets) rm -f $slc1.gc_offsets
992 if (-e $slc1.gc_coffs) rm -f $slc1.gc_coffs
993 if (-e $slc1.gc_coffsets) rm -f $slc1.gc_coffsets
994 #put repeat steps here
995 if ($img_source == -) then
996     offset_pwrmm $slc1.cml $slc1.sim_sar $slc1.diff_par $slc1.gc_offs $slc1.gc_snr 256
997     256 $slc1.gc_offsets 1 8 8 $second_thres
998     offset_fitm $slc1.gc_offs $slc1.gc_snr $slc1.diff_par $slc1.gc_coffs
999     $slc1.gc_coffsets $second_thres 3 0
1000     offset_pwrmm $slc1.cml $slc1.sim_sar $slc1.diff_par $slc1.gc_offs $slc1.gc_snr 128
1001     128 $slc1.gc_offsets 2 24 24 $second_thres
1002     offset_fitm $slc1.gc_offs $slc1.gc_snr $slc1.diff_par $slc1.gc_coffs
1003     $slc1.gc_coffsets $second_thres 3 0 >> geocode_info
1004     endif
1005 if ($img_source != -) then
1006     offset_pwrmm $slc1.cml $slc1.sim_sar $slc1.diff_par $slc1.gc_offs $slc1.gc_snr 128
1007     128 - - 8 32
1008     offset_fitm $slc1.gc_offs $slc1.gc_snr $slc1.diff_par $slc1.gc_coffs
1009     $slc1.gc_coffsets - 1 0
1010     offset_pwrmm $slc1.cml $slc1.sim_sar $slc1.diff_par $slc1.gc_offs $slc1.gc_snr 128
1011     128 - - 16 64
1012     offset_fitm $slc1.gc_offs $slc1.gc_snr $slc1.diff_par $slc1.gc_coffs
1013     $slc1.gc_coffsets - 1 0
1014     offset_pwrmm $slc1.cml $slc1.sim_sar $slc1.diff_par $slc1.gc_offs $slc1.gc_snr 128
1015     128 - - 16 64
1016     offset_fitm $slc1.gc_offs $slc1.gc_snr $slc1.diff_par $slc1.gc_coffs
1017     $slc1.gc_coffsets - 3 0
1018     offset_pwrmm $slc1.cml $slc1.sim_sar $slc1.diff_par $slc1.gc_offs $slc1.gc_snr 128
1019     128 - - 24 96
1020     offset_fitm $slc1.gc_offs $slc1.gc_snr $slc1.diff_par $slc1.gc_coffs
1021     $slc1.gc_coffsets - 4 0
1022     offset_pwrmm $slc1.cml $slc1.sim_sar $slc1.diff_par $slc1.gc_offs $slc1.gc_snr 96
1023     96 - - 24 96
1024     offset_fitm $slc1.gc_offs $slc1.gc_snr $slc1.diff_par $slc1.gc_coffs
1025     $slc1.gc_coffsets - 4 0
1026     offset_pwrmm $slc1.cml $slc1.sim_sar $slc1.diff_par $slc1.gc_offs $slc1.gc_snr 96
1027     96 - - 32 96
1028     offset_fitm $slc1.gc_offs $slc1.gc_snr $slc1.diff_par $slc1.gc_coffs
1029     $slc1.gc_coffsets - 4 0 >> geocode_info

```

```

1013     endif
1014
1015     if ($onemore_init_offsetm) then
1016         echo "test final accuracy estimate - added by bspies 28/10/2013, recommendation
1017         by Andreas Wiesmann"
1018         init_offsetm $slc1.cml_i $slc1.sim_sar $slc1.diff_par - - - - - $second_thres - 0 >>
1019         geocode_info
1020     endif
1021
1022     # offset_pwr_m $slc1.cml_i $slc1.sim_sar $slc1.diff_par $slc1.gc_offs $slc1.gc_snr
1023     256 256 $slc1.gc_offsets 1 8 8 $second_thres
1024     # offset_fitm $slc1.gc_offs $slc1.gc_snr $slc1.diff_par $slc1.gc_coffs
1025     $slc1.gc_coffs $second_thres 3
1026     # offset_pwr_m $slc1.cml_i $slc1.sim_sar $slc1.diff_par $slc1.gc_offs $slc1.gc_snr
1027     256 256 $slc1.gc_offsets 1 16 16 $second_thres
1028     # offset_fitm $slc1.gc_offs $slc1.gc_snr $slc1.diff_par $slc1.gc_coffs
1029     $slc1.gc_coffs $second_thres 3
1030     #put repeat steps here
1031
1032     if (-e $slc1.utm_to_rdc) rm -f $slc1.utm_to_rdc
1033     gc_map_fine $slc1.rough.utm_to_rdc $dem_width $slc1.diff_par $slc1.utm_to_rdc
1034     0
1035     if (-e $slc1_dir1\_cml_i_utm_$grd_rsp\_m_$pol\_$_geocode_path) rm -f
1036     $slc1_dir1\_cml_i_utm_$grd_rsp\_m_$pol\_$_geocode_path
1037     #if ($slc2_dir1 != -) then
1038     #if (-e $slc2_dir1.cml_i.utm_$grd_rsp\_m_$pol) rm -f
1039     $slc2_dir1.cml_i.utm_$grd_rsp\_m_$pol
1040     #if (-e $slc1_dir1\_$_slc2_dir1.cc.utm_$grd_rsp\_m_$pol) rm -f
1041     $slc1\_$_slc2_dir1.cc.utm_$grd_rsp\_m_$pol
1042     #endif
1043     geocode_back $slc1.cml_i $image_width $slc1.utm_to_rdc
1044     $slc1_dir1\_cml_i_utm_$grd_rsp\_m_$pol\_$_geocode_path $dem_width 0
1045     $interp_mode 0
1046     cp $slc1\_utm_dem_seg.dem_par
1047     $slc1_dir1\_cml_i_utm_$grd_rsp\_m_$pol\_$_geocode_path.par_file
1048     #if ($slc2_dir1 != -) then
1049     #if (-e $slc2_dir1.cml_i.utm_$grd_rsp\_m_$pol.par_file) rm -f
1050     $slc2_dir1.cml_i.utm_$grd_rsp\_m_$pol.par_file
1051     #if (-e $slc1_dir1\_$_slc2_dir1.cc.utm_$grd_rsp\_m_$pol.par_file) rm -f
1052     $slc1_dir1\_$_slc2_dir1.cc.utm_$grd_rsp\_m_$pol.par_file
1053
1054     #geocode_back $slc2.cml_i $image_width $slc1.utm_to_rdc
1055     $slc2_dir1.cml_i.utm_$grd_rsp\_m_$pol $dem_width 0 3 0
1056     #geocode_back $slc1\_$_slc2_dir1.cc $image_width $slc1.utm_to_rdc
1057     $slc1_dir1\_$_slc2_dir1.cc.utm_$grd_rsp\_m_$pol $dem_width 0 1 0
1058     #cp $slc1.dem_par $slc2_dir1.cml_i.utm_$grd_rsp\_m_$pol.par_file

```

```

1043 #cp $slc1.dem_par $slc1_dir1\_ $slc2_dir1.cc.utm_$grd_rsp\m_$pol.par_file
1044 #endif
    if (-e $slc1_dir1\_cmli_utm_$grd_rsp\m_$pol\_ $geocode_path.bmp) rm -f
1045 $slc1_dir1\_cmli_utm_$grd_rsp\m_$pol\_ $geocode_path.bmp
    raspwr $slc1_dir1\_cmli_utm_$grd_rsp\m_$pol\_ $geocode_path $dem_width 1 0
1046 1 1 1.0 0.35 1 $slc1_dir1\_cmli_utm_$grd_rsp\m_$pol\_ $geocode_path.bmp

1047 #if ($slc2_dir1 != -) then
    #if (-e $slc2_dir1.cmli.utm_$grd_rsp\m_$pol.bmp) rm -f
1048 $slc2_dir1.cmli.utm_$grd_rsp\m.bmp_$pol
1049 #if (-e $slc1_dir1\_ $slc2_dir1.cc.utm_$grd_rsp\m_$pol.bmp) rm -f
1050 $slc1_dir1\_ $slc2_dir1.cc.utm_$grd_rsp\m_$pol.bmp
1051 #raspwr $slc2_dir1.cmli.utm_$grd_rsp\m_$pol $dem_width 1 0 1 1 1.0 0.35 1
1052 $slc2_dir1.cmli.utm_$grd_rsp\m_$pol.bmp
1053 #ras_linear $slc1_dir1\_ $slc2_dir1.cc.utm_$grd_rsp\m_$pol $dem_width 1 0 1 1
1054 0.0 1.0 1 $slc1_dir1\_ $slc2_dir1.cc.utm_$grd_rsp\m_$pol.bmp
1055 #endif
1056 #endif
1057 endif #endif: do_geocode1_next
1058
1059 #display the image interactively
1060 #dispwr $slc1_dir1\_cmli_utm_$grd_rsp\m_$pol\_ $geocode_path $dem_width
1061 #dispwr 20070824_FBD_422_7140.cmli.utm_25m_HV 3228
1062
1063 #####
1064 ## Create ENVI header file ##
1065 #####
    if($do_header1) then
1066 cp $slc1\_utm_dem_seg.dem_par $slc1.dem_par
    #./create_header.csh $slc1 $hemisphere

1067
1068 set pixels = `awk '{if($1 == "width:") printf("%d", $2)}' < $slc1.dem_par`
    set lines = `awk '{if($1 == "nlines:") printf("%d", $2)}' < $slc1.dem_par`
1069 set ul_easting = `awk '{if($1 == "corner_east:") printf("%.3f", $2)}' <
    $slc1.dem_par`
1070 set ul_northing = `awk '{if($1 == "corner_north:") printf("%.3f", $2)}' <
    $slc1.dem_par`
1071
1072 set pixel_size = `awk '{if($1 == "post_east:") printf("%.10f", $2)}' < $slc1.dem_par`
1073 set projection = `awk '{if($1 == "projection_name:") printf("%s", $2)}' <
    $slc1.dem_par`
1074
1075 set utm_zone = `awk '{if($1 == "projection_zone:") printf("%d", $2)}' <
    $slc1.dem_par`
1076
1077
1078 set envi_hdr_file = $slc1_dir1.hdr
1079 echo "ENVI" > $envi_hdr_file
1080 echo "description = {" >> $envi_hdr_file
1081 echo " File Imported into ENVI. }" >> $envi_hdr_file

```

```

1082 echo "samples = $pixels" >> $envi_hdr_file
1083 echo "lines = $lines" >> $envi_hdr_file
1084 echo "bands = 1" >> $envi_hdr_file
    echo "header offset = 0" >> $envi_hdr_file
    echo "file type = ENVI Standard" >> $envi_hdr_file
1085 echo "data type = 4" >> $envi_hdr_file
1086 echo "interleave = bsq" >> $envi_hdr_file
1087 echo "sensor type = UNKNOWN" >> $envi_hdr_file
1088 echo "byte order = 1" >> $envi_hdr_file
1089 echo "map info = {$projection, 1.0000, 1.0000, $ul_easting, $ul_northing,
1090 $pixel_size, $pixel_size, $utm_zone, $hemisphere, WGS-84, units=Meters}" >>
1091 $envi_hdr_file
1092 echo "wavelength units = Unknown" >> $envi_hdr_file
1093 echo "band names = {" >> $envi_hdr_file
1094 echo "}" >> $envi_hdr_file
1095
1096 cp $slc1_dir1.hdr $slc1_dir1\_cmli_utm_$grd_rsp\_m_$pol\_ $geocode_path.hdr
1097 #if ($slc2_dir1 != -) then
1098     #cp $slc1_dir1.hdr $slc2_dir1.cmli.utm_$grd_rsp\_m_$pol.hdr
1099     #cp $slc1_dir1.hdr $slc1_dir1\_ $slc2_dir1.cc.utm_$grd_rsp\_m_$pol.hdr
1100     #endif
1101 #rm -f $slc1_dir1.hdr
    endif
1102
    if($do_header2) then
1103     cp $slc1\_utm_dem_seg.dem_par $slc1.dem_par
1104     set pixels = `awk '{if($1 == "width:") printf("%d", $2)}' < $slc1.dem_par`
1105     set lines = `awk '{if($1 == "nlines:") printf("%d", $2)}' < $slc1.dem_par`
1106     set ul_easting = `awk '{if($1 == "corner_east:") printf("%.3f", $2)}' <
    $slc1.dem_par`
1107     set ul_northing = `awk '{if($1 == "corner_north:") printf("%.3f", $2)}' <
    $slc1.dem_par`
1108     set pixel_size = `awk '{if($1 == "post_east:") printf("%.10f", $2)}' < $slc1.dem_par`
1109     set projection = `awk '{if($1 == "projection_name:") printf("%s", $2)}' <
    $slc1.dem_par`
1110     set utm_zone = `awk '{if($1 == "projection_zone:") printf("%d", $2)}' <
    $slc1.dem_par`
1111
1112     set envi_hdr_file = $slc1_dir1.hdr
1113     echo "ENVI" > $envi_hdr_file
1114     echo "description = {" >> $envi_hdr_file
1115     echo " File Imported into ENVI. }" >> $envi_hdr_file
1116     echo "samples = $pixels" >> $envi_hdr_file
1117     echo "lines = $lines" >> $envi_hdr_file
1118     echo "bands = 1" >> $envi_hdr_file
1119     echo "header offset = 0" >> $envi_hdr_file
1120     echo "file type = ENVI Standard" >> $envi_hdr_file

```

```

1121 echo "data type = 4" >> $envi_hdr_file
1122 echo "interleave = bsq" >> $envi_hdr_file
1123 echo "sensor type = UNKNOWN" >> $envi_hdr_file
1124 echo "byte order = 1" >> $envi_hdr_file
1125 echo "map info = {$projection, 1.0000, 1.0000, $ul_easting, $ul_northing,
1126 $pixel_size, $pixel_size, $utm_zone, $hemisphere, WGS-84, units=Meters}" >>
1127 $envi_hdr_file
1128 echo "wavelength units = Unknown" >> $envi_hdr_file
1129 echo "band names = {" >> $envi_hdr_file
1130 echo "}" >> $envi_hdr_file
1131
1132 cp $slc1_dir1.hdr $slc1_dir1\_cmli_utm_$grd_rsp\m_$pol\_geocode_path.hdr
1133 #if ($slc2_dir1 != -) then
1134 #cp $slc1_dir1.hdr $slc2_dir1.cmli.utm_$grd_rsp\m_$pol.hdr
1135 #cp $slc1_dir1.hdr $slc1_dir1\_slc2_dir1.cc.utm_$grd_rsp\m_$pol.hdr
1136 #endif
1137 #rm -f $slc1_dir1.hdr
1138 endif
1139
1140 #####
1141 ## Create Google Earth KML file ##
1142 #####

1142 if ($do_kml) then
1143     if (-e $slc1\.eqa_par) rm -f $slc1\.eqa_par
1144     if (-e $slc2\.eqa_par) rm -f $slc2\.eqa_par
1145     if (-e $slc1_dir1.cmli.eqa_$grd_rsp\m_$pol.bmp) rm -f
1146     $slc1_dir1.cmli.eqa_$grd_rsp\m_$pol.bmp
1147     if (-e $slc2_dir1.cmli.eqa_$grd_rsp\m_$pol.bmp) rm -f
1148     $slc2_dir1.cmli.eqa_$grd_rsp\m_$pol.bmp
1149     if (-e $slc1_dir1\_slc2_dir1.cc.eqa_$grd_rsp\m_$pol.bmp) rm -f
1150     $slc1_dir1\_slc2_dir1.cc.eqa_$grd_rsp\m_$pol.bmp
1151     if (-e $slc1_dir1.cmli.eqa_$grd_rsp\m_$pol.bmp.kml) rm -f
1152     $slc1_dir1.cmli.eqa_$grd_rsp\m_$pol.bmp.kml
1153     if (-e $slc2_dir1.cmli.eqa_$grd_rsp\m_$pol.bmp.kml) rm -f
1154     $slc2_dir1.cmli.eqa_$grd_rsp\m_$pol.bmp.kml
1155     if (-e $slc1_dir1\_slc2_dir1.cc.eqa_$grd_rsp\m_$pol.kml) rm -f
1156     $slc1_dir1\_slc2_dir1.cc.eqa_$grd_rsp\m_$pol.kml

1151 create_dem_par $slc1\.eqa_par $slc1\_cmli.par - - $grd_rsp $grd_rsp <
1152 $working_dir/DEM/create_kml
1153 create_dem_par $slc2\.eqa_par $slc2\_cmli.par - - $grd_rsp $grd_rsp <
1154 $working_dir/DEM/create_kml

1153 map_trans $slc1\_utm.dem_par $slc1_dir1.cmli.utm_$grd_rsp\m_$pol.bmp
1154 $slc1\.eqa_par $slc1_dir1.cmli.eqa_$grd_rsp\m_$pol.bmp - - 0 2

```



```

1155 map_trans $slc1\_utm.dem_par $slc2_dir1.cmlt.utm_$grd_rsp\m_$pol.bmp
$slc2\_eqa_par $slc2_dir1.cmlt.eqa_$grd_rsp\m_$pol.bmp - - 0 2
map_trans $slc1\_utm.dem_par
1156 $slc1_dir1\_slc2_dir1.cc.utm_$grd_rsp\m_$pol.bmp $slc1\_eqa_par
1157 $slc1_dir1\_slc2_dir1.cc.eqa_$grd_rsp\m_$pol.bmp - - 0 2
1158 kml_map $slc1_dir1.cmlt.eqa_$grd_rsp\m_$pol.bmp $slc1\_eqa_par
1159 $slc1_dir1.cmlt.eqa_$grd_rsp\m_$pol.kml
1160 kml_map $slc2_dir1.cmlt.eqa_$grd_rsp\m_$pol.bmp $slc2\_eqa_par
1161 $slc2_dir1.cmlt.eqa_$grd_rsp\m_$pol.kml
1162 kml_map $slc1_dir1\_slc2_dir1.cc.eqa_$grd_rsp\m_$pol.bmp $slc1\_eqa_par
1163 $slc1_dir1\_slc2_dir1.cc.eqa_$grd_rsp\m_$pol.kml
1164 endif
1165
1166
1167 #####
# tidy up ##
#####
1168 if ($do_tidy_up) then
echo "move the output files from process to output folder, to easily copy for further
steps"
1169
1170 cd $working_dir
1171 set output_dir =
1172 $region\_output/$output_name\_range_looks\azimuth_looks\_grd_pixel_spa
1173 cing\m_geocode_path
1174 set output_folder =
1175 $output_name\_range_looks\azimuth_looks\_grd_pixel_spacing\m_geocode
1176 _path
1177 if (! (-d $working_dir/$output_dir) ) mkdir $working_dir/$output_dir
1178 cd $working_dir/$output_dir
1179 if (-e $slc1_dir1.hdr) rm *
1180 if (-e $output_folder\.img) rm *
1181
1182 cd $working_dir/$process_dir
1183
mv $slc1_dir1* $working_dir/$output_dir
1184
1185 echo "slc1 $slc1"
1186 echo "slc1_dir $slc1_dir1"

echo "reach1"
1187 cp $slc1.slc.par $working_dir/$output_dir
1188 mv $working_dir/$output_dir/$slc1.slc.par
$working_dir/$output_dir/$output_folder\.slc.par
rm $working_dir/$output_dir/$slc1_dir1.hdr
1189 echo "reach2"
1190

```

```

mv
$working_dir/$output_dir/$slc1_dir1\_cmli_utm_$grd_rsp\m_$pol\_Sgeocode_pat
1191 h $working_dir/$output_dir/$output_folder.img
1192 echo "reach3"
mv
$working_dir/$output_dir/$slc1_dir1\_cmli_utm_$grd_rsp\m_$pol\_Sgeocode_pat
1193 h\.bmp $working_dir/$output_dir/$output_folder.bmp
1194 echo "reach4"
1195 mv
1196 $working_dir/$output_dir/$slc1_dir1\_cmli_utm_$grd_rsp\m_$pol\_Sgeocode_pat
1197 h\.hdr $working_dir/$output_dir/$output_folder.hdr
1198 echo "reach5"
1199 mv
$working_dir/$output_dir/$slc1_dir1\_cmli_utm_$grd_rsp\m_$pol\_Sgeocode_pat
1200 h\.par_file $working_dir/$output_dir/$output_folder.par_file
1201 echo "reach6"
1202
1203 endif #if_do_tidy_up
1204
1205 if ($do_tidy_up_Matt) then
1206   rm -f $slc1.slc $slc2.slc $slc2.rslc $slc1.cslc $slc2.cslc
1207   if (!( -d $working_dir/$process_dir/processing_files )) mkdir
1208   $working_dir/$process_dir/processing_files
1209   mv *.cmli $working_dir/$process_dir/processing_files
1210   mv *.par $working_dir/$process_dir/processing_files
1211   mv *.dem $working_dir/$process_dir/processing_files
1212   mv *.gc* $working_dir/$process_dir/processing_files
1213   mv *.mli $working_dir/$process_dir/processing_files
1214   mv *.utm_to_rdc $working_dir/$process_dir/processing_files
1215   mv *.sim_sar $working_dir/$process_dir/processing_files
1216   mv *.base $working_dir/$process_dir/processing_files
1217   mv *.txt $working_dir/$process_dir/processing_files
1218   mv *.cc* $working_dir/$process_dir/processing_files
1219   mv *.cc $working_dir/$process_dir/processing_files
1220   mv *.flt* $working_dir/$process_dir/processing_files
1221   mv *.int $working_dir/$process_dir/processing_files
1222   mv *.snr $working_dir/$process_dir/processing_files
1223   mv *.rmli $working_dir/$process_dir/processing_files
1224   mv *.off* $working_dir/$process_dir/processing_files
1225   mv *_flip.* $working_dir/$process_dir/processing_files
1226   mv *.coff* $working_dir/$process_dir/processing_files
1227   mv *.dem_par $working_dir/$process_dir/processing_files
   mv *.dem_seg $working_dir/$process_dir/processing_files
   mv *.diff_par $working_dir/$process_dir/processing_files
   mv *.img $working_dir/$process_dir/processing_files
   # mv *.par_file* processing_files
endif

```

```
#cd ..
#cd ..
#chmod -R 755 *
```

To run the automate.csh script in Script A1.1, a script is written that includes a line of input for each SLC SAR image file, and grouped by site, called do\_automate\_site.csh. An extract from the do\_automate\_Tanzania.csh is shown in Script A1.2. To run a .csh file, type ./scriptname.csh in the console to run scriptname.csh.

*Script A1.2 An extract from do\_automate\_Tanzania.csh*

```
1  #Tanzania_Dry_ASAR_20100715_SLC_0_VVVH_VH
2  echo "module load gamma;cd /scratch/gionet/bfs4/phd; ./automate.csh
   /scratch/gionet/bfs4 /scratch/gionet/bfs4/phd tanzania ASAR
   Tanzania_Dry_ASAR_20100715_SLC_0_VVVH -
   Tanzania_Dry_ASAR_20100715_VV_VH.N1
   Tanzania_Dry_ASAR_20100715_SLC_0_VVVH -
   Tanzania_Dry_ASAR_20100715_VV_VH.N1 2 4 VH 15 1 Descending 0 srtm
   landsat 4.000 7.0 South SLC geocode " | qsub -N
   Tanzania_Dry_ASAR_20100715_SLC_0_VVVH_VH_geocode -M bfs4@le.ac.uk -
   l walltime=04:00:00,nodes=1:ppn=1,pvmem=12Gb
3
4  #Tanzania_Dry_PALSAR_20080915_SLC_1_Quad_HH
5  echo "module load gamma;cd /scratch/gionet/bfs4/phd; ./automate.csh
   /scratch/gionet/bfs4 /scratch/gionet/bfs4/phd tanzania PALSAR
   Tanzania_Dry_PALSAR_20080915_SLC_1_Quad - IMG-HH-ALPSRP140877110-
   P1.1__A Tanzania_Dry_PALSAR_20080915_SLC_1_Quad - IMG-HH-
   ALPSRP140877110-P1.1__A 1 3 HH 10 1 Ascending 0 srtm landsat 4.000 7.0
   South SLC geocode " | qsub -N
   Tanzania_Dry_PALSAR_20080915_SLC_1_Quad_HH_geocode -M
   bfs4@le.ac.uk -l walltime=04:00:00,nodes=1:ppn=1,pvmem=12G
6
7  #Tanzania_Dry_TSX_20100920_SLC_0_VVVH_VH
8  echo "module load gamma;cd /scratch/gionet/bfs4/phd; ./automate.csh
   /scratch/gionet/bfs4 /scratch/gionet/bfs4/phd tanzania TSX
   Tanzania_Dry_TSX_20100920_SLC_0_VVVH TSX-
   1.SAR.L1B/TSX1_SAR__SSC____SM_D_SRA_20100920T160312_20100920T1
   60320 IMAGE_VH_SRA_stripNear_006.cos
   Tanzania_Dry_TSX_20100920_SLC_0_VVVH TSX-
   1.SAR.L1B/TSX1_SAR__SSC____SM_D_SRA_20100920T160312_20100920T1
   60320 IMAGE_VH_SRA_stripNear_006.cos 5 2 VH 5 1 Ascending 0 srtm landsat
   4.000 7.0 South SLC geocode " | qsub -N
   Tanzania_Dry_TSX_20100920_SLC_0_VVVH_VH_geocode -M bfs4@le.ac.uk -l
   walltime=04:00:00,nodes=1:ppn=1,pvmem=12Gb
```

Before any of the .csh scripts will work, the permissions needs to be adjusted as in Script A1.3, to make the scripts executable in linux. The \$-sign represents the linux console / terminal.

*Script A1.3 Console / terminal commands to make .csh scripts executable.*

```
$ chmod +x automate.csh
$ chmod +x do_automate_Tanzania.csh
```

The remaining scripts, needs to be located within the directory structure as shown in Figure A1.1 and Figure A1.2.

*Script A1.4 create\_header.csh, located in folder /phd*

```
1  #!/bin/csh -fe
2  set image = $1      #image file name without the .dem_par extension
3  set hemisphere = $2  #North or South
4  #echo "current directory:"
5  #echo pwd
6
7  #cd $1
8  #cd $insardir\_ $range_looks\x$azimuth_looks\_ $pol
9
10 set pixels = `awk '{if($1 == "width:") printf("%d", $2)}' < $image.dem_par`
11 set lines = `awk '{if($1 == "nlines:") printf("%d", $2)}' < $image.dem_par`
12 set ul_easting = `awk '{if($1 == "corner_east:") printf("%.3f", $2)}' <
    $image.dem_par`
13 set ul_northing = `awk '{if($1 == "corner_north:") printf("%.3f", $2)}' <
    $image.dem_par`
14 set pixel_size = `awk '{if($1 == "post_east:") printf("%.10f", $2)}' < $image.dem_par`
15 set projection = `awk '{if($1 == "projection_name:") printf("%s", $2)}' <
    $image.dem_par`
16 set utm_zone = `awk '{if($1 == "projection_zone:") printf("%d", $2)}' <
    $image.dem_par`
17
18 set envi_hdr_file = $image.hdr
19 echo "ENVI" > $envi_hdr_file
20 echo "description = {" >> $envi_hdr_file
21 echo " File Imported into ENVI. }" >> $envi_hdr_file
22 echo "samples = $pixels" >> $envi_hdr_file
23 echo "lines = $lines" >> $envi_hdr_file
24 echo "bands = 1" >> $envi_hdr_file
25 echo "header offset = 0" >> $envi_hdr_file
26 echo "file type = ENVI Standard" >> $envi_hdr_file
27 echo "data type = 4" >> $envi_hdr_file
```

```

27 echo "interleave = bsq" >> $envi_hdr_file
28 echo "sensor type = UNKNOWN" >> $envi_hdr_file
29 echo "byte order = 1" >> $envi_hdr_file
30 echo "map info = {$projection, 1.0000, 1.0000, $ul_easting, $ul_northing,
31 $pixel_size, $pixel_size, $utm_zone, $hemisphere, WGS-84, units=Meters}" >>
  $envi_hdr_file
  echo "wavelength units = Unknown" >> $envi_hdr_file
32 echo "band names = {" >> $envi_hdr_file
33 echo "}" >> $envi_hdr_file
34

```

*Script A1.5 dem\_byte\_swap.csh, located in /phd folder.*

```

1  #!/bin/csh -fe
2
3  set inputfile = $1
4  set outputfile = $2
5
6  swap_bytes $inputfile $outputfile 2

```

#### A1.4.2 /DEM folder scripts

*Script A1.6 create\_diff\_par file, in /DEM folder*

```

1  dem_diff_par
2
3
4
5  4.0
6

```

The create\_kml script in Script A1.7 is not currently implemented, but left here for future reference.

*Script A1.7 create\_kml in /DEM folder*

```
1 EQA
2 WGS84
3 1
4
5
6
7
8
9
10
11
12
13
```

*Script A1.8 create\_offset\_input\_FVI, in /DEM folder*

```
1 interferogram parameters
2
3
4
5
6
7
8
```

*Script A1.9 create\_offset\_input\_ICC, in /DEM folder*

```
1 interferogram parameters
2 0 0
3 32 32
4 64 64
5
6
```

### A1.4.3 /DEM/landsat folder scripts

In the following scripts, when more than one script is presented for different sites, the differences are highlighted in **red** font.

*Script A1.10 create\_dem\_par\_utm\_site files for Cameroon, DRC, Tanzania and Chad, in /DEM/landsat folder.*

create\_dem\_par\_cameroon

```
1 UTM
2 WGS84
3 1
4 33
5
6 REAL*4
7
8
9
10
11
12
```

create\_dem\_par\_drc

```
1 UTM
2 WGS84
3 1
4 34
5
6 REAL*4
7
8
9
10
11
12
```

create\_dem\_par\_tanzania

```
1 UTM
2 WGS84
3 1
4 36
5
6 REAL*4
7
8
9
10
11
12
```

create\_dem\_par\_chad

```
1 UTM
2 WGS84
3 1
4 34
5
6 REAL*4
7
8
9
10
11
12
```

All the \_utm.dem\_par files (Scripts A1.11-A1.14) are created by taking the values from the original .hdr file. The false\_easting and false\_northing values needed for the UTM zones shown in Table A1.1. These values are then used in Scripts A1.11-A1.14 as required.

Table A1.1 The false easting and false northing values for the northern and southern hemisphere.

	For Northern Hemisphere	For Southern Hemisphere
false_easting:	500000.000 m	500000.000 m
false_northing:	0.000 m	10000000.000 m

Script A1.11 cameroon\_landsat\_utm.dem\_par

```

1 Gamma DIFF&GEO DEM/MAP parameter file
2 title: DEM
3 DEM_projection: UTM
4 data_format: REAL*4
5 DEM_hgt_offset: 0.00000
6 DEM_scale: 1.00000
7 width: 29181
8 nlines: 24721
9 corner_north: 904807.500 m
10 corner_east: 87292.500 m
11 post_north: -15.0000000 m
12 post_east: 15.0000000 m
13
14 ellipsoid_name: WGS 84
15 ellipsoid_ra: 6378137.000 m
16 ellipsoid_reciprocal_flattening: 298.2572236
17
18 datum_name: WGS 1984
19 datum_shift_dx: 0.000 m
20 datum_shift_dy: 0.000 m
21 datum_shift_dz: 0.000 m
22 datum_scale_m: 0.00000e+00
23 datum_rotation_alpha: 0.00000e+00 arc-sec
24 datum_rotation_beta: 0.00000e+00 arc-sec
25 datum_rotation_gamma: 0.00000e+00 arc-sec
26 datum_country_list Global Definition, WGS84, World
27
28 projection_name: UTM
29 projection_zone: 33
30 false_easting: 500000.000 m
31 false_northing: 0.000 m
32 projection_k0: 0.9996000
33 center_longitude: 15.0000000 decimal degrees
34 center_latitude: 0.0000000 decimal degrees

```



```
1 Gamma DIFF&GEO DEM/MAP parameter file
2 title: DEM
3 DEM_projection: UTM
4 data_format: REAL*4
5 DEM_hgt_offset: 0.00000
6 DEM_scale: 1.00000
7 width: 29381
8 nlines: 24682
9 corner_north: 266107.500 m
10 corner_east: 504592.500 m
11 post_north: -15.0000000 m
12 post_east: 15.0000000 m
13
14 ellipsoid_name: WGS 84
15 ellipsoid_ra: 6378137.000 m
16 ellipsoid_reciprocal_flattening: 298.2572236
17
18 datum_name: WGS 1984
19 datum_shift_dx: 0.000 m
20 datum_shift_dy: 0.000 m
21 datum_shift_dz: 0.000 m
22 datum_scale_m: 0.00000e+00
23 datum_rotation_alpha: 0.00000e+00 arc-sec
24 datum_rotation_beta: 0.00000e+00 arc-sec
25 datum_rotation_gamma: 0.00000e+00 arc-sec
26 datum_country_list Global Definition, WGS84, World
27
28 projection_name: UTM
29 projection_zone: 35
30 false_easting: 500000.000 m
31 false_northing: 0.000 m
32 projection_k0: 0.9996000
33 center_longitude: 27.0000000 decimal degrees
34 center_latitude: 0.0000000 decimal degrees
```

```
1 Gamma DIFF&GEO DEM/MAP parameter file
2 title: DEM
3 DEM_projection: UTM
4 data_format: REAL*4
5 DEM_hgt_offset: 0.00000
6 DEM_scale: 1.00000
7 width: 29281
```

```

8  nlines:      24621
9  corner_north: 9785207.500 m
10 corner_east:  249892.500 m
11 post_north:   -15.0000000 m
12 post_east:    15.0000000 m
13
14 ellipsoid_name: WGS 84
15 ellipsoid_ra:   6378137.000 m
16 ellipsoid_reciprocal_flattening: 298.2572236
17
18 datum_name: WGS 1984
19 datum_shift_dx: 0.000 m
20 datum_shift_dy: 0.000 m
21 datum_shift_dz: 0.000 m
22 datum_scale_m:  0.00000e+00
23 datum_rotation_alpha: 0.00000e+00 arc-sec
24 datum_rotation_beta: 0.00000e+00 arc-sec
25 datum_rotation_gamma: 0.00000e+00 arc-sec
26 datum_country_list Global Definition, WGS84, World
27
28 projection_name: UTM
29 projection_zone:      36
30 false_easting:      500000.000 m
31 false_northing:     1000000.000 m
32 projection_k0:      0.9996000
33 center_longitude:   33.0000000 decimal degrees
34 center_latitude:    0.0000000 decimal degrees

```

*Script A1.14 chad\_landsat\_utm.dem\_par*

```

1  Gamma DIFF&GEO DEM/MAP parameter file
2  title: DEM
3  DEM_projection:  UTM
4  data_format:    REAL*4
5  DEM_hgt_offset:  0.00000
6  DEM_scale:      1.00000
7  width:      28981
8  nlines:     24701
9  corner_north: 1704307.500 m
10 corner_east:  451492.500 m
11 post_north:   -15.0000000 m
12 post_east:    15.0000000 m
13
14 ellipsoid_name: WGS 84
15 ellipsoid_ra:   6378137.000 m

```

```

16 ellipsoid_reciprocal_flattening: 298.2572236
17
18 datum_name: WGS 1984
19 datum_shift_dx:      0.000 m
20 datum_shift_dy:      0.000 m
21 datum_shift_dz:      0.000 m
22 datum_scale_m:      0.00000e+00
23 datum_rotation_alpha: 0.00000e+00 arc-sec
24 datum_rotation_beta: 0.00000e+00 arc-sec
25 datum_rotation_gamma: 0.00000e+00 arc-sec
26 datum_country_list Global Definition, WGS84, World
27
28 projection_name: UTM
29 projection_zone:      34
30 false_easting:      500000.000 m
31 false_northing:      0.000 m
32 projection_k0:      0.9996000
33 center_longitude:    21.0000000 decimal degrees
34 center_latitude:     0.0000000 decimal degrees

```

#### A1.4.4 /DEM/landsat/landsat\_preparation folder

The landsat images are first combined in a mosaic (where necessary) and then converted to float format in ENVI. The \_utm\_float images are then copied to /DEM/landsat/landsat\_preparation folder on ALICE. This is then use to create the site\_landsat\_utm.img and site\_landsat\_utm.dem\_par files in the /DEM/landsat/ folder. The files are named:

site\_landsat\_utm\_float and  
site\_landsat\_utm\_float.hdr,

with site replaced with the specific site-name as appropriate. The do\_dem\_byte\_swap needs to be run, for the automate.csh script to read the correct format of the DEM file.

*Script A1.15 dem\_byte\_swap*

```

1  #!/bin/csh -fe
2  set inputfile = $1
3  set outputfile = $2
4  swap_bytes $inputfile $outputfile 4

```

*Script A1.16 do\_dem\_byte-swap*

```
1 #byte swap in gamma
2 module load gamma
3 #./dem_byte_swap.csh                cameroon1_landsat_wide_utm_float
4 cameroon1_landsat_utm_new.dem
5 ./dem_byte_swap.csh                cameroon2_landsat_utm_float
6 cameroon2_landsat_utm.img
7 #./dem_byte_swap.csh chad_landsat_utm_float chad_landsat_utm.img
8 #./dem_byte_swap.csh drc_landsat_utm_float drc_landsat_utm.img
9 #./dem_byte_swap.csh sudan_landsat_utm_float sudan_landsat_utm.img
10 #./dem_byte_swap.csh tanzania_landsat_utm_float tanzania_landsat_utm.img
11
12 # move the output files to the main /DEM/landsat folder
13 #mv cameroon1_landsat_utm.img ../cameroon1_landsat_utm.img
14 mv cameroon2_landsat_utm.img ../cameroon2_landsat_utm.img
15 #mv chad_landsat_utm.img ../chad_landsat_utm.img
16 #mv drc_landsat_utm.img ../drc_landsat_utm.img
17 #mv sudan_landsat_utm.img ../sudan_landsat_utm.img
18 #mv tanzania_landsat_utm.img ../tanzania_landsat_utm.img
```

#### A1.4.5 /DEM/palsar

This folder is used to geo-terrain correct to a PALSAR image, as was done by Waldram (2014). In this study this was not used, but is kept for future reference.

#### A1.4.6 /DEM/srtm

The site\_srtm\_eqa.dem\_par files accompany each a site\_srtm\_eqa.dem file, as shown in Scripts A1.17-A1.20.

*Script A1.17 cameroon1\_srtm\_eqa.dem\_par*

```
1 Gamma DIFF&GEO DEM/MAP parameter file
2 title: DEM
3 DEM_projection:  EQA
4 data_format:    INTEGER*2
5 DEM_hgt_offset:  0.00000
6 DEM_scale:      1.00000
7 width:          6000
8 nlines:         6000
9 corner_lat:     10.000000 decimal degrees
```

```

10 corner_lon: 10.0000000 decimal degrees
11 post_lat: -8.3333330e-04 decimal degrees
12 post_lon: 8.3333330e-04 decimal degrees
13
14 ellipsoid_name: WGS 84
15 ellipsoid_ra: 6378137.000 m
16 ellipsoid_reciprocal_flattening: 298.2572236
17
18 datum_name: WGS 1984
19 datum_shift_dx: 0.000 m
20 datum_shift_dy: 0.000 m
21 datum_shift_dz: 0.000 m
22 datum_scale_m: 0.00000e+00
23 datum_rotation_alpha: 0.00000e+00 arc-sec
24 datum_rotation_beta: 0.00000e+00 arc-sec
25 datum_rotation_gamma: 0.00000e+00 arc-sec
26 datum_country_list Global Definition, WGS84, World

```

*Script A1.18 drc\_srtm\_eqa.dem\_par*

```

1 Gamma DIFF&GEO DEM/MAP parameter file
2 title: DEM
3 DEM_projection: EQA
4 data_format: INTEGER*2
5 DEM_hgt_offset: 0.00000
6 DEM_scale: 1.00000
7 width: 12002
8 nlines: 12002
9 corner_lat: 5.00000000 decimal degrees
10 corner_lon: 24.99958382 decimal degrees
11 post_lat: -8.3333330e-04 decimal degrees
12 post_lon: 8.3333330e-04 decimal degrees
13
14 ellipsoid_name: WGS 84
15 ellipsoid_ra: 6378137.000 m
16 ellipsoid_reciprocal_flattening: 298.2572236
17
18 datum_name: WGS 1984
19 datum_shift_dx: 0.000 m
20 datum_shift_dy: 0.000 m
21 datum_shift_dz: 0.000 m
22 datum_scale_m: 0.00000e+00
23 datum_rotation_alpha: 0.00000e+00 arc-sec
24 datum_rotation_beta: 0.00000e+00 arc-sec
25 datum_rotation_gamma: 0.00000e+00 arc-sec
26 datum_country_list Global Definition, WGS84, World

```

```
1 Gamma DIFF&GEO DEM/MAP parameter file
2 title: DEM
3 DEM_projection: EQA
4 data_format: INTEGER*2
5 DEM_hgt_offset: 0.00000
6 DEM_scale: 1.00000
7 width: 6002
8 nlines: 12003
9 corner_lat: 0.00041725 decimal degrees
10 corner_lon: 29.99958333 decimal degrees
11 post_lat: -8.3333330e-04 decimal degrees
12 post_lon: 8.3333330e-04 decimal degrees
13
14 ellipsoid_name: WGS 84
15 ellipsoid_ra: 6378137.000 m
16 ellipsoid_reciprocal_flattening: 298.2572236
17
18 datum_name: WGS 1984
19 datum_shift_dx: 0.000 m
20 datum_shift_dy: 0.000 m
21 datum_shift_dz: 0.000 m
22 datum_scale_m: 0.00000e+00
23 datum_rotation_alpha: 0.00000e+00 arc-sec
24 datum_rotation_beta: 0.00000e+00 arc-sec
25 datum_rotation_gamma: 0.00000e+00 arc-sec
26 datum_country_list Global Definition, WGS84, World
```

```
1 Gamma DIFF&GEO DEM/MAP parameter file
2 title: DEM
3 DEM_projection: EQA
4 data_format: INTEGER*2
5 DEM_hgt_offset: 0.00000
6 DEM_scale: 1.00000
7 width: 6000
8 nlines: 6000
9 corner_lat: 15.00000000 decimal degrees
10 corner_lon: 20.00000000 decimal degrees
11 post_lat: -8.3333330e-04 decimal degrees
12 post_lon: 8.3333330e-04 decimal degrees
13
14 ellipsoid_name: WGS 84
15 ellipsoid_ra: 6378137.000 m
16 ellipsoid_reciprocal_flattening: 298.2572236
```

```

17
18 datum_name: WGS 1984
19 datum_shift_dx:      0.000  m
20 datum_shift_dy:      0.000  m
21 datum_shift_dz:      0.000  m
22 datum_scale_m:       0.00000e+00
23 datum_rotation_alpha: 0.00000e+00 arc-sec
24 datum_rotation_beta:  0.00000e+00 arc-sec
25 datum_rotation_gamma: 0.00000e+00 arc-sec
26 datum_country_list Global Definition, WGS84, World

```

#### A1.4.7 /DEM/srtm/srtm\_preparation folder

Copy the site\_srtm\_eqa\_beforebyteswap.dat and site\_srtm\_eqa\_beforebyteswap.hdr files to the /DEM/SRTM/SRTM\_preparation folder.

Run the do\_dem\_byte\_swap\_srtm.csh script (Script A1.22) to convert the SRTM to the correct format. This will enable the automate.csh script (Script A1.1) to run correctly.

*Script A1.21 dem\_byte\_swap*

```

1  #!/bin/csh -fe
2
3  set inputfile = $1
4  set outputfile = $2
5
6  swap_bytes $inputfile $outputfile 2
7

```

*Script A1.22 do\_dem\_byte\_swap*

```

1  #byte swap in gamma
2  module load gamma
3  #./dem_byte_swap.csh      cameroon1_srtm_eqa_beforebyteswap.dem
   cameroon1_srtm_eqa_new
4  #./dem_byte_swap.csh      cameroon2_srtm_eqa_beforebyteswap
   cameroon2_srtm_eqa.dem
5  #./dem_byte_swap.csh      chad_srtm_eqa_beforebyteswap.dat
   chad_srtm_eqa.dem
6  #./dem_byte_swap.csh drc_srtm_eqa_beforebyteswap.dat drc_srtm_eqa.dem

```

```

7 #./dem_byte_swap.csh                sudan_srtm_eqa_beforebyteswap.dat
  sudan_srtm_eqa.dem
8 ./dem_byte_swap.csh                tanzania_srtm_eqa_beforebyteswap.dat
  tanzania_srtm_eqa.dem
9 #./dem_byte_swap.csh                drc_srtm_eqa_beforebyteswap_4tiles.dat
  drc_srtm_eqa.dem

10 # move the output files to the main /DEM/srtm folder
11 #mv cameroon1_srtm_eqa.dem ../cameroon1_srtm_eqa.dem
12 #mv cameroon2_srtm_eqa.dem ../cameroon2_srtm_eqa.dem
13 #mv chad_srtm_eqa.dem ../chad_srtm_eqa.dem
14 #mv drc_srtm_eqa.dem ../drc_srtm_eqa.dem
15 #mv sudan_srtm_eqa.dem ../sudan_srtm_eqa.dem
16 mv tanzania_srtm_eqa.dem ../tanzania_srtm_eqa.dem
17 #mv drc_srtm_eqa.dem ../drc_srtm_eqa.dem
18

```

## A1.5 Additional Linux commands and notes

This section presents a handful of linux commands that proved useful during this research.

To load GAMMA on ALICE:

```
$ module load gamma
```

Available modules:

```
$ module avail
```

To see if your script is running on ALICE, replace bfs4 with your username (it might take about 10-15 seconds, or longer before the job appears on the queue):

```
$ showq -u bfs4
```

Any errors are shown in files named (1234567 changes after each run):

```
Gamma_Proc.e1234567
```

The screen output is all written to files named (1234567 changes after each run):

```
Gamma_Proc.o1234567
```

List files and directories, with or without the -l:

```
$ ls -l
```

List files by time:

```
$ ls -l -t
```



List files by time in reverse:

```
$ ls -ltr
```

Current directory:

```
$ pwd
```

To run the .csh file:

```
$ ./filename.csh
```

To copy and paste in Linux:

Mark the text to be copied in the terminal. Click with the middle mouse button where you want to paste the text

Move / rename a file:

```
$ mv filename_old filename_new
```

Copy a file:

```
$ cp location1\filename location2\filename
```

Change from icon-view to list view in the file browser on linux:

Cntrl-2

Change directory to the parent directory, one level up:

```
$ cd ..
```

Copy directories with all sub-directories and files (recursively) (see <http://www.tuxfiles.org/linuxhelp/dirman.html>):

```
$ cp -r dir1 dir2
```

Renaming directories, the same as renaming file-names:

```
$ mv dir1 dir2
```

Remove all files starting with e.g. Gamma\_Proc, by using the wildcard (\*) character:

```
$ rm Gamma_Proc.*
```

Or move all Gamma\_Proc files to the extras directory:

```
$ mv Gamma_Proc.* extras
```

Get the size of a whole folder:

```
$ du -sh
```

To kill a running job:

```
$ qdel <jobid>
```

e.g.

309

## **Appendix Two: GDAL scripts to create image overlap**

Appendix Two provides the scripts to perform the pre-processing steps of creation of an image overlap (Section 4.2), calculation of the derived layers (Section 4.3) and combining the images into a layer stack (Section 4.4). The scripts are a combination of bash, which is the GNU shell scripting language (FSF, 2014a, 2014b), the Geospatial Data Abstraction Library (GDAL) scripts (GDAL, 2013), and GRASS GIS (Neteler et al., 2012).

### **A2.1 Overview of GDAL scripts**

Once the GAMMA orthorectification scripts are run, all the output files are located in the folders /site\_output, as shown in Figure A1.1. Copy this whole folder onto a local computer, to the new base directory, called /step2\_after\_geocode, using WinSCP or a similar program.

All the scripts in Appendix Two, apart from Scripts 2.9-2.14, should also be placed in the new base directory, and can be run from there. Scripts 2.9-2.14, which creates the additional layers of slope and aspect from the SRTM elevation images for each site, should be run from the /Z\_SRTM\_UTM folder, which is on the same level as the base directory.

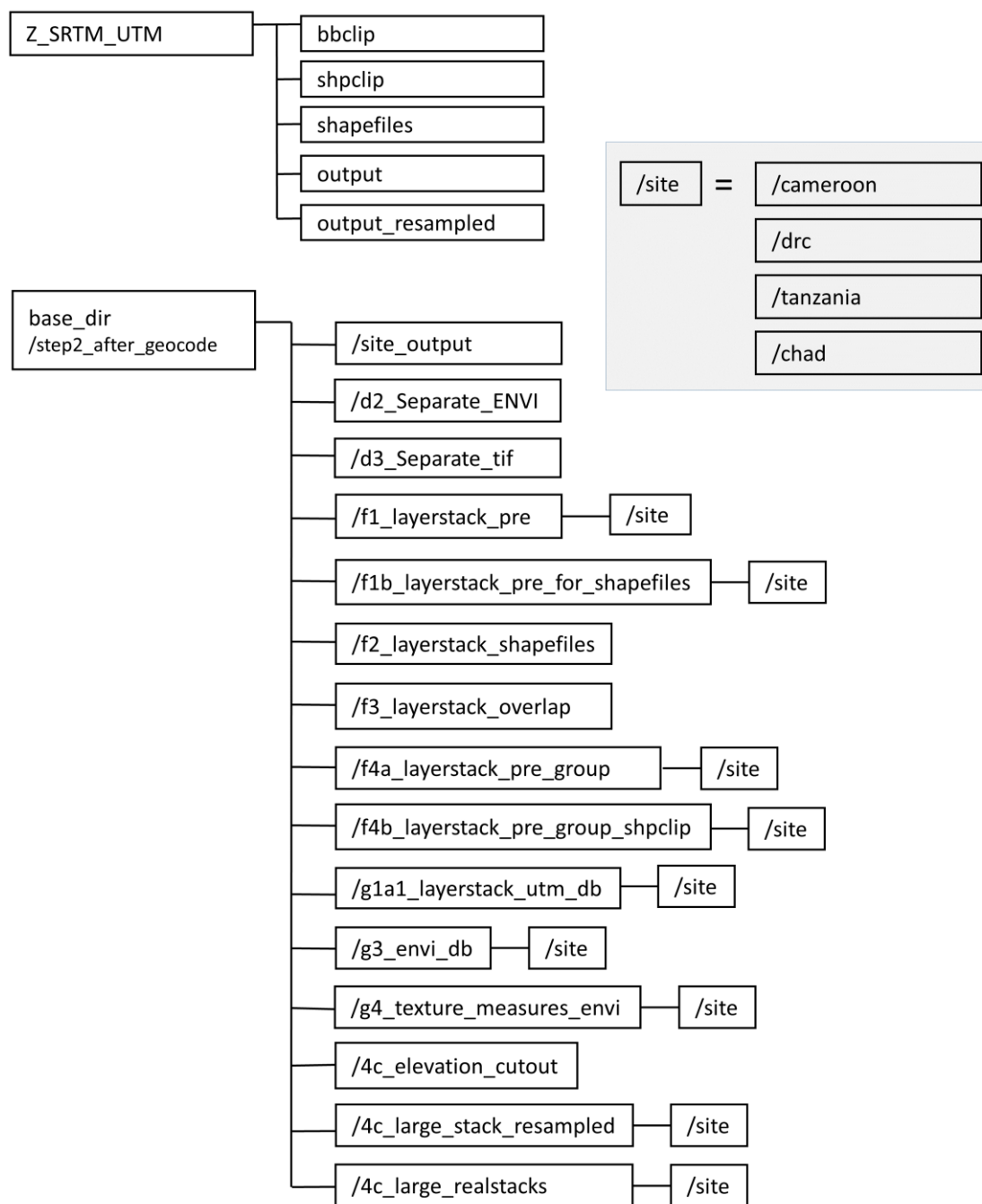


Figure A2.1 Folder structure for the GDAL scripts data preparation steps.

Table A2.1 Structure of bash scripts for data preparation.

Script name	Script number
<i>Unfold output files to new folder</i> d1_unfold_copy_to_site_output_geocode_folder.sh	A2.1
<i>Create overlap</i> do_overlap_create.sh d2_convert_envi_to_tif.sh e_mosaic_images.sh f1a_prepare_tif_directories.sh f1_create_layer_stack.sh f2_unfold_shapefiles.sh f3_intersect_shapefiles.sh	A2.2 A2.3 A2.4 A2.5 A2.6 A2.7 A2.8
<i>Cutout DEM scripts</i> do_demcalc_cutout_pre.sh demcalc_cutout_pre.sh do_demcalc_allsites.sh (in GRASS) demcalc_allsites.sh (in GRASS) do_srtm_resample.sh srtm_resample.sh	A2.9 A2.10 A2.11 A2.12 A2.13 A2.14
<i>Calculated additional layers and create data stack</i> do_processflow.sh f4b_complete_stack_envi.sh g2b2_bandcalc_ENVI.sh texture_calc_take2.sh ( in GRASS) g3b_copy_to_largestack_folder.sh g3b_envi_virtualstack.sh g3b_envi_realstacks.sh	A2.15 A2.16 A2.17 A2.18 A2.19 A2.20 A2.21
<i>Additional scripts before R modelling</i> do_convert_to_raster_all.sh convert_to_raster_all.sh	A2.22 A2.23

## A2.2 GDAL bash scripts

### A2.2.1 Unfold multiple folders into a single folder

First copy the site\_output folders to the new base directory, as shown in Figure A2.1. Copy Script A2.1 into the site\_output folder for each site, and run the script from inside this folder. The image files are unfolded into the /d2\_Separate\_ENVI folder.

*Script A2.1 d1\_unfold\_copy\_to\_site\_output\_geocode\_folder.sh*

```
1  #!/bin/bash
2
3  #if [ -f dirlist ] ; then
4  # rm -f dirlist
5  #fi
6  #ls -d */ >> dirlist
7  #for dir in $(dirlist);
8
9  # see      http://stackoverflow.com/questions/2107945/how-to-loop-over-directories-in-linux
10
11 # Notes:
12 # copy this file to the d1_site_output_geocode folder
13
14 for dir in */;
15 do
16     #echo dir
17     dir=${dir%*/}
18     echo ${dir##*/}
19     cd ${dir##*/}
20
21     #cp *.img ../../Tanzania_geocodeG
22     #cp *.hdr ../../Tanzania_geocodeG
23     cp *.img ../../d2_Separate_ENVI
24     cp *.hdr ../../d2_Separate_ENVI
25     cd ../
26 done
```

### A2.2.2 Creating an overlap between multiple shapefiles script

The `do_overlap_create.sh` script in Script A2.2 calls Scripts A2.3-A2.8. Parameters can be set for all of these scripts from Script A2.2.

*Script A2.2 `do_overlap_create.sh`*

```
1  #!/bin/bash
2
3  #file to create the shapefile overlap, before the do_processflow.sh
   programs
4
5  #site='Cameroon1'
6  #site='Cameroon2'
7  #site='Chad'
8  #site='DRC'
9  #site='Sudan'
10 site='Tanzania'
11 do_mosaic_images=0;
12 do_resample=0;
13 do_create_shapefile_outlines=0;
14 do_unfold_shapefiles=0; #check manually if all shapefiles were created -
   otherwise create new shapefiles in QGIS
15 do_intersect_shapefiles=1;
16
17 if [ $do_mosaic_images == 1 ];
18 then
19     ./d2_convert_envi_to_tif.sh
20     ./e_mosaic_images.sh
21 fi
22
23 if [ $do_resample == 1 ];
24 then
25     ./f1_create_layer_stack.sh $site 1 0
26 fi
27
28 if [ $do_create_shapefile_outlines == 1 ];
29 then
30     ./f1_create_layer_stack.sh $site 0 1
31 fi
32
33 if [ $do_unfold_shapefiles == 1 ];
34 then
35     ./f2_unfold_shapefiles.sh $site
```

```

35 fi
36
37 if [ $do_intersect_shapefiles == 1 ];
38 then
39     ./f3_intersect_shapefiles.sh $site geocodeG
40     ./f3_intersect_shapefiles.sh $site geocodeS
41 fi

```

*Script A2.3 d2\_convert\_envi\_to\_tif.sh*

```

1  #!/bin/bash
2  # program to convert ENVI files to .tif files using gdal.
3  # Bernard Spies, 11 July 2013
4
5  #copy this convert_envi_to_tif.sh script to the parent folder of
   d_Separate_ENVI and d_Separate_tif
6  #run the program using ./convert_envi_to_tif.sh from the parent folder
7
8  #step 1 convert from envi to tif
9  cd ./d2_Separate_ENVI/
10
11 for filename in *.img;
12 do
13     echo $filename
14     gdal_translate -of GTiff -a_nodata 0 $filename
       ../d3_Separate_tif/$filename.tif
15 done
16
17 #change back to parent directory for step 2
18 cd ..
19
20 #step 2: rename all the .img.tif filenames to .tif filenames
21 #see http://stackoverflow.com/questions/1224766/how-do-i-rename-the-extension-of-a-batch-of-files
22 cd ../d3_Separate_tif/
23
24 for filename in *.img.tif;
25 do
26     mv "$filename" "`basename $filename .img.tif`.tif"
27 done
28 cd ..
29
30 #test line
31 #gdal_translate -of GTiff -a_nodata 0
32 Cameroon1_Dry_ASAR_20100411_SLC_0_VV_VH_VH_2x4_15m_geocode.
   img ../d_Separate_tif/test.tif

```

Script A2.4 *e\_mosaic\_images.sh*

```
1  #!/bin/bash
2
3  #this program mosaics the _1_ and _2_ images into _3_, to cover a
   larger area, using gdal
4
5  cd ./d3_Separate_tif
6
7  #mkdir {process1and2,output3}
8  #mkdir {process1,process2,process3}
9
10 mv *_1_* ./process1
11 mv *_2_* ./process2
12
13 cd ./process2
14
15 for x in *_2_*;
16 do
17     #echo $x
18     mv $x ${x/_2/_1_};
19 done
20
21 #mv *_2_* *_1_*
22 for file in *geocode.tif;
23 do
24     mv $file "`basename $file geocode.tif`geocode_nr2.tif"
25     #mv $file "`basename $file geocode.tif.aux.xml`geocode_nr2"
26 done
27
28 for file in *geocode.tif.aux.xml;
29 do
30     mv $file "`basename $file
   _geocode.tif.aux.xml`_geocode_nr2.tif.aux.xml"
31 done
32 cd ..
33
34 cd ./process1
35 for file in *geocode.tif;
36 do
37     mv $file "`basename $file geocode.tif`geocode_nr1.tif"
38     #mv $file "`basename $file geocode.tif.aux.xml`geocode_nr2"
39 done
40
41 for file in *geocode.tif.aux.xml;
42 do
43
```



```

44 mv $file "`basename $file
45 _geocode.tif.aux.xml`_geocode_nr1.tif.aux.xml"
46 done
47
48 cd ..
49
50 #mkdir process3
51 mv ./process1/*.tif ./process3/
52 mv ./process2/*.tif ./process3/
53 cd ./process3
54
55 #cd ./process1and2
56
57 for file1 in *.tif;
58 #
59 # #for file1 in [find ./process1/ -maxdepth 1 -type f -name '*.tif'];
60 # #for file1 in [cd ./process1; ./ -maxdepth 1 -type f -name '*.tif'];
61 #
62 do
63     for file2 in *.tif;
64     do
65         #echo "file1: $file1"
66         #echo "file2: $file2"
67         test1=`basename $file1 _nr1.tif`
68         test2=`basename $file2 _nr2.tif`
69         #if [ "basename $file1 _nr1.tif" == "basename $file1 _nr2.tif" ]
70         if [ "$test1" == "$test2" ]
71         then
72             #if [ *_1_.tif == *_2_.tif ]
73             echo "file1 nr1: $file1"
74             echo "file2 nr2: $file2"
75             gdal_merge.py -o ../output3/"`basename $file1
76             _nr1.tif`_mosaic.tif" -of GTiff -n 0 $file1 $file2
77         fi
78     done
79 done
80
81 for x in *_1_*;
82 do
83     #echo $x
84     mv $x ${x/_1/_3_};
85 done
86 mv * ../

```

*Script A2.5 f1a\_prepare\_tif\_directories.sh*

```
1 # 3. clip the raster file to the boundingbox of the shapefile
2 cd ./f1_layerstack_pre
3
4 #create a for loop for all the .tif files
5 #make the separate site directories - only in later steps?!
6 mkdir {Cameroon1,Cameroon2,Chad,DRC,Sudan,Tanzania}
7
8 #move the files to each separate directory
9
10 mv Cameroon1*.tif ./Cameroon1/
11 mv Cameroon2*.tif ./Cameroon2/
12 mv Chad*.tif ./Chad/
13 mv DRC*.tif ./DRC/
14 mv Sudan*.tif ./Sudan/
15 mv Tanzania*.tif ./Tanzania/
```

*Script A2.6 f1\_create\_layer\_stack.sh*

```
1 #!/bin/bash
2 site=$1
3 resample=$2
4 polygons=$3
5
6 #see http://www.techques.com/question/26-30111/Merging-raster-data-with-different-resolutions
7
8 #first test on a test sample of 3 images (1 from TSX, 1 from ASAR, 1 from PALSAR)
9
10 #=====#
11 # Step 1 - resample
12 # gdalwarp with -r bilinear (or nearest neighbor)
13 # resample the lower resolution to get close to the higher resolution data sets
14
15 #loop over each file individually
16 ## test files
17 ##cd ./d_Separate_tif_test
18 ## all the files
19
20 if [ $resample == 1 ];
21 then
22 #cd ./d3_Separate_tif
23 cd ./f1_layerstack_pre/$site
```

```

24
25 #for filename in *.tif;
26 for filename in *.img;
27 #for filename in
28 Cameroon1_Dry_ASAR_20100411_SLC_0_VVHH_VH_2x4_15m_geocod
eG.img;
29
30 ##for filename in
31 Cameroon1_Dry_TSX_20100313_SLC_0_HHHV_HH_5x2_5m_geocode.t
if;
32 do
33     echo $filename
34
35     ## gdalwarp step - resample to 5m
36     #original ##gdalwarp -tr 5 5 -r near -srcnodata 0 -dstnodata 0 -
dstalpha -of GTiff -overwrite $filename
    ../f1b_layerstack_pre_for_shapefiles/$site/"`basename $filename
    .img`_resampled.tif"
37
38     #gdalwarp -tr 5 5 -r near -srcnodata 0 -dstnodata 0 -dstalpha -of GTiff -
overwrite $filename
    ../f1b_layerstack_pre_for_shapefiles/$site/"`basename $filename
    .img`_resampled.tif"
39
40     #correctone for ENVI files
41     gdalwarp -r near -srcnodata 0 -dstnodata 0 -dstalpha -of GTiff -
overwrite $filename
42     ../f1b_layerstack_pre_for_shapefiles/$site/"`basename $filename
43     .img`_resampled.tif"
44
45
46     #gdalwarp -tr 5 5 -r near -srcnodata 0 -dstnodata 0 -dstalpha -of ENVI -
overwrite $filename
    ../f1b_layerstack_pre_for_shapefiles/$site/"`basename $filename
    .img`_resampled.img"
47     #gdalwarp -r near -srcnodata 0 -dstnodata 0 -dstalpha -of ENVI -
overwrite $filename
    ../f1b_layerstack_pre_for_shapefiles/$site/"`basename $filename
    .img`_resampled.img"
48 done
49
50 cd .././
51 fi #resample
52
53 if [ $polygons == 1 ];
54 then
55     cd ../f1b_layerstack_pre_for_shapefiles/$site
56

```

```

57 for filename in *.tif;
58 #for filename in *.img;
59 #for filename in
   Cameroon1_Dry_TSX_20100313_SLC_0_HHHV_HH_5x2_5m_geocode_
   resampled.tif;
60
61 do
62     echo $filename
63     #gdal_polygonize.py $filename -b 2 -f "ESRI Shapefile"
   ../../f2_layerstack_shapefiles/$site\_shp/"`basename $filename
   .img`_outline" "`basename $filename .img`_outline"
64     gdal_polygonize.py $filename -b 2 -f "ESRI Shapefile"
   ../../f2_layerstack_shapefiles/$site\_shp/"`basename $filename
   .tif`_outline" "`basename $filename .tif`_outline"
65 done
66 fi #polygons
67
68 #=====#
69 # Step 2 create stack
70
71 #do the gdal_merge on all the files in the folder in one go (not looping
   through each individually)
72
73 #for file in ./d_Separate_tif_test/*
74 #do
75 # gdal_merge -o ./e_layerstack/layerstack3files.tif -o GTiff -ps 5 5 -
   separate -n 0 a_nodata 0 -r #bilinear "$file"
76 #done
77 #step 2 gdal_merge

```

*Script A2.7 f2\_unfold\_shapefiles.sh*

```

1  #!/bin/bash
2  # see http://stackoverflow.com/questions/2107945/how-to-loop-over-
3  directories-in-linux
4
5  # to expand the program see:
6  # http://stackoverflow.com/questions/4842130/how-do-i-count-the-
7  number-of-directories-in-a-path-using-bash
8  # and http://nixcraft.com/shell-scripting/18400-bash-script-count-
9  number-files-directory.html
10
11 site=$1
12 cd ./f2_layerstack_shapefiles/$site\_shp

```

```

12 echo "f2_layerstack_shapefiles/$site\_shp"
13
14 #unfold the shapefiles
15 for dir in ./*/;
16 do
17     dir=${dir%*/}
18     echo ${dir##*/}
19     cd ${dir##*/}
20
21 # mv * ../../f2_layerstack_shapefiles/$site\_shp
22     mv * ../
23     cd ..
24 done
25
26 #remove the original directories, leaving only the shapefiles, all in the
    same directory
27 for dir in ./*/;
28 do
29     dir=${dir%*/}
30     echo "removing: " ${dir##*/}
31     rmdir ${dir##*/}
32 done
33 cd ../../

```

*Script A2.8 f3\_intersect\_shapefiles.sh*

```

1  #!/bin/bash
2  #script to create one shapefile which is the intersect of all the shapefiles
    over the same site.
3
4  # see https://github.com/dwtkns/gdal-cheat-sheet for clipping tips
5  # see also error on GOOMETRYCOLLECTION suggestion:
    http://www.forumsig.org/archive/index.php/t-32130.html
6  # and http://georezo.net/forum/viewtopic.php?id=76424
7
8  sitename=$1
9  geocodeGS=$2
10 echo "sitename: $sitename"
11 echo "$geocodeGS"
12
13 #set counter
14 i=0;
15 cd ./f2_layerstack_shapefiles
16 cd ./${sitename}\_shp
17
18 #for filename in *.shp;

```

```

19 #do
20 #
21 #           #echo  $sitename\_shp_polygons/"`basename  $filename
                .shp\_polygon.shp"
22 #  echo $filename
23 #           ogr2ogr -overwrite ../$sitename\_shp_polygons/"`basename
                $filename .shp\_polygon.shp" -nlt POLYGON $filename
24 #done
25 #cd ../$sitename\_shp_polygons
26
27 #set first file to $tmp variable
28 #for filename in *.shp;
29 for filename in *$geocodeGS*.shp;
30 do
31 #set the first filename
32   if [ $i -eq 0 ]; then
33     echo "shapefile number " $i $filename
34     tmp=$filename;
35     i=$((i+1));
36   fi
37 done
38
39 #loop through all the shapefiles. create a temporary output, and intersect
    each time with the next shapefile in the list
40 # the -skipfailures option is important, otherwise it bombs out, due to
    POINT vs POLYGON discrepancies - not sure how to
41 # get around this without the -skipfailures option(!)
42
43 #for filename in *.shp;
44 for filename in *geocodeG*.shp;
45 do
46   #intersect the next filename with the previous one
47   if [ "$tmp" != "$filename" ]; then
48     echo "shapefile number " $i $filename
49     ogr2ogr      -skipfailures      -overwrite      -clipsrc      $tmp
                ../f3_layerstack_overlap/overlap_$sitename\_$geocodeGS\shp  -nlt
                POLYGON $filename
50
51 tmp=../f3_layerstack_overlap/overlap_$sitename\_$geocodeGS\shp
52   i=$((i+1));
53   fi
54 done

```

### A2.2.3 Elevation data preparation

*Script A2.9 do\_demcalc\_cutout\_pre.sh*

```
1 #!/bin/bash
2 # do_demcalc_cutout.sh to run demcalc_cutout.sh for each site
3 ./demcalc_cutout_pre.sh cameroon1 utm33N 0 1 0 #cameroon1
4 ./demcalc_cutout_pre.sh cameroon2 utm33N 0 1 0 #cameroon2
5 ./demcalc_cutout_pre.sh drc utm35N 0 1 0 #drc
6 ./demcalc_cutout_pre.sh tanzania utm36S 0 1 0 #tanzania
7 ./demcalc_cutout_pre.sh chad utm34N 0 1 0 #chad
8 ./demcalc_cutout_pre.sh sudan utm34N 0 1 0 #sudan
```

*Script A2.10 demcalc\_cutout\_pre.sh*

```
1 #!/bin/bash
2 # demcalc_cutout.sh
3 # 3 Feb 2014, Bernard Spies
4 # similar to f4b_complete_stack.sh in the step2_after_geocode folder
5 # program to cut DEM smaller, for faster processing and less space usage
6 # ideas taken from Tim Sutton's Linfiniti.com blog :
7 # http://linfiniti.com/2009/09/clipping-rasters-with-gdal-using-polygons/
8
9 # Four steps to follow:
10 # 1. create shpfile containing the features you wish to use as a clip mask -
11 DONE
12 # 2. determine the extent of your clipmask
13 # 3. perform a rectangular clip on the image
14 # 4. perform a polygon clip on the image
15 # 5. my added step: then combine these clipped rasters into a layer stack.
16
17 # 2. create shapefile extent
18 ### ==== Run these steps in loops over all the files in the folder,
19 ### ==== then add the final gdalwarp step in the end to create a stack for each
   site
20 site=$1
21 utm=$2
22 do_shapefile_extent=$3 #1=yes, 0=no
23 do_bbclip=$4 #1=yes, 0=no
24 do_shpclip=$5 #1=yes, 0=no
25
26 if [ $do_shapefile_extent == 1 ]
27 then
```

```

28 cd ./shapefiles
29 #SHPFILE=$1
30 SHPFILE=overlap_1.shp
31 BASE=`basename $SHPFILE .shp`
32 EXTENT=`ogrinfo -so $SHPFILE $BASE | grep Extent \
33 | sed 's/Extent: //g' | sed 's/(//g' | sed 's/)//g' \
34 | sed 's/ - /, /g`
35 EXTENT=`echo $EXTENT | awk -F ' ' '{print $1 " " $4 " " $3 " " $2}'`
36 echo $EXTENT
37 # back to base directory
38 cd ..
39 fi
40 #extent: Top-left, Bottom-right
41 # cameroon1 extent: 250955.000000 726170.000000 265250.000000
42 687000.000000
43 # cameroon2 extent: 488360.000000 462455.000000 511350.000000
44 411620.000000
45 # drc extent: 799860.000000 187950.000000 809620.000000
46 133080.000000
47 # tanzania extent: 392660.000000 9577095.000000 403980.000000
48 9532990.000000
49 # chad extent: 608480.000000 1529840.000000 637230.000000
50 1495780.000000
51 # sudan extent: 698415.000000 1458735.000000 714050.000000
52 1444050.000000
53
54 #New extent (added 500m in all directions)
55 #cameroon1 250455.000000 726670.000000 265750.000000
56 686500.000000
57 #cameroon2 487860.000000 462955.000000 511850.000000
58 411120.000000
59 #drc 799360.000000 188450.000000 810120.000000
60 132580.000000
61 #tanzania 392160.000000 9577595.000000 404480.000000
62 9532490.000000
63 #chad 607980.000000 1530340.000000 637730.000000
64 1495280.000000
65 #sudan697915.000000 1459235.000000 714550.000000
66 1443550.000000
67 EXTENT_cameroon1="250455.000000 726670.000000 265750.000000
68 686500.000000"
69 EXTENT_cameroon2="487860.000000 462955.000000 511850.000000
70 411120.000000"
71 EXTENT_drc="799360.000000 188450.000000 810120.000000
72 132580.000000"
73 EXTENT_tanzania="392160.000000 9577595.000000 404480.000000
74 9532490.000000"

```



```

59 EXTENT_chad="607980.000000      1530340.000000 637730.000000
1495280.000000"
60 EXTENT_sudan="697915.000000 1459235.000000 714550.000000
61      1443550.000000"
62
63 # 3. clip the raster file to the boundingbox of the shapefile
64 #cd ./f1_layerstack_pre/$sitename
65 #create a for loop for all the .tif files
66 #RASTERFILE=$2
67
68 ## currently these next three lines are done on ALICE:
69 if [ $do_bbclip == 1 ]
70 then
71     #cd ./f4_layerstack_pre_group/$sitename
72     echo $site
73     echo $utm
74     echo $site\_srtm\_utm.tif
75     echo EXTENT_$site
76     echo $EXTENT_cameroon1
77
78     for RASTERFILE in $site\_srtm\_utm.tif;
79     do
80         echo $RASTERFILE
81         echo EXTENT_$site
82         echo `basename $RASTERFILE .tif`_bbclip.tif
83         if [ $site == "cameroon1" ];
84         then
85             gdal_translate -projwin 250455.000000      726670.000000
265750.000000      686500.000000 -of GTiff $RASTERFILE
./bbclip/`basename $RASTERFILE .tif`_bbclip.tif
            fi
86             if [ $site == "cameroon2" ];
87             then
88                 gdal_translate -projwin 487860.000000      462955.000000
511850.000000      411120.000000 -of GTiff $RASTERFILE
./bbclip/`basename $RASTERFILE .tif`_bbclip.tif
                    fi
90                 if [ $site == "drc" ];
91                 then
92                     gdal_translate -projwin 799360.000000 188450.000000 810120.000000
132580.000000 -of GTiff $RASTERFILE ./bbclip/`basename $RASTERFILE
.tif`_bbclip.tif
                            fi
94                     if [ $site == "tanzania" ];
95                     then
96
97

```

```

    gdal_translate -projwin 392160.000000 9577595.000000
404480.000000 9532490.000000 -of GTiff $RASTERFILE ./bbclip/`basename
98 $RASTERFILE .tif`_bbclip.tif
99     fi
100     if [ $site == "chad" ];
101     then
        gdal_translate -projwin 607980.000000      1530340.000000
637730.000000 1495280.000000 -of GTiff $RASTERFILE ./bbclip/`basename
102 $RASTERFILE .tif`_bbclip.tif
103     fi
104     if [ $site == "sudan" ];
105     then
        gdal_translate -projwin 697915.000000 1459235.000000
714550.000000 1443550.000000 -of GTiff $RASTERFILE ./bbclip/`basename
106 $RASTERFILE .tif`_bbclip.tif
107     fi
        #gdal_translate -projwin EXTENT_$site -of GTiff $RASTERFILE
108 destination_$site.tif
        #gdal_translate -projwin 250455.000000 726670.000000 265750.000000
109 686500.000000 -of GTiff cameroon1_srtm_utm33N.tif test.tif
110     done
111 fi
112
113 # 4. perform polygon clip on the image
114 #for file in *_bbclip.tif;
115 if [ $do_shpclip == 1 ]
116 then
117     cd ./bbclip
118     for RASTERFILE in $site\*_bbclip.tif;
        #for RASTERFILE in
119 Cameroon1_Dry_ASAR_20100411_SLC_0_VVH_VH_2x4_15m_geocode_res
120 amplified_bbclip.tif;
        do
            gdalwarp -co COMPRESS=DEFLATE -co TILED=YES -of GTiff -r lanczos -
cutline ../shapefiles/$SHPFILE -srcnodata 0 -dstnodata -999 `basename
$RASTERFILE _bbclip.tif`_bbclip.tif ../shpclip/`basename $RASTERFILE
121 _bbclip.tif`_shpclip.tif
122     done
        fi

```

*Script A2.11 do\_demcalc\_allsites.sh*

```

1 #!/bin/bash
2 ./demcalc_allsites.sh cameroon1 utm33N
3 ./demcalc_allsites.sh cameroon2 utm33N
4 ./demcalc_allsites.sh drc utm35N

```

```
5 ./demcalc_allsites.sh tanzania utm36S
6 ./demcalc_allsites.sh chad utm34N
7 ./demcalc_allsites.sh sudan utm34N
```

*Script A2.12 demcalc\_allsites.sh*

```
1 #!/bin/bash
2 # demcalc_allsites - calculate elevation, slope and aspect for each site
3 # 3 Feb 2014, Bernard Spies
4 # these commands are to be run in grass
5 # step 1 is to test that it works, if time permits, the bash code can be rewritten
  as a python script
6 #this is a program to calculate the slope and aspect from a DEM
7 site=$1
8 utm=$2
9 #site="cameroon1"
10 #utm="utm33N"
11
12 if [ $site == "cameroon1" ]
13 then
14     g.mapset mapset=cameroon1_dry location=cameroon1_dry
15 fi
16 if [ $site == "cameroon2" ]
17 then
18     g.mapset mapset=cameroon1_dry location=cameroon1_dry
19 fi
20 if [ $site == "drc" ]
21 then
22     g.mapset mapset=drc_dry location=drc_dry
23 fi
24 if [ $site == "tanzania" ]
25 then
26     g.mapset mapset=tanzania_dry location=tanzania_dry
27 fi
28 if [ $site == "chad" ]
29 then
30     g.mapset mapset=chad_dry location=chad_dry
31 fi
32 if [ $site == "sudan" ]
33 then
34     g.mapset mapset=chad_dry location=chad_dry
35 fi
36
37 #site="cameroon2"
38 #utm="utm33N"
39
```

```

40 #read in the DEM file
41 r.in.gdal -e -k
   input=/home/bernard/sharefolder/A_Data_PhD/A_Alldata/Z_SRTM_UTM/bbc
   lip/$site\_srtm\_$utm\_bbclip.tif output=dem_$site
42 g.region rast=dem_$site
43
44 d.mon wx0 #open a display window
45 #d.erase #clean current display
46 d.rast dem_$site
47
48 # --o allow output files to overwrite existing files
49 r.slope.aspect elevation=dem_$site slope=$site.slope aspect=$site.aspect
   format=degrees prec=FCCELL zfactor=1.0 min_slp_allowed=0.0 --o
50
51 #d.rast $site.slope
52 #d.rast $site.aspect
53
54 #r.slope.aspect elevation=dem_$site slope=$site.slope aspect=$site.aspect
   format=degrees prec=float pcurv=dar.pcurv tcurv=dar.tcurv dx=dar.dx
   dy=dar.dy dxx=dar.dxx dyy=dar.dyy dxy=dar.dxy zfactor=1.0
   min_slp_allowed=0.0 --o
55
56 r.mapcalc "$site.slope2 = round($site.slope)" --overwrite
57 r.mapcalc "$site.aspect2 = round($site.aspect)" --overwrite
58
59 #d.rast $site.slope2
60 #d.rast $site.aspect2
61
62
63 #r.out.gdal help
64 #write the files back to TIF or ENVI format
65 #r.out.gdal input=$site.pcurv format=GTiff type=Float32
   output=/home/bspies/hostshare/SRTM/dar_pcurv.tif nodata=-32768
66 #r.out.gdal input=dar.tcurv format=GTiff type=Float32
   output=/home/bspies/hostshare/SRTM/dar_tcurv.tif nodata=-32768
67 r.out.gdal input=dem_$site format=ENVI type=Int16
   output=/home/bernard/sharefolder/A_Data_PhD/A_Alldata/Z_SRTM_UTM/o
   utput/$site\_elevation.dat nodata=-99 --overwrite
68 r.out.gdal input=$site.slope2 format=ENVI type=Int16
69 output=/home/bernard/sharefolder/A_Data_PhD/A_Alldata/Z_SRTM_UTM/o
70 utput/$site\_slope.dat nodata=-99 --overwrite
   r.out.gdal input=$site.aspect2 format=ENVI type=Int16
71 output=/home/bernard/sharefolder/A_Data_PhD/A_Alldata/Z_SRTM_UTM/o
   utput/$site\_aspect.dat nodata=-99 --overwrite

```

Script A2.13 *do\_srtm\_resample.sh*

```
1 #!/bin/bash
2 # do_srtm_resample.sh
3 ./srtm_resample.sh
```

Script A2.14 *srtm\_resample.sh*

```
1 #!/bin/bash
2 # srtm_resample.sh
3 # program to resample SRTM data to 5x5m to be able to use in the layer stack
4 with the SAR data
5
6 cd ./output/
7 for filename in *.dat;
8 do
9     echo $filename
9     gdalwarp -tr 5 5 -r bilinear -dstnodata -999 -of ENVI -overwrite $filename
10    ../output_resampled/"`basename $filename .dat`_resampled.dat"
11 done
```

#### A2.2.4 Calculate additional features layers process flow script

Script A2.15 *do\_processflow.sh*

```
1 #!/bin/bash
2
3 #site='Cameroon1'
4 #site='Cameroon2'
5 #site='Chad'
6 #site='DRC'
7 site='Sudan'
8 #site='Tanzania'
9
10 do_move_files=0
11 do_f4b_shpclip=0          #take the _bbc_clip files and clip them to the
12 shapefiles, NN resampling
13 #do_f4c_complete_stack_tifs=0    #complete the stack
14 #do_g2b_linear_to_db_R=0        #convert from linear to decibels
15 #do_g2c_nodata_values=0        #?? is this step really necessary ??
16 do_g2b2_bandcalc=0
17 #do_g3a_tif_to_envi=0          #input into grass for tif files did not work,
18                               #therefore convert to ENVI again.
18 do_texture_calc=0         #calculate texture in GRASS
19
```

```

do_copy_to_largestack_folder=0 #copy all necessary layers to largestack
20 folder, and resample to 5m resolution (nearest neighbour)
21 do_g3b_envi_virtualstack=1 #create virtualstack
do_g3c_envi_realstacks=0 #create the realstacks from RF modelling, to
22 be used in EnMap
23 do_g3c_trainingtest_cutout=0
24
25 #add elevation, slope and aspect layers (where/when are they calculated?)
26 #move data from d3_separate_tif to f1_layerstack_pre/$site
27 if [ $do_move_files == 1 ] && [ $site == "Cameroon1" ];
28 then
    cd
    /home/bernard/sharefolder/A_Data_PhD/A_Alldata/A_Cameroon1/d3_Sep
29 arate_tif
    mv *
    /home/bernard/sharefolder/A_Data_PhD/A_Alldata/step2_after_geocode/
30 f1_layerstack_pre/Cameroon1
    cd
31 /home/bernard/sharefolder/A_Data_PhD/A_Alldata/step2_after_geocode/
fi
32
33 #take the _bbc_clip files and clip them to the shapefiles
34 # read from (bbclip): ./f1_layerstack_pre/$sitename
35 # write to (bbclip): ./f4a_layerstack_pre_group/$site
36 # read from (shpclip): ./f4a_layerstack_pre_group/$site
37 # writes to (shpclip): ./f4b_layerstack_pre_group_shpclip/$site
38 #sitename=$1
39 #do_bbclip=$2 #1=yes, 0=no
40 #do_shpclip=$3ge #1=yes, 0=no
41 if [ $do_f4b_shpclip == 1 ];
42 then
43     #./f4b_complete_stack.sh $site 1 0
44     ./f4b_complete_stack_envi.sh $site 1 1
45 fi
46
47 #complete the stack
48 # read from: ./f4b_layerstack_pre_group_shpclip/$site
# writes to: ./f4b_layerstack_pre_group_shpclip/$site/$site_stack_all.vrt ,
49 _dry.vrt, _wet.vrt
50
51 #if [ $do_f4c_complete_stack_tifs == 1 ];
52 #then
53 #     ./f4c_complete_stack_tifs.sh $site
54 #fi
55
56 #convert from linear to decibels
57 # read from: ./f4b_layerstack_pre_group_shpclip/$site/

```

```

58 # write to: ./g1a1_layerstack_utm_db/$site
59 #if [ $do_g2b_linear_to_db_R == 1 ];
60 #then
61 #   Rscript g2b_linear_to_db_R.R $site UTM linux >
62 g2b_linear_to_db_R_UTM_$site.results
63 #fi
64 # run a histogram to see the values of the data, and decide if the next step
65 (conversion to NA values)
66 # is necessary or not
67
68 # create virtualstack, convert -3.3999999521444e+38 values to 0 / NA
69 values
70 # read from: ./g1a1_layerstack_utm_db/$sitename
71 # write to: ./g1a2_layerstack_utm_db_NA/$sitename
72 #if [ $do_g2c_nodata_values == 1 ];
73 #then
74 #   #./g2c_create_virtualstack.sh $site
75 #   ./g2c_nodata_values.sh $site
76 #fi
77
78 # create band ratios, create a new virtual stack
79 # read from: g1a1_layerstack_utm_db_NA/$sitename/ (old one)
80 # read from: ./f4b_layerstack_pre_group_shpclip/$site (new calculations)
81 # write to: g1a1_layerstack_utm_db/$sitename/
82 if [ $do_g2b2_bandcalc == 1 ];
83 then
84     #./g2b2_bandcalc.sh $site 1 1
85     ./g2b2_bandcalc_ENVI.sh $site 1 1 0 S
86     ./g2b2_bandcalc_ENVI.sh $site 0 1 0 G
87     ./g2b2_bandcalc_ENVI.sh $site 0 0 1 S
88     ./g2b2_bandcalc_ENVI.sh $site 0 0 1 G
89 fi
90
91 ## convert .tif to ENVI files
92 ## read from: ./g1a2_layerstack_utm_db_NA/$sitename
93 ## write to: ./g3_envi_db/$sitename/
94 #if [ $do_g3a_tif_to_envi == 1 ];
95 #then
96 #   ./g3a_tif_to_envi.sh $site
97 #fi
98
99 # Texture calculations, via GRASS GIS
100 # read from: ./g3_envi_db/$site
101 # write to: ./g4_texture_measures_envi/$site/
102 #site=$1
103 #season=$2

```

```

103 #do_ringdal=$3 #1=yes, 0=no
104 #do_texture=$4 #1=yes, 0=no
105 #do_convert_to_int=$5 #1=yes, 0=no
106 #do_output=$6 #1=yes, 0=no
    #quick=$7 # quick = 1 for do_convertst_to_int, all in one go. quick = 0 to run
107 manually step-by-step
108
109 if [ $do_texture_calc == 1 ];
110 then
111     #./texture_calc.sh
112     ./texture_calc_take2.sh $site dry 1 1 1 1 1
113     ./texture_calc_take2.sh $site wet 1 1 1 1 1
114 fi
115
116 # read from: ./g4_texture_measures_envi/$sitename
117 # write to: ./g4c_large_stack_resampled/$sitename
118 if [ $do_copy_to_largestack_folder == 1 ];
119 then
120     ./g3b_copy_to_largestack_folder.sh $site
121 fi
122
123 # read from: ./g4c_large_stack/$sitename
124
125 if [ $do_g3b_envi_virtualstack == 1 ];
126 then
127     ./g3b_envi_virtualstack.sh $site
128 fi
129
130 if [ $do_g3c_envi_realstacks == 1 ];
131 then
132     ./g3c_envi_realstacks.sh $site
133 fi
134
135 if [ $do_g3c_trainingtest_cutout == 1 ];
136 then
137     echo "cutting out training and test data to overlap area"
138     ./g3c_realstacks_trainingtest_cutout.sh $site 0 1
139 fi
140
141 #g3a_tif_to_envi.sh
142 #g3a_tif_to_envi_DRC_specific.sh
143
144     ##g3b_envi_virtualstack.sh
145
146 #g3c_envi_realstack.sh
147 #texture_calc.sh
148 #texture_calc_extra.sh

```



```
##run_texture_grass.py
```

*Script A2.16 f4b\_complete\_stack\_envi.sh*

```
1  #!/bin/bash
2  # bash script program to create a layer stack per site, of the
   overlapping area
3  # ideas taken from Tim Sutton's Linfiniti.com blog :
4  # http://linfiniti.com/2009/09/clipping-rasters-with-gdal-using-
   polygons/
5
6  # 1. create shpfile containing the features you wish to use as a clip
   mask - DONE
7  # 2. determine the extent of your clipmask
8  # 3. perform a rectangular clip on the image
9  # 4. perform a polygon clip on the image
10
11 #sitename=$1
12 #do_bbcclip=$2 #1=yes, 0=no
13 #do_shpclip=$3 #1=yes, 0=no
14
15
16 # step big-1: mv the .img files into separate sites
17
18 # ideas taken from Tim Sutton's Linfiniti.com blog :
19 # http://linfiniti.com/2009/09/clipping-rasters-with-gdal-using-
   polygons/
20
21 # Four steps to follow:
22
23 # 1. create shpfile containing the features you wish to use as a clip
   mask - DONE
24 # 2. determine the extent of your clipmask
25 # 3. perform a rectangular clip on the image
26 # 4. perform a polygon clip on the image
27
28 # 5. my added step: then combine these clipped rasters into a layer
   stack.
29
30
31 # 2. create shapefile extent
32 ### ==== Run these steps in loops over all the files in the folder,
33
```

```

### === then add the final gdalwarp step in the end to create a stack
34 for each site
35
36 sitename=$1
37 do_bbcclip=$2 #1=yes, 0=no
38 do_shpclip=$3 #1=yes, 0=no
39
40 cd ./f3_layerstack_overlap
41
42 SHPFILE=overlap_$1.shp
43 #SHPFILE=overlap_DRC4.shp
44 BASE=`basename $SHPFILE .shp`
45 EXTENT=`ogrinfo -so $SHPFILE $BASE | grep Extent \
46 | sed 's/Extent: //g' | sed 's/(//g' | sed 's/)//g' \
47 | sed 's/ - /, /g'`
48 EXTENT=`echo $EXTENT | awk -F ',' '{print $1 " " $4 " " $3 " " $2}'`
49 echo $sitename
50 echo $EXTENT
51
52 #Obtain Extent from Overlap shapefile
53 #cd ./f3_layerstack_overlap
54 #
55 #SHPFILE=overlap_$1.shp
56 #SHPFILE=overlap_DRC4.shp
57 #BASE=`basename $SHPFILE .shp`
58 #EXTENT=`ogrinfo -so $SHPFILE $BASE | grep Extent \
59 #| sed 's/Extent: //g' | sed 's/(//g' | sed 's/)//g' \
60 #| sed 's/ - /, /g'`
61 #EXTENT=`echo $EXTENT | awk -F ',' '{print $1 " " $4 " " $3 " " $2}'`
62 #echo $sitename
63 #echo $EXTENT
64
65 # DRC3 799999.960133 178576.920543 809620.027194 133059.987578
66 # DRC4 800040.027194 178422.457594 809620.027194 133059.987578
67
68 # Cameroon1 250955.000000 726170.000000 265250.000000
69 687000.000000
70 # Cameroon2 488360.000000 462455.000000 511350.000000
71 411620.000000
72 # Chad 608480.000000 1529840.000000 637230.000000
73 1495780.000000
74 # DRC 800040.027194 178422.457594 809620.027194
75 133059.987578
76 # Sudan 698415.000000 1458735.000000 714050.000000
77 1444050.000000
78 # Tanzania 392660.000000 9577095.000000 403980.000000
79 9532990.000000

```



```

112     echo 'bbclip processing: ' $RASTERFILE
        gdal_translate -projwin 488360.000000 462455.000000
511350.000000 411620.000000 -of ENVI $RASTERFILE
../../f4a_layerstack_pre_group/$sitename/`basename $RASTERFILE
113 .img`_bbclip.img
114     done
115     cd ../../
116 fi
117
118 if [ $do_bbclip == 1 ] && [ $sitename == "Chad" ]; then
119     echo "do_bbclip"
120     echo "$site"
121     cd ./f1_layerstack_pre/$sitename
122     #cd ./f1_layerstack_pre/DRC_Texture
123     for RASTERFILE in *.img;
124     do
125         echo 'bbclip processing: ' $RASTERFILE
            #gdal_translate -projwin 608480.000000 1529840.000000
637230.000000 1495780.000000 -of ENVI $RASTERFILE
            ../../f4a_layerstack_pre_group/$sitename/`basename $RASTERFILE
126 .img`_bbclip.img
            gdal_translate -projwin 614183.754502 1518376.961734
634162.966218 1496430.000000 -of ENVI $RASTERFILE
            ../../f4a_layerstack_pre_group/$sitename/`basename $RASTERFILE
127 .img`_bbclip.img #bspies 20150225 outline with TSX
128     done
129     cd ../../
130 fi
131
132 if [ $do_bbclip == 1 ] && [ $sitename == "DRC" ]; then
133     echo "do_bbclip"
134     echo "$site"
135     cd ./f1_layerstack_pre/$sitename
136     #cd ./f1_layerstack_pre/DRC_Texture
137     for RASTERFILE in *.img;
138     do
139         echo 'bbclip processing: ' $RASTERFILE
            #gdal_translate -projwin 800040.027194 178422.457594
809620.027194 133059.987578 -of ENVI $RASTERFILE
            ../../f4a_layerstack_pre_group/$sitename/`basename $RASTERFILE
140 .img`_bbclip.img
            #gdal_translate -projwin 799835.000000 181065.000000
805000.000000 167245.000000 -of ENVI $RASTERFILE
            ../../f4a_layerstack_pre_group/$sitename/`basename $RASTERFILE
141 .img`_bbclip.img #bspies 20150225 outline with TSX
            gdal_translate -projwin 799835.000000 181065.000000
819600.459909 133060.000000 -of ENVI $RASTERFILE

```

```

142 ../f4a_layerstack_pre_group/$sitename/`basename $RASTERFILE
143 .img`_bbclip.img #bspies 20150225 outline with TSX
144     done
145     cd ../../
146 fi
147 if [ $do_bbclip == 1 ] && [ $sitename == "Sudan" ]; then
148     echo "do_bbclip"
149     echo "$site"
150     cd ./f1_layerstack_pre/$sitename
151     #cd ./f1_layerstack_pre/DRC_Texture
152     for RASTERFILE in *.img;
153     do
154         echo 'bbclip processing: ' $RASTERFILE
155
156         #gdal_translate -projwin 698415.000000 1458735.000000
157         714050.000000 1444050.000000 -of ENVI $RASTERFILE
158         ../f4a_layerstack_pre_group/$sitename/`basename $RASTERFILE
159         .img`_bbclip.img
160         gdal_translate -projwin 698315.000000 1456355.125502
161         713572.614335 1444010.000000 -of ENVI $RASTERFILE
162         ../f4a_layerstack_pre_group/$sitename/`basename $RASTERFILE
163         .img`_bbclip.img #bspies 20150225 outline with TSX
164     done
165     cd ../../
166 fi
167 if [ $do_bbclip == 1 ] && [ $sitename == "Tanzania" ]; then
168     echo "do_bbclip"
169     echo "$site"
170     cd ./f1_layerstack_pre/$sitename
171     #cd ./f1_layerstack_pre/DRC_Texture
172     for RASTERFILE in *.img;
173     do
174         echo 'bbclip processing: ' $RASTERFILE
175         # gdal_translate -projwin 392660.000000 9577095.000000
176         403980.000000 9532990.000000 -of ENVI $RASTERFILE
177         ../f4a_layerstack_pre_group/$sitename/`basename $RASTERFILE
178         .img`_bbclip.img
179         gdal_translate -projwin 390720.000000 9577710.000000
180         403890.000000 9532185.000000 -of ENVI $RASTERFILE
181         ../f4a_layerstack_pre_group/$sitename/`basename $RASTERFILE
182         .img`_bbclip.img
183     done
184     cd ../../
185 fi

```

```

176 ##cd ./f1_layerstack_pre/$sitename
177 #create a for loop for all the .img files
178 #RASTERFILE=$2
179
180 ## currently these next three lines are done on ALICE:
181 # echo "cutting to extent"
182 #cd ../../
183
184 # 4. perform polygon clip on the image
185 #for file in *_bbclip.img;
186 if [ $do_shpclip == 1 ]; then
187     echo 'do_shpclip'
188     echo ./f4a_layerstack_pre_group/$sitename
189     cd ./f4a_layerstack_pre_group/$sitename
190
191     for RASTERFILE in *_bbclip.img;
192     #for RASTERFILE in
193     Cameroon1_Dry_ASAR_20100411_SLC_0_VV_VH_VH_2x4_15m_geocod
194     e_resampled_bbclip.img;
195     #for RASTERFILE in sigma0_hh_db_output_bbclip.img;
196     do
197         #.tif version
198         #gdalwarp -co COMPRESS=DEFLATE -co TILED=YES -of ENVI -r near -
199         cutline ../../f3_layerstack_overlap/$SHPPFILE -srcnodata 0 -dstnodata 0
200         `basename $RASTERFILE _bbclip.img`_bbclip.img
201         ../../f4b_layerstack_pre_group_shpclip/$sitename/`basename
202         $RASTERFILE _bbclip.img`_shpclip.img
203         echo 'shpclip processing: ' $RASTERFILE
204         #.envi version
205         gdalwarp -of ENVI -r near -cutline
206         ../../f3_layerstack_overlap/$SHPPFILE -srcnodata 0 -dstnodata 0
207         `basename $RASTERFILE _bbclip.img`_bbclip.img
208         ../../f4b_layerstack_pre_group_shpclip/$sitename/`basename
209         $RASTERFILE _bbclip.img`_shpclip.img
210         done
211     fi
212
213     #echo 'made this line1'
214     ##echo
215     ##echo "SHPPFILE: $SHPPFILE"
216     ##echo "RASTERFILE: $RASTERFILE"
217     #echo
218
219     ##echo `basename $RASTERFILE .img`_bbclip.img
220     ##echo `basename $RASTERFILE .img`_shpclip.img
221
222

```

```

213 ##gdalwarp -of ENVI -r lanczos -cutline
214 ../f3_layerstack_overlap/$SHPPFILE /tmp/`basename $RASTERFILE
215 .img`_bbclip.img ../g_layerstack/`basename $RASTERFILE
216 .img`_shpclip.img
217 ##echo 'made this line2'

218 # # step big-2: create layer stack of overlap between the tif files, using
219 the shapefile intersect from step f3
220 #
221 # #for file in ./d_Separate_tif_test/*
222 # #do
223 # # gdal_merge.py -o ./e_layerstack/layerstack3files.img -of ENVI -ps 5
224 5 -tap -#separate -n 0 a_nodata 0
225 ##done
226 #=====#
227 # used in presentation:
228 #if [ $do_shpclip == 1 ]; then
229 # echo 'do_shpclip'
230 # cd ../f4a_layerstack_pre_group/$sitename
231 #
232 # for RASTERFILE in *_bbclip.img;
233 # do
234 #     gdalwarp -of ENVI -r near -cutline
235 #     ../f3_layerstack_overlap/$SHPPFILE -srcnodata 0 -dstnodata 0
236 #     `basename $RASTERFILE _bbclip.img`_bbclip.img
237 #     ../f4b_layerstack_pre_group_shpclip/$sitename/`basename
238 $RASTERFILE _bbclip.img`_shpclip.img
239 # done
240 #fi

```

*Script A2.17 g2b2\_bandcalc\_ENVI.sh*

```

1 #!/bin/bash
2 # program to use gdal_calc.py to do band ratios and calculations
3 sitename=$1
4 convert_TSX=$2
5 linear_to_db=$3
6 other_calculations=$4
7 geocodeGS=$5 #geocodeG or geocodeS
8 echo $1
9 echo $2
10 echo $3
11 echo $4
12 echo $5
13

```

```

14 #cd g1a2_layerstack_utm_db_NA/$sitename/
15 #cd g1a1_layerstack_utm_db/$sitename/
16 cd f4b_layerstack_pre_group_shpclip/$sitename/
17
18 if [ ! -d "./process/" ]
19 then
20     mkdir process
21 fi
22
23 if [ ! -d "./process_geocodeG/" ]
24 then
25     mkdir process_geocodeG
26 fi
27
28 if [ ! -d "./process_geocodeS/" ]
29 then
30     mkdir process_geocodeS
31 fi
32
33 if [ ! -d "./tsx_dB_extra/" ]
34 then
35     mkdir tsx_dB_extra
36 fi
37
38 if [[ $sitename == "Chad" && $convert_TSX == 1 ]];
39 then
40     mv *TSX*.img ./tsx_dB_extra/
41     mv *TSX*.hdr ./tsx_dB_extra/
42     mv *TSX*.img.aux.xml ./tsx_dB_extra/
43
44     cd ./tsx_dB_extra/
45     for file in *.img;
46     do
47         echo " processing `basename $file .img`_dB.img"
48         gdal_calc.py -A $file --outfile=../"`basename $file
         _shpclip.img`_linear_shpclip.img" --calc="exp((A/10))" --format='ENVI' -
         -NoDataValue=0 --overwrite
49     done
50     cd ..
51 fi
52
53 if [[ $sitename == "DRC" && $convert_TSX == 1 ]];
54 then
55     mv *TSX*.img ./tsx_dB_extra/
56     mv *TSX*.hdr ./tsx_dB_extra/
57     mv *TSX*.img.aux.xml ./tsx_dB_extra/
58

```



```

59 cd ./tsx_dB_extra/
60 for file in *.img;
61 do
62     echo " processing `basename $file .img`_dB.img"
63     gdal_calc.py -A $file --outfile=../"`basename $file
        _shpclip.img`_linear_shpclip.img" --calc="exp((A/10))" --format='ENVI' -
        -NoDataValue=0 --overwrite
64 done
65 cd ..
66 fi
67
68 if [[ $sitename == "Sudan" && $convert_TSX == 1 ]];
69 then
70     mv *TSX*.img ./tsx_dB_extra/
71     mv *TSX*.hdr ./tsx_dB_extra/
72     mv *TSX*.img.aux.xml ./tsx_dB_extra/
73     cd ./tsx_dB_extra/
74
75     for file in *.img;
76     do
77         echo " processing `basename $file .img`_dB.img"
78         gdal_calc.py -A $file --outfile=../"`basename $file
            _shpclip.img`_linear_shpclip.img" --calc="exp((A/10))" --format='ENVI' -
            -NoDataValue=0 --overwrite
79     done
80     cd ..
81 fi
82
83
84 if [ ! -f ./${site}/process_geocodeG/*_geocodeG*.img ];
85 #if [ -f *_geocodeG*.img ];
86 then
87     echo "moving geocodeG files"
88     mv *_geocodeG*.img ./process_geocodeG/
89     mv *_geocodeG*.hdr ./process_geocodeG/
90     mv *_geocodeG*.img.aux.xml ./process_geocodeG/
91 fi
92
93 if [ ! -f ./${site}/process_geocodeS/*_geocodeS*.img ];
94 #if [ -f "*_geocodeS*.img" ];
95 then
96     echo "moving geocodeS files"
97     mv *_geocodeS*.img ./process_geocodeS/
98     mv *_geocodeS*.hdr ./process_geocodeS/
99     mv *_geocodeS*.img.aux.xml ./process_geocodeS/
100 fi
101

```

```

102 #linear to db conversion - geocodeS
103 #Cameroon1
104 if [[ $linear_to_db == 1 && "$geocodeGS" == "S" ]];
105 then
106     cd ./process_geocodeS
107     for file in *.img;
108     do
109         echo " processing `basename $file .img`_dB.img"
110         gdal_calc.py -A $file --outfile=./process/"`basename $file
111 .img`_dB.img" --calc="10*log10(A)" --format='ENVI' --NoDataValue=0 --
overwrite
112     done
113     echo "moving files to main folder:"
114     mv ../process/* ../../g1a1_layerstack_utm_db/$sitename
115     cd ..
116 fi
117
118 #linear to db conversion - geocodeG
119 #Cameroon1
120 if [[ $linear_to_db == 1 && "$geocodeGS" == "G" ]];
121 then
122     cd ./process_geocodeG
123     for file in *.img;
124     do
125         echo " processing `basename $file .img`_dB.img"
126         gdal_calc.py -A $file --outfile=./process/"`basename $file
127 .img`_dB.img" --calc="10*log10(A)" --format='ENVI' --NoDataValue=0 --
overwrite
128     done
129     echo "moving files to main folder:"
130     mv ../process/* ../../g1a1_layerstack_utm_db/$sitename
131     cd ..
132 fi
133
134 #other calculations - geocodeS
135 if [[ $other_calculations == 1 && "$geocodeGS" == "S" ]];
136 then
137     cd ./process_geocodeS
138
139     for file1 in *.img;
140     do
141         for file2 in *.img;
142         do
143             ##=====check1 for file1 =====#
144             if [ "$sitename" == "Cameroon1" ]
145             then
146                 #echo ""

```

```

146     #echo $sitename
147     #echo ""
148     if [[ "${file1:14:3}" == "ASA" ]]
149     then
150         #echo "file1 for ASAR: $file1"
151         #echo "check1 for ASAR: ${file1:9:19}"
152         check1=${file1:9:19}
153     elif [[ "${file1:14:3}" == "PAL" ]]
154     then
155         #echo "file1 for PALSAR: $file1"
156         #echo "check1 for PALSAR: ${file1:9:21}"
157         check1=${file1:9:21}
158     elif [[ "${file1:14:3}" == "TSX" ]]
159     then
160         #echo "file1 for TSX: $file1"
161         #echo "check1 for TSX: ${file1:9:18}"
162         check1=${file1:9:18}
163     else
164         echo "something is not right: $sitename"
165     fi
166 fi
167
168 if [ "$sitename" == "Cameroon2" ]
169 then
170     #echo ""
171     #echo $sitename
172     #echo ""
173     if [[ "${file1:14:3}" == "ASA" ]]
174     then
175         #echo "file1 for ASAR: $file1"
176         #echo "check1 for ASAR: ${file1:9:19}"
177         check1=${file1:9:19}
178     elif [[ "${file1:14:3}" == "PAL" ]]
179     then
180         #echo "file1 for PALSAR: $file1"
181         #echo "check1 for PALSAR: ${file1:9:21}"
182         check1=${file1:9:21}
183     elif [[ "${file1:14:3}" == "TSX" ]]
184     then
185         #echo "file1 for TSX: $file1"
186         #echo "check1 for TSX: ${file1:9:18}"
187         check1=${file1:9:18}
188     else
189         echo "something is not right: $sitename"
190     fi
191 fi
192

```

```

193     if [ "$sitename" == "Chad" ]
194     then
195         #echo ""
196         #echo $sitename
197         #echo ""
198         if [[ "${file1:9:3}" == "ASA" ]]
199         then
200             #echo "file1 for ASAR: $file1"
201             #echo "check1 for ASAR: ${file1:4:19}"
202             check1=${file1:4:19}
203         elif [[ "${file1:9:3}" == "PAL" ]]
204         then
205             #echo "file1 for PALSAR: $file1"
206             #echo "check1 for PALSAR: ${file1:4:21}"
207             check1=${file1:4:21}
208         elif [[ "${file1:9:3}" == "TSX" ]]
209         then
210             #echo "file1 for TSX: $file1"
211             #echo "check1 for TSX: ${file1:4:18}"
212             check1=${file1:4:18}
213         else
214             echo "something is not right: $sitename"
215         fi
216     fi
217
218     if [ "$sitename" == "DRC" ]
219     then
220         #echo ""
221         #echo $sitename
222         #echo ""
223         if [[ "${file1:8:3}" == "ASA" ]]
224         then
225             #echo "file1 for ASAR: $file1"
226             #echo "check1 for ASAR: ${file1:3:19}"
227             check1=${file1:3:19}
228         elif [[ "${file1:8:3}" == "PAL" ]]
229         then
230             #echo "file1 for PALSAR: $file1"
231             #echo "check1 for PALSAR: ${file1:3:21}"
232             check1=${file1:3:21}
233         elif [[ "${file1:8:3}" == "TSX" ]]
234         then
235             #echo "file1 for TSX: $file1"
236             #echo "check1 for TSX: ${file1:3:18}"
237             check1=${file1:3:18}
238         else
239             echo "something is not right: $sitename"

```

```

240     fi
241 fi
242
243 if [ "$sitename" == "Sudan" ]
244 then
245     #echo ""
246     #echo $sitename
247     #echo ""
248     if [[ "${file1:10:3}" == "ASA" ]]
249     then
250         #echo "file1 for ASAR: $file1"
251         #echo "check1 for ASAR: ${file1:5:19}"
252         check1=${file1:5:19}
253     elif [[ "${file1:10:3}" == "PAL" ]]
254     then
255         #echo "file1 for PALSAR: $file1"
256         #echo "check1 for PALSAR: ${file1:5:21}"
257         check1=${file1:5:21}
258     elif [[ "${file1:10:3}" == "TSX" ]]
259     then
260         #echo "file1 for TSX: $file1"
261         #echo "check1 for TSX: ${file1:5:18}"
262         check1=${file1:5:18}
263     else
264         echo "something is not right: $sitename"
265     fi
266 fi
267
268 if [ "$sitename" == "Tanzania" ]
269 then
270     #echo ""
271     #echo $sitename
272     #echo ""
273     if [[ "${file1:13:3}" == "ASA" ]]
274     then
275         #echo "file1 for ASAR: $file1"
276         #echo "check1 for ASAR: ${file1:8:19}"
277         check1=${file1:8:19}
278     elif [[ "${file1:13:3}" == "PAL" ]]
279     then
280         #echo "file1 for PALSAR: $file1"
281         #echo "check1 for PALSAR: ${file1:8:21}"
282         check1=${file1:8:21}
283     elif [[ "${file1:13:3}" == "TSX" ]]
284     then
285         #echo "file1 for TSX: $file1"
286         #echo "check1 for TSX: ${file1:8:18}"

```

```

287         check1=${file1:8:18}
288     else
289         echo "something is not right: $sitename"
290     fi
291 fi
292
293 ##=====check2 for file2 =====#
294 if [ "$sitename" == "Cameroon1" ]
295 then
296     #echo ""
297     #echo $sitename
298     #echo ""
299     if [[ "${file2:14:3}" == "ASA" ]]
300     then
301         #echo "file2 for ASAR: $file2"
302         #echo "check2 for ASAR: ${file2:9:19}"
303         check2=${file2:9:19}
304     elif [[ "${file2:14:3}" == "PAL" ]]
305     then
306         #echo "file2 for PALSAR: $file2"
307         #echo "check2 for PALSAR: ${file2:9:21}"
308         check2=${file2:9:21}
309     elif [[ "${file2:14:3}" == "TSX" ]]
310     then
311         #echo "file2 for TSX: $file2"
312         #echo "check2 for TSX: ${file2:9:18}"
313         check2=${file2:9:18}
314     else
315         echo "something is not right: $sitename"
316     fi
317 fi
318
319 if [ "$sitename" == "Cameroon2" ]
320 then
321     #echo ""
322     #echo $sitename
323     #echo ""
324     if [[ "${file2:14:3}" == "ASA" ]]
325     then
326         #echo "file2 for ASAR: $file2"
327         #echo "check2 for ASAR: ${file2:9:19}"
328         check2=${file2:9:19}
329     elif [[ "${file2:14:3}" == "PAL" ]]
330     then
331         #echo "file2 for PALSAR: $file2"
332         #echo "check2 for PALSAR: ${file2:9:21}"
333         check2=${file2:9:21}

```

```

334     elif [[ "${file2:14:3}" == "TSX" ]]
335     then
336         #echo "file2 for TSX: $file2"
337         #echo "check2 for TSX: ${file2:9:18}"
338         check2=${file2:9:18}
339     else
340         echo "something is not right: $sitename"
341     fi
342 fi
343
344 if [ "$sitename" == "Chad" ]
345 then
346     #echo ""
347     #echo $sitename
348     #echo ""
349     if [[ "${file2:9:3}" == "ASA" ]]
350     then
351         #echo "file2 for ASAR: $file2"
352         #echo "check2 for ASAR: ${file2:4:19}"
353         check2=${file2:4:19}
354     elif [[ "${file2:9:3}" == "PAL" ]]
355     then
356         #echo "file2 for PALSAR: $file2"
357         #echo "check2 for PALSAR: ${file2:4:21}"
358         check2=${file2:4:21}
359     elif [[ "${file2:9:3}" == "TSX" ]]
360     then
361         #echo "file2 for TSX: $file2"
362         #echo "check2 for TSX: ${file2:4:18}"
363         check2=${file2:4:18}
364     else
365         echo "something is not right: $sitename"
366     fi
367 fi
368
369 if [ "$sitename" == "DRC" ]
370 then
371     #echo ""
372     #echo $sitename
373     #echo ""
374     if [[ "${file2:8:3}" == "ASA" ]]
375     then
376         #echo "file2 for ASAR: $file2"
377         #echo "check2 for ASAR: ${file2:3:19}"
378         check2=${file2:3:19}
379     elif [[ "${file2:8:3}" == "PAL" ]]
380     then

```

```

381         #echo "file2 for PALSAR: $file2"
382         #echo "check2 for PALSAR: ${file2:3:21}"
383         check2=${file2:3:21}
384     elif [[ "${file2:8:3}" == "TSX" ]]
385     then
386         #echo "file2 for TSX: $file2"
387         #echo "check2 for TSX: ${file2:3:18}"
388         check2=${file2:3:18}
389     else
390         echo "something is not right: $sitename"
391     fi
392 fi
393
394 if [ "$sitename" == "Sudan" ]
395 then
396     #echo ""
397     #echo $sitename
398     #echo ""
399     if [[ "${file2:10:3}" == "ASA" ]]
400     then
401         #echo "file2 for ASAR: $file2"
402         #echo "check2 for ASAR: ${file2:5:19}"
403         check2=${file2:5:19}
404     elif [[ "${file2:10:3}" == "PAL" ]]
405     then
406         #echo "file2 for PALSAR: $file2"
407         #echo "check2 for PALSAR: ${file2:5:21}"
408         check2=${file2:5:21}
409     elif [[ "${file2:10:3}" == "TSX" ]]
410     then
411         #echo "file2 for TSX: $file2"
412         #echo "check2 for TSX: ${file2:5:18}"
413         check2=${file2:5:18}
414     else
415         echo "something is not right: $sitename"
416     fi
417 fi
418
419 if [ "$sitename" == "Tanzania" ]
420 then
421     #echo ""
422     #echo $sitename
423     #echo ""
424     if [[ "${file2:13:3}" == "ASA" ]]
425     then
426         #echo "file2 for ASAR: $file2"
427         #echo "check2 for ASAR: ${file2:8:19}"

```



```

428         check2=${file2:8:19}
429         elif [[ "${file2:13:3}" == "PAL" ]]
430         then
431             #echo "file2 for PALSAR: $file2"
432             #echo "check2 for PALSAR: ${file2:8:21}"
433             check2=${file2:8:21}
434         elif [[ "${file2:13:3}" == "TSX" ]]
435         then
436             #echo "file2 for TSX: $file2"
437             #echo "check2 for TSX: ${file2:8:18}"
438             check2=${file2:8:18}
439         else
440             echo "something is not right: $sitename"
441         fi
442     fi
443
444
445     # HHVV - DualPol
446     if [[ "$file1" == *HHVV_HH_* ]] && [[ "$file2" == *HHVV_VV_* ]]
447 && [[ "$check1" == "$check2" ]]
448     then
449         echo "HHVV_HH:    $file1"
450         echo "HHVV_VV:    $file2"
451         echo "check1:    $check1" #check images are the same date
452         echo "check2:    $check2" #check images are the same date
453
454         echo " processing `basename $file1 .img`_HHdivVV_linear.img"
455         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
456 $file1 .img`_HHdivVV_linear.img" --calc="A/B" --NoDataValue=0 --
457 format='ENVI' --overwrite
458         echo " processing `basename $file1 .img`_HHminVV_linear.img"
459         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
460 $file1 .img`_HHminVV_linear.img" --calc="A-B" --NoDataValue=0 --
461 format='ENVI' --overwrite
462         echo " processing `basename $file1 .img`_HHplusVV_linear.img"
463         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
464 $file1 .img`_HHplusVV_linear.img" --calc="A+B" --NoDataValue=0 --
465 format='ENVI' --overwrite
466         echo " processing `basename $file1 .img`_VVdivHH_rdB.img"
467         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
468 $file1 .img`_VVdivHH_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
469 NoDataValue=0 --format='ENVI' --overwrite
470         echo " processing `basename $file1
471 .img`_VVdivHH_copol_gamma.img"
472         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
473 $file1 .img`_VVdivHH_copol_gamma.img" --

```

```

464 calc="(10*log10(B))/(10*log10(A))" --NoDataValue=0 --format='ENVI' --
    overwrite
465     echo " processing `basename $file1
    .img`_VVdivHHpVV_depol_delta.img"
466     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
    $file1 .img`_VVdivHHpVV_depol_delta.img" --
    calc="(10*log10(B))/((10*log10(A))+(10*log10(B)))" --NoDataValue=0 --
    format='ENVI' --overwrite
467     echo " processing `basename $file1
468     .img`_HHmVVdivHHpVV_NDI.img"
    gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
469 $file1 .img`_HHmVVdivHHpVV_NDI.img" --calc="(A-B)/(A+B)" --
    NoDataValue=0 --format='ENVI' --overwrite
    fi
470
471     # VVVH - DualPol
472     if [[ "$file1" == *VVVH_VV_* ]] && [[ "$file2" == *VVVH_VH_* ]]
473 && [[ "$check1" == "$check2" ]]
    then
474         echo "VVVH_VV:    $file1"
475         echo "VVVH_VH:    $file2"
476         echo "check1:     $check1"
477         echo "check2:     $check2"
478         #echo "file1 string 1to25: ${file1:0:25}"
479         #echo "file2 string 1to25: ${file2:0:25}"
480
481         echo " processing `basename $file1 .img`_VHdivVV_linear.img"
482         gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
    $file1 .img`_VHdivVV_linear.img" --calc="B/A" --NoDataValue=0 --
    format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_VVminVH_linear.img"
483         gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
    $file1 .img`_VVminVH_linear.img" --calc="A-B" --NoDataValue=0 --
484 format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_VVplusVH_linear.img"
485         gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
    $file1 .img`_VVplusVH_linear.img" --calc="A+B" --NoDataValue=0 --
486 format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_VHdivVV_rdB.img"
487         gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
    $file1 .img`_VHdivVV_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
488 NoDataValue=0 --format='ENVI' --overwrite
        echo " processing `basename $file1
489 .img`_VVmVHdivVVpVH_NDI.img"
        gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
490 $file1 .img`_VVmVHdivVVpVH_NDI.img" --calc="(A-B)/(A+B)" --
    NoDataValue=0 --format='ENVI' --overwrite

```

```

fi
491
492 #HHHV- DualPol
493 if [[ "$file1" == *HHHV_HH_* ]] && [[ "$file2" == *HHHV_HV_* ]]
494 && [[ "$check1" == "$check2" ]]
    then
495     echo "HHHV_HH:    $file1"
496     echo "HHHV_HV:    $file2"
497     echo "check1:     $check1"
498     echo "check2:     $check2"
499     #echo "file1 string 1to25: ${file1:0:25}"
500     #echo "file2 string 1to25: ${file2:0:25}"
501
502     echo " processing `basename $file1 .img`_HVdivHH_linear.img"
503     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
504 $file1 .img`_HVdivHH_linear.img" --calc="B/A" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHminHV_linear.img"
505     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
506 $file1 .img`_HHminHV_linear.img" --calc="A-B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHplusHV_linear.img"
507     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
508 $file1 .img`_HHplusHV_linear.img" --calc="A+B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HVdivHH_rdB.img"
509     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
510 $file1 .img`_HVdivHH_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
NoDataValue=0 --format='ENVI' --overwrite
        echo " processing `basename $file1
511 .img`_HHmHVdivHHpHV_NDI.img"
512     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
513 $file1 .img`_HHmHVdivHHpHV_NDI.img" --calc="(A-B)/(A+B)" --
NoDataValue=0 --format='ENVI' --overwrite
    fi
514
515 #HHHV - QuadPol
516 if [[ "$file1" == *Quad_HH_* ]] && [[ "$file2" == *Quad_HV_* ]]
517 && [[ "$check1" == "$check2" ]]
    then
518     echo "Quad_HH:    $file1"
519     echo "Quad_HV:    $file2"
520     echo "check1:     $check1"
521     echo "check2:     $check2"
522     #echo "file1 string 1to25: ${file1:0:25}"
523     #echo "file2 string 1to25: ${file2:0:25}"
524

```

```

525     echo " processing `basename $file1 .img`_HVdivHH_linear.img"
526     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
527 $file1 .img`_HVdivHH_linear.img" --calc="B/A" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHminHV_linear.img"
528     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
529 $file1 .img`_HHminHV_linear.img" --calc="A-B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHplusHV_linear.img"
530     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
531 $file1 .img`_HHplusHV_linear.img" --calc="A+B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HVdivHH_rdB.img"
532     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
533 $file1 .img`_HVdivHH_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
NoDataValue=0 --format='ENVI' --overwrite
        echo " processing `basename $file1
534 .img`_HHmHVdivHHpHV_NDI.img"
        gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
535 $file1 .img`_HHmHVdivHHpHV_NDI.img" --calc="(A-B)/(A+B)" --
NoDataValue=0 --format='ENVI' --overwrite
        fi

536
537     #VVVH - QuadPol
538     if [[ "$file1" == *Quad_VV_* ]] && [[ "$file2" == *Quad_VH_* ]]
539 && [[ "$check1" == "$check2" ]]
        then
540         echo "Quad_VV:    $file1"
541         echo "Quad_VH:    $file2"
542         echo "check1:    $check1"
543         echo "check2:    $check2"
544         #echo "file1 string 1to25: ${file1:0:25}"
545         #echo "file2 string 1to25: ${file2:0:25}"
546
547         echo " processing `basename $file1 .img`_VHdivVV_linear.img"
548         gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
549 $file1 .img`_VVdivVH_linear.img" --calc="B/A" --NoDataValue=0 --
format='ENVI' --overwrite
                echo " processing `basename $file1 .img`_VVminVH_linear.img"
550         gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
551 $file1 .img`_VVminVH_linear.img" --calc="A-B" --NoDataValue=0 --
format='ENVI' --overwrite
                echo " processing `basename $file1 .img`_VVplusVH_linear.img"
552         gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
553 $file1 .img`_VVplusVH_linear.img" --calc="A+B" --NoDataValue=0 --
format='ENVI' --overwrite
                echo " processing `basename $file1 .img`_VHdivVV_rdB.img"

```

```

554     gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
555 $file1 .img`_VHdivVV_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
NoDataValue=0 --format='ENVI' --overwrite
    echo " processing `basename $file1
556 .img`_VVmVHdivVVpVH_NDI.img"
    gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
557 $file1 .img`_VVmVHdivVVpVH_NDI.img" --calc="(A-B)/(A+B)" --
NoDataValue=0 --format='ENVI' --overwrite
    fi

558
559     #HHVV - QuadPol
560     if [[ "$file1" == *Quad_HH_* ]] && [[ "$file2" == *Quad_VV_* ]]
561 && [[ "$check1" == "$check2" ]]
    then
562         echo "Quad_HH:    $file1"
563         echo "Quad_VV:    $file2"
564         echo "check1:     $check1"
565         echo "check2:     $check2"
566         #echo "file1 string 1to25: ${file1:0:25}"
567         #echo "file2 string 1to25: ${file2:0:25}"
568
569         echo " processing `basename $file1 .img`_HHdivVV_linear.img"
570         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
571 $file1 .img`_HHdivVV_linear.img" --calc="A/B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHminVV_linear.img"
572         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
573 $file1 .img`_HHminVV_linear.img" --calc="A-B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHplusVV_linear.img"
574         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
575 $file1 .img`_HHplusVV_linear.img" --calc="A+B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_VVdivHH_rdB.img"
576         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
577 $file1 .img`_VVdivHH_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
NoDataValue=0 --format='ENVI' --overwrite
        echo " processing `basename $file1
578 .img`_VVdivHH_copol_gamma.img"
        gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
579 $file1 .img`_VVdivHH_copol_gamma.img" --
calc="(10*log10(B))/(10*log10(A))" --NoDataValue=0 --format='ENVI' --
overwrite
        echo " processing `basename $file1
580 .img`_VVdivHHpVV_depol_delta.img"
        gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
581 $file1 .img`_VVdivHHpVV_depol_delta.img" --

```

```

calc="(10*log10(B))/((10*log10(A))+(10*log10(B)))" --NoDataValue=0 --
format='ENVI' --overwrite
    echo " processing `basename $file1
582 .img`_HHmVVdivHHpVV_NDI.img"
    gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
583 $file1 .img`_HHmVVdivHHpVV_NDI.img" --calc="(A-B)/(A+B)" --
NoDataValue=0 --format='ENVI' --overwrite
    fi
584 done #for file2
585 done #for file1
586
587 echo "moving files to main folder:"
588 #mv ../process/* ./
589 mv ../process/* ../../g1a1_layerstack_utm_db/$sitename
590 cd ..
591
592 fi # if other_calculations - geocodeS
593
594 #=====#
595 #other calculations - geocodeS
596 if [[ $other_calculations == 1 && "$geocodeGS" == "G" ]];
597 then
598     cd ./process_geocodeG
599     for file1 in *.img;
600     do
601         for file2 in *.img;
602         do
603             ##=====check1 for file1 =====#
604             if [ "$sitename" == "Cameroon1" ]
605             then
606                 #echo ""
607                 #echo $sitename
608                 #echo ""
609                 if [[ "${file1:14:3}" == "ASA" ]]
610                 then
611                     #echo "file1 for ASAR: $file1"
612                     #echo "check1 for ASAR: ${file1:9:19}"
613                     check1=${file1:9:19}
614                 elif [[ "${file1:14:3}" == "PAL" ]]
615                 then
616                     #echo "file1 for PALSAR: $file1"
617                     #echo "check1 for PALSAR: ${file1:9:21}"
618                     check1=${file1:9:21}
619                 elif [[ "${file1:14:3}" == "TSX" ]]
620                 then
621                     #echo "file1 for TSX: $file1"
622                     #echo "check1 for TSX: ${file1:9:18}"

```

```

623         check1=${file1:9:18}
624     else
625         echo "something is not right: $sitename"
626     fi
627 fi
628
629 if [ "$sitename" == "Cameroon2" ]
630 then
631     #echo ""
632     #echo $sitename
633     #echo ""
634     if [[ "${file1:14:3}" == "ASA" ]]
635     then
636         #echo "file1 for ASAR: $file1"
637         #echo "check1 for ASAR: ${file1:9:19}"
638         check1=${file1:9:19}
639     elif [[ "${file1:14:3}" == "PAL" ]]
640     then
641         #echo "file1 for PALSAR: $file1"
642         #echo "check1 for PALSAR: ${file1:9:21}"
643         check1=${file1:9:21}
644     elif [[ "${file1:14:3}" == "TSX" ]]
645     then
646         #echo "file1 for TSX: $file1"
647         #echo "check1 for TSX: ${file1:9:18}"
648         check1=${file1:9:18}
649     else
650         echo "something is not right: $sitename"
651     fi
652 fi
653
654 if [ "$sitename" == "Chad" ]
655 then
656     #echo ""
657     #echo $sitename
658     #echo ""
659     if [[ "${file1:9:3}" == "ASA" ]]
660     then
661         #echo "file1 for ASAR: $file1"
662         #echo "check1 for ASAR: ${file1:4:19}"
663         check1=${file1:4:19}
664     elif [[ "${file1:9:3}" == "PAL" ]]
665     then
666         #echo "file1 for PALSAR: $file1"
667         #echo "check1 for PALSAR: ${file1:4:21}"
668         check1=${file1:4:21}
669     elif [[ "${file1:9:3}" == "TSX" ]]

```

```

670     then
671         #echo "file1 for TSX: $file1"
672         #echo "check1 for TSX: ${file1:4:18}"
673         check1=${file1:4:18}
674     else
675         echo "something is not right: $sitename"
676     fi
677 fi
678
679 if [ "$sitename" == "DRC" ]
680 then
681     #echo ""
682     #echo $sitename
683     #echo ""
684     if [[ "${file1:8:3}" == "ASA" ]]
685     then
686         #echo "file1 for ASAR: $file1"
687         #echo "check1 for ASAR: ${file1:3:19}"
688         check1=${file1:3:19}
689     elif [[ "${file1:8:3}" == "PAL" ]]
690     then
691         #echo "file1 for PALSAR: $file1"
692         #echo "check1 for PALSAR: ${file1:3:21}"
693         check1=${file1:3:21}
694     elif [[ "${file1:8:3}" == "TSX" ]]
695     then
696         #echo "file1 for TSX: $file1"
697         #echo "check1 for TSX: ${file1:3:18}"
698         check1=${file1:3:18}
699     else
700         echo "something is not right: $sitename"
701     fi
702 fi
703
704 if [ "$sitename" == "Sudan" ]
705 then
706     #echo ""
707     #echo $sitename
708     #echo ""
709     if [[ "${file1:10:3}" == "ASA" ]]
710     then
711         #echo "file1 for ASAR: $file1"
712         #echo "check1 for ASAR: ${file1:5:19}"
713         check1=${file1:5:19}
714     elif [[ "${file1:10:3}" == "PAL" ]]
715     then
716         #echo "file1 for PALSAR: $file1"

```



```

717         #echo "check1 for PALSAR: ${file1:5:21}"
718         check1=${file1:5:21}
719         elif [[ "${file1:10:3}" == "TSX" ]]
720         then
721             #echo "file1 for TSX: $file1"
722             #echo "check1 for TSX: ${file1:5:18}"
723             check1=${file1:5:18}
724         else
725             echo "something is not right: $sitename"
726         fi
727     fi
728
729     if [ "$sitename" == "Tanzania" ]
730     then
731         #echo ""
732         #echo $sitename
733         #echo ""
734         if [[ "${file1:13:3}" == "ASA" ]]
735         then
736             #echo "file1 for ASAR: $file1"
737             #echo "check1 for ASAR: ${file1:8:19}"
738             check1=${file1:8:19}
739         elif [[ "${file1:13:3}" == "PAL" ]]
740         then
741             #echo "file1 for PALSAR: $file1"
742             #echo "check1 for PALSAR: ${file1:8:21}"
743             check1=${file1:8:21}
744         elif [[ "${file1:13:3}" == "TSX" ]]
745         then
746             #echo "file1 for TSX: $file1"
747             #echo "check1 for TSX: ${file1:8:18}"
748             check1=${file1:8:18}
749         else
750             echo "something is not right: $sitename"
751         fi
752     fi
753
754     ##=====check2 for file2 =====#
755     if [ "$sitename" == "Cameroon1" ]
756     then
757         #echo ""
758         #echo $sitename
759         #echo ""
760         if [[ "${file2:14:3}" == "ASA" ]]
761         then
762             #echo "file2 for ASAR: $file2"
763             #echo "check2 for ASAR: ${file2:9:19}"

```

```

764         check2=${file2:9:19}
765     elif [[ "${file2:14:3}" == "PAL" ]]
766     then
767         #echo "file2 for PALSAR: $file2"
768         #echo "check2 for PALSAR: ${file2:9:21}"
769         check2=${file2:9:21}
770     elif [[ "${file2:14:3}" == "TSX" ]]
771     then
772         #echo "file2 for TSX: $file2"
773         #echo "check2 for TSX: ${file2:9:18}"
774         check2=${file2:9:18}
775     else
776         echo "something is not right: $sitename"
777     fi
778 fi
779
780 if [ "$sitename" == "Cameroon2" ]
781 then
782     #echo ""
783     #echo $sitename
784     #echo ""
785     if [[ "${file2:14:3}" == "ASA" ]]
786     then
787         #echo "file2 for ASAR: $file2"
788         #echo "check2 for ASAR: ${file2:9:19}"
789         check2=${file2:9:19}
790     elif [[ "${file2:14:3}" == "PAL" ]]
791     then
792         #echo "file2 for PALSAR: $file2"
793         #echo "check2 for PALSAR: ${file2:9:21}"
794         check2=${file2:9:21}
795     elif [[ "${file2:14:3}" == "TSX" ]]
796     then
797         #echo "file2 for TSX: $file2"
798         #echo "check2 for TSX: ${file2:9:18}"
799         check2=${file2:9:18}
800     else
801         echo "something is not right: $sitename"
802     fi
803 fi
804
805 if [ "$sitename" == "Chad" ]
806 then
807     #echo ""
808     #echo $sitename
809     #echo ""
810     if [[ "${file2:9:3}" == "ASA" ]]

```

```

811     then
812         #echo "file2 for ASAR: $file2"
813         #echo "check2 for ASAR: ${file2:4:19}"
814         check2=${file2:4:19}
815     elif [[ "${file2:9:3}" == "PAL" ]]
816     then
817         #echo "file2 for PALSAR: $file2"
818         #echo "check2 for PALSAR: ${file2:4:21}"
819         check2=${file2:4:21}
820     elif [[ "${file2:9:3}" == "TSX" ]]
821     then
822         #echo "file2 for TSX: $file2"
823         #echo "check2 for TSX: ${file2:4:18}"
824         check2=${file2:4:18}
825     else
826         echo "something is not right: $sitename"
827     fi
828 fi
829
830 if [ "$sitename" == "DRC" ]
831 then
832     #echo ""
833     #echo $sitename
834     #echo ""
835     if [[ "${file2:8:3}" == "ASA" ]]
836     then
837         #echo "file2 for ASAR: $file2"
838         #echo "check2 for ASAR: ${file2:3:19}"
839         check2=${file2:3:19}
840     elif [[ "${file2:8:3}" == "PAL" ]]
841     then
842         #echo "file2 for PALSAR: $file2"
843         #echo "check2 for PALSAR: ${file2:3:21}"
844         check2=${file2:3:21}
845     elif [[ "${file2:8:3}" == "TSX" ]]
846     then
847         #echo "file2 for TSX: $file2"
848         #echo "check2 for TSX: ${file2:3:18}"
849         check2=${file2:3:18}
850     else
851         echo "something is not right: $sitename"
852     fi
853 fi
854
855 if [ "$sitename" == "Sudan" ]
856 then
857     #echo ""

```

```

858     #echo $sitename
859     #echo ""
860     if [[ "${file2:10:3}" == "ASA" ]]
861     then
862         #echo "file2 for ASAR: $file2"
863         #echo "check2 for ASAR: ${file2:5:19}"
864         check2=${file2:5:19}
865     elif [[ "${file2:10:3}" == "PAL" ]]
866     then
867         #echo "file2 for PALSAR: $file2"
868         #echo "check2 for PALSAR: ${file2:5:21}"
869         check2=${file2:5:21}
870     elif [[ "${file2:10:3}" == "TSX" ]]
871     then
872         #echo "file2 for TSX: $file2"
873         #echo "check2 for TSX: ${file2:5:18}"
874         check2=${file2:5:18}
875     else
876         echo "something is not right: $sitename"
877     fi
878 fi
879
880
881 if [ "$sitename" == "Tanzania" ]
882 then
883     #echo ""
884     #echo $sitename
885     #echo ""
886     if [[ "${file2:13:3}" == "ASA" ]]
887     then
888         #echo "file2 for ASAR: $file2"
889         #echo "check2 for ASAR: ${file2:8:19}"
890         check2=${file2:8:19}
891     elif [[ "${file2:13:3}" == "PAL" ]]
892     then
893         #echo "file2 for PALSAR: $file2"
894         #echo "check2 for PALSAR: ${file2:8:21}"
895         check2=${file2:8:21}
896     elif [[ "${file2:13:3}" == "TSX" ]]
897     then
898         #echo "file2 for TSX: $file2"
899         #echo "check2 for TSX: ${file2:8:18}"
900         check2=${file2:8:18}
901     else
902         echo "something is not right: $sitename"
903     fi
904 fi

```

```

905
906     # HHVV - DualPol
907     if [[ "$file1" == *HHVV_HH_* ]] && [[ "$file2" == *HHVV_VV_* ]]
908 && [[ "$check1" == "$check2" ]]
909     then
910         echo "HHVV_HH:    $file1"
911         echo "HHVV_VV:    $file2"
912         echo "check1:    $check1" #check images are the same date
913         echo "check2:    $check2" #check images are the same date
914         echo " processing `basename $file1 .img`_HHdivVV_linear.img"
915         gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
$file1 .img`_HHdivVV_linear.img" --calc="A/B" --NoDataValue=0 --
format='ENVI' --overwrite
916         echo " processing `basename $file1 .img`_HHminVV_linear.img"
gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
917 $file1 .img`_HHminVV_linear.img" --calc="A-B" --NoDataValue=0 --
format='ENVI' --overwrite
918         echo " processing `basename $file1 .img`_HHplusVV_linear.img"
gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
919 $file1 .img`_HHplusVV_linear.img" --calc="A+B" --NoDataValue=0 --
format='ENVI' --overwrite
920         echo " processing `basename $file1 .img`_VVdivHH_rdB.img"
gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
921 $file1 .img`_VVdivHH_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
NoDataValue=0 --format='ENVI' --overwrite
922         echo " processing `basename $file1
.img`_VVdivHH_copol_gamma.img"
gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
923 $file1 .img`_VVdivHH_copol_gamma.img" --
calc="(10*log10(B))/(10*log10(A))" --NoDataValue=0 --format='ENVI' --
overwrite
924         echo " processing `basename $file1
.img`_VVdivHHpVV_depol_delta.img"
gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
925 $file1 .img`_VVdivHHpVV_depol_delta.img" --
calc="(10*log10(B))/((10*log10(A))+(10*log10(B)))" --NoDataValue=0 --
format='ENVI' --overwrite
926         echo " processing `basename $file1
.img`_HHmVVdivHHpVV_NDI.img"
gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
$file1 .img`_HHmVVdivHHpVV_NDI.img" --calc="(A-B)/(A+B)" --
NoDataValue=0 --format='ENVI' --overwrite
927         fi
928     # VVVH - DualPol
929     if [[ "$file1" == *VVVH_VV_* ]] && [[ "$file2" == *VVVH_VH_* ]]
930 && [[ "$check1" == "$check2" ]]

```

```

then
931     echo "VVVH_VV:    $file1"
932     echo "VVVH_VH:    $file2"
933     echo "check1:     $check1"
934     echo "check2:     $check2"
935     #echo "file1 string 1to25: ${file1:0:25}"
936     #echo "file2 string 1to25: ${file2:0:25}"
937
938     echo " processing `basename $file1 .img`_VHdivVV_linear.img"
939     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
940 $file1 .img`_VHdivVV_linear.img" --calc="B/A" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_VVminVH_linear.img"
941     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
942 $file1 .img`_VVminVH_linear.img" --calc="A-B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_VVplusVH_linear.img"
943     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
944 $file1 .img`_VVplusVH_linear.img" --calc="A+B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_VHdivVV_rdB.img"
945     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
946 $file1 .img`_VHdivVV_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
NoDataValue=0 --format='ENVI' --overwrite
        echo " processing `basename $file1
947 .img`_VVMVHdivVVPVH_NDI.img"
        gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
948 $file1 .img`_VVMVHdivVVPVH_NDI.img" --calc="(A-B)/(A+B)" --
NoDataValue=0 --format='ENVI' --overwrite
        fi
949
950     #HHHV- DualPol
951     if [[ "$file1" == *HHHV_HH_* ]] && [[ "$file2" == *HHHV_HV_* ]]
952 && [[ "$check1" == "$check2" ]]
        then
953     echo "HHHV_HH:    $file1"
954     echo "HHHV_HV:    $file2"
955     echo "check1:     $check1"
956     echo "check2:     $check2"
957     #echo "file1 string 1to25: ${file1:0:25}"
958     #echo "file2 string 1to25: ${file2:0:25}"
959
960     echo " processing `basename $file1 .img`_HVdivHH_linear.img"
961     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
962 $file1 .img`_HVdivHH_linear.img" --calc="B/A" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHminHV_linear.img"

```

```

963     gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
964 $file1 .img`_HHminHV_linear.img" --calc="A-B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHplusHV_linear.img"
965     gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
966 $file1 .img`_HHplusHV_linear.img" --calc="A+B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HVdivHH_rdB.img"
967     gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
968 $file1 .img`_HVdivHH_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
NoDataValue=0 --format='ENVI' --overwrite
        echo " processing `basename $file1
969 .img`_HHmHVdivHHpHV_NDI.img"
        gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
970 $file1 .img`_HHmHVdivHHpHV_NDI.img" --calc="(A-B)/(A+B)" --
NoDataValue=0 --format='ENVI' --overwrite
        fi

971
972     #HHHV - QuadPol
973     if [[ "$file1" == *Quad_HH_* ]] && [[ "$file2" == *Quad_HV_* ]]
974 && [[ "$check1" == "$check2" ]]
        then
975         echo "Quad_HH:    $file1"
976         echo "Quad_HV:    $file2"
977         echo "check1:     $check1"
978         echo "check2:     $check2"
979         #echo "file1 string 1to25: ${file1:0:25}"
980         #echo "file2 string 1to25: ${file2:0:25}"
981
982         echo " processing `basename $file1 .img`_HVdivHH_linear.img"
983         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
984 $file1 .img`_HVdivHH_linear.img" --calc="B/A" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHminHV_linear.img"
985         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
986 $file1 .img`_HHminHV_linear.img" --calc="A-B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHplusHV_linear.img"
987         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
988 $file1 .img`_HHplusHV_linear.img" --calc="A+B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HVdivHH_rdB.img"
989         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
990 $file1 .img`_HVdivHH_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
NoDataValue=0 --format='ENVI' --overwrite
        echo " processing `basename $file1
991 .img`_HHmHVdivHHpHV_NDI.img"

```

```

992     gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
993 $file1 .img`_HHmHVdivHHpHV_NDI.img" --calc="(A-B)/(A+B)" --
    NoDataValue=0 --format='ENVI' --overwrite
    fi

994
995     #VVVH - QuadPol
996     if [[ "$file1" == *Quad_VV_* ]] && [[ "$file2" == *Quad_VH_* ]]
997 && [[ "$check1" == "$check2" ]]
    then
998         echo "Quad_VV:    $file1"
999         echo "Quad_VH:    $file2"
1000         echo "check1:    $check1"
1001         echo "check2:    $check2"
1002         #echo "file1 string 1to25: ${file1:0:25}"
1003         #echo "file2 string 1to25: ${file2:0:25}"
1004
1005         echo " processing `basename $file1 .img`_VHdivVV_linear.img"
1006         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
1007 $file1 .img`_VVdivVH_linear.img" --calc="B/A" --NoDataValue=0 --
    format='ENVI' --overwrite
            echo " processing `basename $file1 .img`_VVminVH_linear.img"
1008         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
1009 $file1 .img`_VVminVH_linear.img" --calc="A-B" --NoDataValue=0 --
    format='ENVI' --overwrite
            echo " processing `basename $file1 .img`_VVplusVH_linear.img"
1010         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
1011 $file1 .img`_VVplusVH_linear.img" --calc="A+B" --NoDataValue=0 --
    format='ENVI' --overwrite
            echo " processing `basename $file1 .img`_VHdivVV_rdB.img"
1012         gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
1013 $file1 .img`_VHdivVV_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
    NoDataValue=0 --format='ENVI' --overwrite
            echo " processing `basename $file1
1014 .img`_VVmVHdivVVpVH_NDI.img"
            gdal_calc.py -A $file1 -B $file2 --outfile=./process/"`basename
1015 $file1 .img`_VVmVHdivVVpVH_NDI.img" --calc="(A-B)/(A+B)" --
    NoDataValue=0 --format='ENVI' --overwrite
            fi

1016
1017     #HHVV - QuadPol
1018     if [[ "$file1" == *Quad_HH_* ]] && [[ "$file2" == *Quad_VV_* ]]
1019 && [[ "$check1" == "$check2" ]]
    then
1020         echo "Quad_HH:    $file1"
1021         echo "Quad_VV:    $file2"
1022         echo "check1:    $check1"
1023         echo "check2:    $check2"

```



```

1024     #echo "file1 string 1to25: ${file1:0:25}"
1025     #echo "file2 string 1to25: ${file2:0:25}"
1026
1027     echo " processing `basename $file1 .img`_HHdivVV_linear.img"
1028     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
1029 $file1 .img`_HHdivVV_linear.img" --calc="A/B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHminVV_linear.img"
1030     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
1031 $file1 .img`_HHminVV_linear.img" --calc="A-B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_HHplusVV_linear.img"
1032     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
1033 $file1 .img`_HHplusVV_linear.img" --calc="A+B" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1 .img`_VVdivHH_rdB.img"
1034     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
1035 $file1 .img`_VVdivHH_rdB.img" --calc="10*(log10((B*B)/(A*A)))" --
1036 NoDataValue=0 --format='ENVI' --overwrite
        echo " processing `basename $file1
.img`_VVdivHH_copol_gamma.img"
1037     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
$file1 .img`_VVdivHH_copol_gamma.img" --
1038 calc="(10*log10(B))/(10*log10(A))" --NoDataValue=0 --format='ENVI' --
overwrite
        echo " processing `basename $file1
.img`_VVdivHHpVV_depol_delta.img"
1039     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
$file1 .img`_VVdivHHpVV_depol_delta.img" --
1040 calc="(10*log10(B))/((10*log10(A))+(10*log10(B)))" --NoDataValue=0 --
format='ENVI' --overwrite
        echo " processing `basename $file1
.img`_HHmVVdivHHpVV_NDI.img"
1041     gdal_calc.py -A $file1 -B $file2 --outfile=../process/"`basename
$file1 .img`_HHmVVdivHHpVV_NDI.img" --calc="(A-B)/(A+B)" --
1042 NoDataValue=0 --format='ENVI' --overwrite
        fi
        done #for file2
1043     done #for file1
1044
1045     echo "moving files to main folder:"
1046     #mv ../process/* ./
1047     mv ../process/* .././g1a1_layerstack_utm_db/$sitename
1048     cd ..
1049
1050     fi # if other_calculations - geocodeG
1051

```

```

1052 #move all the files to the g3_envi_db_folder
1053 cd ../../
1054 mv ./g1a1_layerstack_utm_db/$sitename/* ./g3_envi_db/$sitename/
1055 #mv ./g1a1_layerstack_utm_db/Cameroon1/*
1056 ./g3_envi_db/Cameroon1/
1057
1058 ##=====#
1059 ## create a virtual stack with the newly calculate (HH+HV, HH-HV,
1060 HH/HV) images included
1061 ##=====#
1062 #
1063 ##create a list of filenames with all image files
1064 #for file in *.img;
1065 # do
1066 #   echo $file >> listofnames_all_with_calc_bands.txt
1067 #done
1068 ##create a list of filenames with all the Dry image files
1069 #for file in *_Dry_*.img;
1070 # do
1071 #   echo $file >> listofnames_dry_with_calc_bands.txt
1072 #done
1073 ##create a list of filenames with all the Wet image files
1074 #for file in *_Wet_*.img;
1075 # do
1076 #   echo $file >> listofnames_wet_with_calc_bands.txt
1077 #done
1078
1079 #build the virtual stacks for all, dry and wet sets of images
1080 #echo "Building virtual stack of all images with calculated bands"
1081 #gdalbuildvrt -tr 5 5 -tap -srcnodata 0 -vrtnodata 0 -separate -
1082 input_file_list listofnames_all_with_calc_bands.txt -overwrite
1083 $sitename\_utm_db_stack_all_calc.vrt
1084 #echo "Building virtual stack of dry images with calculated bands"
1085 #gdalbuildvrt -tr 5 5 -tap -srcnodata 0 -vrtnodata 0 -separate -
1086 input_file_list listofnames_dry_with_calc_bands.txt -overwrite
1087 $sitename\_utm_db_stack_dry_calc.vrt
1088 #echo "Building virtual stack of wet images with calculated bands"
1089 #gdalbuildvrt -tr 5 5 -tap -srcnodata 0 -vrtnodata 0 -separate -
1090 input_file_list listofnames_wet_with_calc_bands.txt -overwrite
1091 $sitename\_utm_db_stack_wet_calc.vrt
1092 #echo "done"

```

First start GRASS with \$ grass70. Then run Script 2.18 from inside GRASS. Create a location for each site, with the corresponding UTM zone.

*Script A2.18 texture\_calc\_take2.sh*

```
1  #!/bin/bash
2  # texture_calc_take2.sh - new approach to use correct spatial resolution
   for each sensor
3  # 6 February 2014, Bernard Spies
4  # new edition: 7 August 2014
5
6  # note input raster map needs to be mapped to 0 to 255 for input to
   r.texture: changed to 1 to 64
7  # first step is to run grass70 in the terminal
8  # open correct location - cameroon1_dry for cameroon
9
10 site=$1
11 season=$2
12 do_ringdal=$3 #1=yes, 0=no
13 do_texture=$4 #1=yes, 0=no
14 do_convert_to_int=$5 #1=yes, 0=no
15 do_output=$6 #1=yes, 0=no
16 quick=$7 # quick = 1 for do_convert_to_int, all in one go. quick = 0 to
   run manually step-by-step
17
18 cd ./g3_envi_db/$site
19
20 if [ -e listofnames_all.txt ]
21 then
22     echo "listofnames_all.txt exists"
23     rm listofnames_all.txt
24 fi
25 if [ -e listofnames_dry.txt ]
26 then
27     echo "listofnames_dry.txt exists"
28     rm listofnames_dry.txt
29 fi
30 if [ -e listofnames_wet.txt ]
31 then
32     echo "listofnames_wet.txt exists"
33     rm listofnames_wet.txt
34 fi
35 if [ -e listofnames_wetdry_geocodeG.txt ]
36 then
37     echo "listofnames_wetdry_geocodeG.txt exists"
```

```

38     rm listofnames_wetdry_geocodeG.txt
39 fi
40 if [ -e listofnames_wetdry_geocodeS.txt ]
41 then
42     echo "listofnames_wetdry_geocodeS.txt exists"
43     rm listofnames_wetdry_geocodeS.txt
44 fi
45
46 #create a list of filenames with all image files
47 for file in *.img;
48 do
49     echo $file >> listofnames_all.txt
50 done
51 #create a list of filenames with all the Dry image files
52 for file in *_Dry_*.img;
53 do
54     echo $file >> listofnames_dry.txt
55 done
56 #create a list of filenames with all the Wet image files
57 for file in *_Wet_*.img;
58 do
59     echo $file >> listofnames_wet.txt
60 done
61 #create a list of filenames with all geocodeG image files
62 for file in *_geocodeG_*.img;
63 do
64     echo $file >> listofnames_wetdry_geocodeG.txt
65 done
66 #create a list of filenames with all geocodeS image files
67 for file in *_geocodeS_*.img;
68 do
69     echo $file >> listofnames_wetdry_geocodeS.txt
70 done
71
72 if [ $do_ringdal == 1 ];
73 then
74     # the list of filenames without any div,plus or min calculations
75     if [ -e listofnames_$season\_vanilla_$site\.txt ]
76     then
77         echo "listofnames_$season\_vanilla_$site\.txt exists"
78         rm listofnames_$season\_vanilla_$site\.txt
79     fi
80     # the list of filenames to refer to in GRASS
81     if [ -e listofnames_$season\_vanilla_$site\_grass.txt ]
82     then
83         echo "listofnames_$season\_vanilla_$site\_grass.txt exists"
84         rm listofnames_$season\_vanilla_$site\_grass.txt

```

```

85     fi
86
87     while read line; do # reading the list of filenames in directory, output
without div,plus or min
88         #if [[ $line != *div* ]] && [[ $line != *plus* ]] && [[ $line != *min* ]];
89         if [[ $line == *_shpclip_dB* ]];
90         then
91             echo $line >> listofnames_$season\_vanilla_$site\.txt
92         fi
93     done < listofnames_$season\.txt
94
95     # iterate through the list without div, plus or min, read into grass,
output names to a new textfile
96     i=1
97     while read line; do
98         echo $i
99         if [[ $line == *ASAR* ]];
100         #if [[ $line ==
DRC_Dry_ASAR_20100705_SLC_0_VVH_VH_2x4_15m_dB* ]];
101         then
102             echo $line
103             r.in.gdal $line output=$site.$season\.asar.$i --overwrite
104             echo $site.$season\.asar.$i >>
listofnames_$season\_vanilla_$site\_grass.txt
105             #r.in.gdal $line output=`basename $line _dB.img` --overwrite
106             #echo `basename $line _dB.img` >>
listofnames_$season\_vanilla_$site\_grass.txt
107         fi
108         if [[ $line == *PALSAR* ]];
109         then
110             echo $line
111             r.in.gdal $line output=$site.$season\.palsar.$i --overwrite
112             echo $site.$season\.palsar.$i >>
listofnames_$season\_vanilla_$site\_grass.txt
113             #r.in.gdal $line output=`basename $line _dB.img` --overwrite
114             #echo `basename $line _dB.img` >>
listofnames_$season\_vanilla_$site\_grass.txt
115         fi
116         if [[ $line == *TSX* ]];
117         then
118             echo $line
119             r.in.gdal $line output=$site.$season\.tsx.$i --overwrite
120             echo $site.$season\.tsx.$i >>
listofnames_$season\_vanilla_$site\_grass.txt
121             #r.in.gdal $line output=`basename $line _dB.img` --overwrite
122             #echo `basename $line _dB.img` >>
listofnames_$season\_vanilla_$site\_grass.txt

```

```

123     fi
124     i=$((i+1))
125
126     done < listofnames_$season\_vanilla_$site\.txt
127 fi # do_ringdal
128
129 #r.texture
130 if [ $do_texture == 1 ]
131 then
132     # test if a textfile with the names of the output texture files exists?
133     if [ -e listofnames_$site\_ $season\_ textures_grass.txt ]
134     then
135         echo "listofnames_$site\_ $season\_ textures_grass.txt exists"
136         rm listofnames_$site\_ $season\_ textures_grass.txt
137     fi
138
139     # iterate through the grass filenames, and create texture layers with
    specified window sizes
140     # output the new GRASS filenames to a textfile, to be used in the
    r.out.gdal step
141     while read line; do
142         if [[ $line == *asar* ]];
143         then
144             g.region rast=$line
145             r.mapcalc "$line.linear=(10^($line/10))*10000"
146             #r.mapcalc
    "asar.vh.1.linear10000=(10^(asar.vh.1.db/10))*10000"
147             #r.mapcalc "Cameroon1.asar=(10^($line/10))*1000"
148             #r.rescale input=$line output=$line.scaled.db to=0,255
149             r.rescale input=$line.linear output=$line.scaled.linear to=1,64
150
151             # for window in 5 7 9
152             for window in 3 5 7 9 #new verion post 20140808
153             do
154                 echo "running the r.texture step for $line"
155                 echo "window size: $window"
156
157                 # standard
158                 #r.texture input=$line prefix=$line\_ $window\x$window
    size=$window distance=1 method=contrast,entr,corr --verbose
159                 #r.neighbors input=$line
    output=$line\_ $window\x$window\_ mean method=average
    size=$window
160
161                 # scaled, but from db (not from linear)
162

```

```

#r.texture input=$line.scaled.db
163 prefix=$line.scaled_$window\x$window.db size=$window distance=1
method=contrast,entr,corr --verbose
r.neighbors input=$line
164 output=$line.scaled_$window\x$window.db_mean method=average
165 size=$window
166
# scaled version, from linear, not from db
r.texture input=$line.scaled.linear
167 prefix=$line.scaled_$window\x$window.linear size=$window distance=1
168 method=contrast,entr,corr,var --verbose

169 echo $line.scaled_$window\x$window.linear_Contr >>
listofnames_$site\_ $season\_ textures_grass.txt
170 echo $line.scaled_$window\x$window.linear_Corr >>
listofnames_$site\_ $season\_ textures_grass.txt
171 echo $line.scaled_$window\x$window.linear_Entr >>
listofnames_$site\_ $season\_ textures_grass.txt
172 echo $line.scaled_$window\x$window.linear_Var >>
listofnames_$site\_ $season\_ textures_grass.txt
173 echo $line.scaled_$window\x$window.db_mean >>
174 listofnames_$site\_ $season\_ textures_grass.txt

r.neighbors input=$line.scaled.linear
175 output=$line.scaled_$window\x$window.linear_mean method=average
176 size=$window
177 done # asar while
178 fi
179
if [[ $line == *palsar* ]];
180 then
181 r.mapcalc "$line.linear=(10^($line/10))*10000"
182 #r.mapcalc
183 "asar.vh.1.linear10000=(10^(asar.vh.1.db/10))*10000"
184 #r.mapcalc "Cameroon1.asar=(10^($line/10))*1000"
185 #r.rescale input=$line output=$line.scaled.db to=0,255
186 r.rescale input=$line.linear output=$line.scaled.linear to=1,64
187
188 # for window in 7 9 11
189 for window in 5 7 9 11 #new verion post 20140808
190 do
191 echo "running the r.texture step for $line"
192 echo "window size: $window"
193
# standard
194 #r.texture input=$line prefix=$line\_ $window\x$window
size=$window distance=1 method=contrast,entr,corr --verbose

```

```

#r.neighbors input=$line
195 output=$line\_x$window\_x$window\_mean method=average
196 size=$window
197
198 # scaled, but from db (not from linear)
199 #r.texture input=$line.scaled.db
prefix=$line.scaled\_x$window.db size=$window distance=1
method=contrast,entr,corr --verbose
200 r.neighbors input=$line
output=$line.scaled\_x$window.db\_mean method=average
201 size=$window
202
# scaled version, from linear, not from db
r.texture input=$line.scaled.linear
203 prefix=$line.scaled\_x$window.linear size=$window distance=1
204 method=contrast,entr,corr,var --verbose

205 echo $line.scaled\_x$window.linear_Contr >>
listofnames_$site\_season\_textures_grass.txt
206 echo $line.scaled\_x$window.linear_Corr >>
listofnames_$site\_season\_textures_grass.txt
207 echo $line.scaled\_x$window.linear_Entr >>
listofnames_$site\_season\_textures_grass.txt
208 echo $line.scaled\_x$window.linear_Var >>
listofnames_$site\_season\_textures_grass.txt
209 echo $line.scaled\_x$window.db\_mean >>
210 listofnames_$site\_season\_textures_grass.txt

#r.neighbors input=$line.scaled.linear
211 output=$line.scaled\_x$window.linear\_mean method=average
212 size=$window
213 done # palsar while
214 fi
215
216 if [[ $line == *tsx* ]];
217 then
r.mapcalc "$line.linear=(10^($line/10))*10000"
218 #r.mapcalc
219 "asar.vh.1.linear10000=(10^(asar.vh.1.db/10))*10000"
220 #r.mapcalc "Cameroon1.asar=(10^($line/10))*1000"
221
222 #r.rescale input=$line output=$line.scaled.db to=0,255
223 r.rescale input=$line.linear output=$line.scaled.linear to=1,64
224
225 # for window in 7 9 11 13
226 for window in 9 11 13 15 #new verion post 20140808
227 do

```



```

228         echo "running the r.texture step for $line"
229         echo "window size: $window"
230
231         # standard
232         #r.texture input=$line prefix=$line\_ $window\x$window
233         size=$window distance=1 method=contrast,entr,corr --verbose
234         #r.neighbors input=$line
235         output=$line\_ $window\x$window\_mean method=average
236         size=$window
237
238         # scaled, but from db (not from linear)
239         #r.texture input=$line.scaled.db
240         prefix=$line.scaled\_ $window\x$window.db size=$window distance=1
241         method=contrast,entr,corr --verbose
242         r.neighbors input=$line
243         output=$line.scaled\_ $window\x$window.db\_mean method=average
244         size=$window
245
246         # scaled version, from linear, not from db
247         r.texture input=$line.scaled.linear
248         prefix=$line.scaled\_ $window\x$window.linear size=$window distance=1
249         method=contrast,entr,corr,var --verbose
250
251         echo $line.scaled\_ $window\x$window.linear_Contr >>
252         listofnames\_ $site\_ $season\_ textures\_grass.txt
253         echo $line.scaled\_ $window\x$window.linear_Corr >>
254         listofnames\_ $site\_ $season\_ textures\_grass.txt
255         echo $line.scaled\_ $window\x$window.linear_Entr >>
256         listofnames\_ $site\_ $season\_ textures\_grass.txt
257         echo $line.scaled\_ $window\x$window.linear_Var >>
258         listofnames\_ $site\_ $season\_ textures\_grass.txt
259         echo $line.scaled\_ $window\x$window.db_mean >>
260         listofnames\_ $site\_ $season\_ textures\_grass.txt
261
262         #r.neighbors input=$line.scaled.linear
263         output=$line.scaled\_ $window\x$window.linear_mean method=average
264         size=$window
265
266         done # tsx while
267     fi
268
269     done < listofnames\_ $season\_ vanilla\_ $site\_ grass.txt
270
271 fi # do_texture
272
273 #while read line; do
274 # if [[ $line == *tsx* ]];

```

```

258 # then
259 #   r.mapcalc "$line.linear=(10^($line/10))*10000"
260 #   r.rescale input=$line.linear output=$line.scaled.linear to=1,64
261 #
262 #   for window in 9 11 13 15
263 #   do
264 #       echo "running the r.texture step for $line"
265 #       echo "window size: $window"
266 #       r.neighbors input=$line
267 output=$line.scaled_$window\x$window.db_mean method=average
268 size=$window
269 #
378 #       # scaled version, from linear, not from db
379 #       r.texture input=$line.scaled.linear
380 prefix=$line.scaled_$window\x$window.linear size=$window distance=1
379 method=contrast,entr,corr,var --verbose
380 #   done # tsx while
381 # fi
382 #done < listofnames_$season\_vanilla_$site\_grass.txt
383
384 # do convert to int, normal version (run all in one go)
385 if [ $do_convert_to_int == 1 ] && [ $quick == 1 ];
386 then
387     while read line; do
388
389         if [[ $line == *Contr* ]];
390         then
391             echo $line
392             r.mapcalc "$line.2=round($line*100)"
393             fi # Contr
394
395             if [[ $line == *Corr* ]];
396             then
397                 echo $line
398                 r.mapcalc "$line.2=round($line*100)"
399                 fi # Corr
400
401                 if [[ $line == *Entr* ]];
402                 then
403                     echo $line
404                     r.mapcalc "$line.2=round($line*100)"
405                     fi # Entr
406
407                     if [[ $line == *Var* ]];
408                     then
409                         echo $line
410                         r.mapcalc "$line.2=round($line*100)"

```

```

411     fi # Entr
412
413     if [[ $line == *mean* ]];
414     then
415         echo $line
416         r.mapcalc "$line.2=round($line)"
417     fi # mean
418
419     #r.out.gdal input=$line output=$line.img format=ENVI type=
420     done < listofnames_$site\_ $season\_ textures_grass.txt
421 fi # do_convert_to_int
422 j=1
423
424 if [ $do_convert_to_int == 1 ] && [ $quick == 0 ];
425 then
426     while read line; do
427         if [[ $line == *Contr* ]] && [[ $line == *.$j.* ]];
428         then
429             echo $line
430             r.mapcalc "$line.2=round($line*100)"
431         fi # Contr
432
433         if [[ $line == *Corr* ]] && [[ $line == *.$j.* ]];
434         then
435             echo $line
436             r.mapcalc "$line.2=round($line*100)"
437         fi # Corr
438
439         if [[ $line == *Entr* ]] && [[ $line == *.$j.* ]];
440         then
441             echo $line
442             r.mapcalc "$line.2=round($line*100)"
443         fi # Entr
444
445         if [[ $line == *Var* ]] && [[ $line == *.$j.* ]];
446         then
447             echo $line
448             r.mapcalc "$line.2=round($line*100)"
449         fi # Entr
450
451         if [[ $line == *mean* ]] && [[ $line == *.$j.* ]];
452         then
453             echo $line
454             r.mapcalc "$line.2=round($line)"
455         fi # mean
456     done
457     #r.out.gdal input=$line output=$line.img format=ENVI type=

```

```

456     done < listofnames_$site\_season\_textures_grass.txt
457 fi # do_convert_to_int
458
459
460     currentdir=${PWD}
461     echo $currentdir
462
463 #r.out.gdal
464 if [ $do_output == 1 ];
465 then
466     #cd ../../g4_texture_measures_envi/$site
467     #currentdir=${PWD}
468     # add a step to read the textfile with all the GRASS texture measures,
469     and replace all the dots (.) with underscores (_)
470     while read line; do
471
472         #r.out.gdal input=$line.2
473         output=../../g4_texture_measures_envi/$site/$line.img format=ENVI
474         type=Int16 nodata=-999 --overwrite
475         if [[ $line == *Contr* ]];
476         then
477             echo $line
478             r.out.gdal input=$line.2
479             output=../../g4_texture_measures_envi/$site/$line.img format=ENVI
480             type=Int16 nodata=-999 --overwrite
481             fi # Contr
482
483             if [[ $line == *Corr* ]];
484             then
485                 echo $line
486                 r.out.gdal input=$line.2
487                 output=../../g4_texture_measures_envi/$site/$line.img format=ENVI
488                 type=Int16 nodata=-999 --overwrite
489                 fi # Corr
490
491                 if [[ $line == *Entr* ]];
492                 then
493                     echo $line

```

```

        r.out.gdal input=$line.2
output=../../g4_texture_measures_envi/$site/$line.img format=ENVI
494 type=Int32 nodata=-999 --overwrite
495     fi
496
497     if [[ $line == *mean* ]];
498     then
499         echo $line
        r.out.gdal input=$line.2
        output=../../g4_texture_measures_envi/$site/$line.img format=ENVI
500 type=Int16 nodata=-999 --overwrite
501     fi # mean
502 done < listofnames_$site\_ $season\_ textures_grass.txt
fi # do_output

```

*Script A2.19 g3b\_copy\_to\_largestack\_folder.sh*

```

1  #!/bin/bash
2  # copy all the intensity, ratios, textures and SRTM layers to the largestack
3  folder
4  # resample to 5m spatial resolution (next step?)
5  site=$1
6  do_copy_elevationdata=1
7  do_copy_intensities_and_ratios=1
8  do_copy_textures=1
9
10 if [ $do_copy_elevationdata == 1 ];
11 then
12     cd ./f3_layerstack_overlap
13     SHPFILE=overlap_$1.shp
14     cd ../
15     cd ./g4c_elevation_cutout
16
17     if ! [ -e *elevation* ];
18     then
19         echo "cutout elevation file"
20         #cp
21         ../../Z_SRTM_UTM/output_resampled/$sitenam\_wet\_dry\_elevation* .
22         #cp ../../Z_SRTM_UTM/output/$site\_wet\_dry\_elevation* .
23         cd ../../Z_SRTM_UTM/output
24         for RASTERFILE in $site\_wet\_dry\_*.img;
25         do
26             gdalwarp -co COMPRESS=DEFLATE -co TILED=YES -of ENVI -r near -
27             cutline ../../step2_after_geocode/f3_layerstack_overlap/$SHPFILE -srcnodata
28             0 -dstnodata 0 $RASTERFILE

```

```

26  ../../step2_after_geocode/g4c_elevation_cutout/`basename $RASTERFILE
27  .img`_cutout.img
28
29  done
30  fi
31
32  #back to starting directory
33  cd ../../step2_after_geocode/
34  echo "copy and resample elevation,slope,aspect"
35  cd ./g4c_large_stack_resampled/$site
36
37  if ! [ -e *wet_dry* ];
38  then
39      cd ../../g4c_elevation_cutout/
40      #echo "moving texture images"
41      #mv ../../g4_texture_measures_envi/$site/* .
42
43      echo "resampling, moving texture images"
44      for filename in $site\_wet_dry_*.img;
45      #for filename in Cameroon1.wet.tsx.9.scaled_9x9.db_mean.img;
46      do
47          echo $filename
48          gdalwarp -tr 5 5 -r near -dstnodata -999 -ot Int16 -of ENVI -overwrite
49          $filename ../../g4c_large_stack_resampled/$site/`basename $filename
50          _cutout.img`_cutout_resampled.img
51      done
52      cd ../../g4c_large_stack_resampled/$site
53      fi
54      cd ../../
55  fi
56
57  #resample the elevation data
58  #if ! [ -e *slope* ]
59  #then
60  # echo "copying slope file"
61  # #cp ../../Z_SRTM_UTM/output_resampled/$sitename\_wet_dry_slope*
62  # .
63  # cp ../../Z_SRTM_UTM/output/$site\_wet_dry_slope* .
64  #
65  #fi
66
67  #if ! [ -e *aspect* ]
68  #then
69  # echo "copying aspect file"
70  # #cp
71  ../../Z_SRTM_UTM/output_resampled/$sitename\_wet_dry_aspect* .
72  # cp ../../Z_SRTM_UTM/output/$site\_wet_dry_aspect* .
73  #

```

```

69     #fi
70 #fi
71
72 if [ $do_copy_intensities_and_ratios == 1 ];
73 then
74     echo "copying intensities and ratios"
75     cd ./g4c_large_stack_resampled/$site
76     if ! [ -e *plus* ];
77     then
78         cd ../../g3_envi_db/$site/
79         echo "resampling, moving intensity and ratio layers"
80         for filename in *.img;
81         #for filename in Cameroon1.wet.tsx.9.scaled_9x9.db_mean.img;
82         do
83             echo $filename
84
85
86             gdalwarp -tr 5 5 -r near -dstnodata -999 -ot Float32 -of ENVI -overwrite
$filename ../../g4c_large_stack_resampled/$site/"`basename $filename
.img`_resampled.img"
87         done
88     fi
89     #back to starting directory
90     cd ../../
91 fi
92
93 if [ $do_copy_textures == 1 ];
94 then
95     echo "copy textures"
96     cd ./g4c_large_stack_resampled/$site
97     if ! [ -e *scaled* ];
98     then
99         cd ../../g4_texture_measures_envi/$site/
100         #echo "moving texture images"
101         #mv ../../g4_texture_measures_envi/$site/* .
102
103         echo "resampling, moving texture images"
104         for filename in *.img;
105         #for filename in Cameroon1.wet.tsx.9.scaled_9x9.db_mean.img;
106         do
107             #echo $filename
108             #gdalwarp -tr 5 5 -r near -dstnodata -999 -ot Int16 -of ENVI -overwrite
$filename ../../g4c_large_stack_resampled/$site/"`basename $filename
.img`_resampled.img"
109
110         #The Var calculations have a greater range than 32676, therefore Int32 is
required instead of Int16,

```

```

111     #as for the Contr,Corr,Entr,Mean Texture measures
112     if [[ $filename == *Contr* ]];
113     then
114         echo $filename
115         gdalwarp -tr 5 5 -r near -dstnodata -999 -ot Int16 -of ENVI -overwrite
$filename ../../g4c_large_stack_resampled/$site/"`basename $filename
.img`_resampled.img"
116     fi # Contr
117
118     if [[ $filename == *Corr* ]];
119     then
120         echo $filename
121         gdalwarp -tr 5 5 -r near -dstnodata -999 -ot Int16 -of ENVI -overwrite
$filename ../../g4c_large_stack_resampled/$site/"`basename $filename
.img`_resampled.img"
122     fi # Corr
123
124     if [[ $filename == *Entr* ]];
125     then
126         echo $filename
127         gdalwarp -tr 5 5 -r near -dstnodata -999 -ot Int16 -of ENVI -overwrite
$filename ../../g4c_large_stack_resampled/$site/"`basename $filename
.img`_resampled.img"
128     fi # Entr
129
130     if [[ $filename == *Var* ]];
131     then
132         echo $filename
133         gdalwarp -tr 5 5 -r near -dstnodata -999 -ot Int32 -of ENVI -overwrite
$filename ../../g4c_large_stack_resampled/$site/"`basename $filename
.img`_resampled.img"
134     fi
135
136     if [[ $filename == *mean* ]];
137     then
138         echo $filename
139         gdalwarp -tr 5 5 -r near -dstnodata -999 -ot Int16 -of ENVI -overwrite
$filename ../../g4c_large_stack_resampled/$site/"`basename $filename
.img`_resampled.img"
140     fi # mean
141
142     done
143 fi
144 #back to starting directory
145 cd ../../
146 fi

```



```
1  #!/bin/bash
2  sitename=$1
3
4  #=== build a virtual stack of the images
5  #cd ./g3_envi_db/$sitename
6  cd ./g4c_large_stack_resampled/$sitename
7
8  if [ -e listofnames_all.txt ]
9  then
10   echo "listofnames_all.txt exists"
11   rm listofnames_all.txt
12 fi
13
14 if [ -e listofnames_dry.txt ]
15 then
16   echo "listofnames_dry.txt exists"
17   rm listofnames_dry.txt
18 fi
19
20 if [ -e listofnames_wet.txt ]
21 then
22   echo "listofnames_wet.txt exists"
23   rm listofnames_wet.txt
24 fi
25 if [ -e listofnames_wetdry_geocodeG.txt ]
26 then
27   echo "listofnames_wetdry_geocodeG.txt exists"
28   rm listofnames_wetdry_geocodeG.txt
29 fi
30 if [ -e listofnames_wetdry_geocodeS.txt ]
31 then
32   echo "listofnames_wetdry_geocodeS.txt exists"
33   rm listofnames_wetdry_geocodeS.txt
34 fi
35
36 #create a list of filenames with all image files
37 for file in *.img;
38 do
39   echo $file >> listofnames_all.txt
40 done
41 #create a list of filenames with all the Dry image files
42 for file in *_Dry_*.img;
43 do
44   echo $file >> listofnames_dry.txt
45 done
```

```

46 for file in *dry*.img;
47 do
48     echo $file >> listofnames_dry.txt
49 done
50
51 #create a list of filenames with all the Wet image files
52 for file in *_Wet_*.img;
53 do
54     echo $file >> listofnames_wet.txt
55 done
56 for file in *wet*.img;
57 do
58     echo $file >> listofnames_wet.txt
59 done
60
61 #create a list of filenames with all geocodeG image files
62 for file in *_geocodeG_*.img;
63 do
64     echo $file >> listofnames_wetdry_geocodeG.txt
65 done
66 #create a list of filenames with all geocodeS image files
67 for file in *_geocodeS_*.img;
68 do
69     echo $file >> listofnames_wetdry_geocodeS.txt
70 done
71
72 #build the virtual stacks for all, dry and wet sets of images
73 echo "Building virtual stack of all images"
74 gdalbuildvrt -tr 5 5 -tap -srcnodata 0 -vrtnodata 0 -separate -input_file_list
75 listofnames_all.txt $sitename\_utm_db_stack_all.vrt
76
77 echo "Building virtual stack of dry images"
78 gdalbuildvrt -tr 5 5 -tap -srcnodata 0 -vrtnodata 0 -separate -input_file_list
79 listofnames_dry.txt $sitename\_utm_db_stack_dry.vrt
80
81 echo "Building virtual stack of dry images"
82 gdalbuildvrt -tr 5 5 -tap -srcnodata 0 -vrtnodata 0 -separate -input_file_list
83 listofnames_wet.txt $sitename\_utm_db_stack_wet.vrt
84
85 echo "Building virtual stack of geocodeG images"
86 gdalbuildvrt -tr 5 5 -tap -srcnodata 0 -vrtnodata 0 -separate -input_file_list
87 listofnames_wetdry_geocodeG.txt
88 $sitename\_utm_db_stack_wetdry_geocodeG.vrt
89 echo "done"
90
91 echo "Building virtual stack of geocodeS images"
92

```

```

93 gdalbuildvrt -tr 5 5 -tap -srcnodata 0 -vrtnodata 0 -separate -input_file_list
94 listofnames_wetdry_geocodeS.txt
95 $sitename\_utm_db_stack_wetdry_geocodeS.vrt
echo "done"

```

*Script A2.21 g3c\_envi\_realstacks.sh*

```

1  #!/bin/bash
2  # create real ENVI stacks from RandomForest Variable Importance results, to
   use in EnMAP
3  sitename=$1
4
5  if [ $sitename == 'Cameroon1' ];
6  then
7      cd ./g4c_large_stack_resampled/$sitename/
8
9      gdal_merge.py -o ../../g4c_realstacks/$sitename/cdlcx_varimp_rf_top20.dat
10 -of ENVI -ps 5 5 -tap -v -separate -n 0 -a_nodata -999 --optfile
11 listofnames_cdlcx_varimp_rf_top20.txt
12 fi

```

*Script A2.22 do\_convert\_to\_raster\_all.sh*

```

1  #!/bin/bash
2
3  #./convert_to_raster_all.sh cameroon1
4  ./convert_to_raster_all.sh chad
5  ./convert_to_raster_all.sh drc
6  ./convert_to_raster_all.sh sudan
7  ./convert_to_raster_all.sh tanzania

```

*Script A2.23 convert\_to\_raster\_all.sh*

```

1  #!/bin/bash
2  #for Cameroon1 site
3  #get extent from dry.vrt stack in R
4
5  #extents
6  #cameroon1 250940 686980 265210 726175
7  #chad 614183.754502 1496430 634162.966218 1518376.961734
8  #drc 799835 133060 819600.459909 181065
9  #sudan 698315 1444010 713572.614335 1456355.125502
10 #tanzania 390720 9532185 403890 9577710
11

```

```

12 site=$1
13 #site='tanzania'
14
15 if [ $site == 'cameroon1' ];
16 then
17     echo 'processing: ' $site
18     cd ./shapefiles/cameroon1_samples_5050/
19
20     #all - training, validation and test
21     gdal_rasterize -a lc10_split -of ENVI -a_nodata -999 -te 250445 686505
22     265745 726680 -tr 5 5 -ot Int16 cameroon1_samples.shp
23     cameroon1_samples_raster.dat
24     cd ../../
25 fi #cameroon1
26
27 if [ $site == 'chad' ];
28 then
29     echo 'processing: ' $site
30     cd ./chad/
31     #all - training, validation and test
32     #gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 614183.754502 1496430
33     634162.966218 1518376.961734 -tr 5 5 -ot Int16 chad_samples_UTM34N.shp
34     ./R_samples_extract_rasters/chad_samples_raster_new.dat
35     gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 607915 1495330 637715
36     1530430 -tr 5 5 -ot Int16 chad_samples_UTM34N.shp
37     ./R_samples_extract_rasters/chad_samples_raster_new.dat
38     cd ../../
39
40 fi #chad
41
42 if [ $site == 'drc' ];
43 then
44     echo 'processing: ' $site
45     cd ./drc/
46
47     #all - training, validation and test
48     #gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 799835 133060
49     819600.459909 181065 -tr 5 5 -ot Int16 drc_samples_all_UTM35N.shp
50     ./R_samples_extract_rasters/drc_samples_raster_new.dat
51
52     gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 799285 132670 819600
53     188515 -tr 5 5 -ot Int16 drc_samples_all_UTM35N.shp
54     ./R_samples_extract_rasters/drc_samples_raster_new.dat
55
56     gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 799285 132670 819605
57     188515 -tr 5 5 -ot Int16 drc_samples_all_UTM35N.shp
58     ./R_samples_extract_rasters/drc_samples_raster_new_wet.dat

```

```

47
48   cd ../../
49   fi #drc
50
51   if [ $site == 'sudan' ];
52   then
53       echo 'processing: ' $site
54       cd ./sudan/
55
56       #all - training, validation and test
57       #gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 799835 133060
58       819600.459909 181065 -tr 5 5 -ot Int16 sudan_samples.shp
59       ./R_samples_extract_rasters/sudan_samples_raster_new.dat
60
61       gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 697850 1443600 714490
62       1459325 -tr 5 5 -ot Int16 sudan_samples.shp
63       ./R_samples_extract_rasters/sudan_samples_raster_new.dat
64
65       gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 697850 1443600 714490
66       1459325 -tr 5 5 -ot Int16 sudan_samples.shp
67       ./R_samples_extract_rasters/sudan_samples_raster_new_wet.dat
68       cd ../../
69       fi #sudan
70
71   #old version 392110 9532550 404375 9577625
72   #tanzania new version extent 390720 9577710 403890 9532185
73
74   #for Tanzania site
75   if [ $site == 'tanzania' ];
76   then
77       echo 'processing: ' $site
78       cd ./tanzania/
79       #all - training, validation and test
80       #gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 390720 9532185 403890
81       9577710 -tr 5 5 -ot Int16 tanzania_samples_all.shp
82       ./R_samples_extract_rasters/tanzania_samples_all_raster_new.dat
83       #get extent values from R tanzania.dry.stack)
84       gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 390715 9532180 404375
85       9577715 -tr 5 5 -ot Int16 tanzania_samples_all.shp
86       ./R_samples_extract_rasters/tanzania_samples_all_raster_new.dat
87
88       gdal_rasterize -a lc_nr -of ENVI -a_nodata -999 -te 390720 9532180 404375
89       9577710 -tr 5 5 -ot Int16 tanzania_samples_all.shp
90       ./R_samples_extract_rasters/tanzania_samples_all_raster_new_wet.dat
91       fi

```

## R modelling and graphing scripts

### A3.1 Overview of R modelling and graphing scripts

An overview of the R scripts with the corresponding script number are given in Table A3.1 for the R data preparation for modelling scripts, in Table A3.2 for the R modelling functions and in Table A3.3 for the R plotting / graphing functions. Only extracts and summaries from the full R scripts are given in Appendix Three, with the focus on R functions that were written for this research project.

*Table A3.1 R data preparation for modelling scripts (Section A3.3.1)*

<b>Script number</b>	<b>Script name</b>	<b>Description</b>
A3.1	Set the working directory and load the required R packages.	R packages used in the remainder of the scripts.
A3.2	Extract data	Extract data from the raster images to data frames in R.
A3.3	Define input feature combinations	Input feature layers groups of backscatter coefficients, interchannel ratios, texture measures and elevation data.

Table A3.2 R functions for modelling (Section A3.3.2)

Script number	Script name	Description
A3.4	model_and_test_rf function	Returns the confusion matrix of the random forest model.
A3.5	model_rf function	Returns the actual random forest model, to be applied on new data.
A3.6	varimp_rf	Returns the MDA and MDG variable importance measures for all input variables.
A3.7	varimp_rf_unscaled	The unscaled version of the MDA and MDG variable importance measures.
A3.8	model_and_test_svm	Returns the confusion matrix from the SVM model.
A3.9	model_svm	Returns the SVM model to be applied on new data.
A3.10	rf.topX.conf	Runs a random forest model on the top X variables, based on either the MDA or MDG variable importance measure. The function returns the confusion matrix between the training and test datasets.
A3.11	rf.topX.model	Returns the random forest model consisting only of the top X variables, based on either the MDA or MDG variable importance measures.
A3.12	rf.topX.varimp	Returns the variable importance measures of the top X variables from a random forest model.
A3.13	svm.topX.conf	Returns the confusion matrix based on the training and test datasets, for the SVM based on the top X variables (based on random forest modelling).
A3.14	svm.topX.model	Returns the SVM model for the top X variables (based on random forest modelling).
A3.15	apply_KL_divergence	Apply the Kullback-Leibler divergence between two distributions of data.

Table A3.3 R functions for plotting graphs (Section A3.3.3)

Script number	Script name	Description
A3.16	multiplot	The multiplot functions enables multiple plots to be plotted to the output window. This is reproduced from <a href="http://www.cookbook-r.com">www.cookbook-r.com</a>
A3.17	plot_overall_accuracy_with_class_accuracy	A plotting function to plot the overall accuracy and class-specific accuracies on the same plot.
A3.18	plot_overall_accuracy_with_class_accuracy_elevation_only	A variation of Script A3.17, that uses only the elevation data as input.
A3.19	convert_pat_to_lcx_dBite	A function the converts the variable names from P (PALSAR), A (ASAR) and T (TerraSAR-X), to L, C and X.
A3.20	corrplot	Using the corrplot function from the corrplot R package.
A3.21	plot_overall_accuracy_with_class_accuracy_top1_to_30_in1s	A function to plot the overall and class-specific accuracies for the top 30 variables, by adding variables incrementally from 1 to 30.
A3.22	apply_model1_on_site2_conf_newsite_cw_traintest	A function the develops the random forest model on one set of data, and apply it on an additional set of data, either of a new season or an additional site.



## A3.2 R random forest modelling scripts

A list of the main R packages used in this research, are given in Script A3.1. This is not the full list, and packages applicable to specific functions are given in the respective scripts.

### A3.2.1. R data preparation for modelling scripts

*Script A3.1 Set working directory and loading R packages*

```
1 rm (list = ls())
2 setwd("/home/bernard/sharefolder/RProjects/signature_extract")
3
4 # packages
5 require(raster)
6 require(sp)
7 require(rgdal)
8 require(shapefiles)
9 library(maptools)
10 library(MASS)
11 library(gplots)
12 library(caTools)
13 library(ggplot2)
14 library(reshape2)
15 library(plyr)
16 library(data.table)
17 library(splancs)
18 library(spatstat)
19 library(caret)
20 library(randomForest)
21 library(ROCR)
22 library(ggplot2)
23 library(ellipse)
24 library(Hmisc)
25 library(dplyr)
```

```

1  cam1.dry.stack <-
   brick("/home/bernard/sharefolder/A_Data_PhD/A_Alldata/step2_after_geoc
   ode/g4c_large_stack_resampled/Cameroon1/Cameroon1_utm_db_stack_dr
   y.vrt")
2      extent(cam1.dry.stack) # get extent for convert_raster_to_shape.sh
   script
3
4  c.trv_t.raster <-
   raster("/home/bernard/sharefolder/qgis/shapefiles/cameroon1_samples_50
   50/Cameroon1_all_samples_5050_raster_from_prev_one.dat")
5      c.trv.raster <-
   raster("/home/bernard/sharefolder/qgis/shapefiles/cameroon1_samples_50
   50/Cameroon1_training_samples_5050_raster_clipped.envi")
6      c.test.raster <-
   raster("/home/bernard/sharefolder/qgis/shapefiles/cameroon1_samples_50
   50/Cameroon1_test_samples_5050_raster_clipped.envi")

7  set.seed(12345)
8      #cameroon1 training
9      c.trv.samples3000 <- sampleStratified(c.trv.raster,
10 size=3000,na.rm=TRUE,xy=T,sp=T)
11      c.trv.samples3000
12 table(c.trv.samples3000$Cameroon1_training_samples_5050_raster_clipped
13 )

14 c.test.samples3000 <- sampleStratified(c.test.raster,
15 size=3000,na.rm=TRUE,xy=T,sp=T)
16      c.test.samples3000
17
18 table(c.test.samples3000$Cameroon1_test_samples_5050_raster_clipped)
19
20 save(c.trv.samples3000,c.test.samples3000,file="cameroon1_training_test_s
21 amples3000.rda")
22      writeOGR(c.trv.samples3000, dsn='polys/', layer='c1s3000trv',
23 driver='ESRI Shapefile',overwrite=TRUE)
24      writeOGR(c.test.samples3000, dsn='polys/', layer='c1s3000test',
25 driver='ESRI Shapefile',overwrite=TRUE)

26 #=====
27 #now extract the values from the stack of images
28 c.trv_t.raster <-
   raster("/home/bernard/sharefolder/qgis/shapefiles/cameroon1_samples_50
   50/Cameroon1_all_samples_5050_raster_from_prev_one.dat")

```

```

29   c.trv.raster <-
raster("/home/bernard/sharefolder/qgis/shapefiles/cameroon1_samples_50
50/Cameroon1_training_samples_5050_raster_clipped.envi")
30   c.test.raster <-
raster("/home/bernard/sharefolder/qgis/shapefiles/cameroon1_samples_50
50/Cameroon1_test_samples_5050_raster_clipped.envi")
31
32   cam1.dry.stack <-
brick("/home/bernard/sharefolder/A_Data_PhD/A_Alldata/step2_after_geoc
ode/g4c_large_stack_resampled/Cameroon1/Cameroon1_utm_db_stack_dr
y.vrt")
33   cam1.dry.stack <- addLayer(cam1.dry.stack,c.trv_t.raster)
34   cam1.dry.stack
35
36   c.trv_t.raster <-
raster("/home/bernard/sharefolder/qgis/shapefiles/cameroon1_samples_50
50/Cameroon1_all_samples_5050_raster_from_prev_one.dat")
37   c.trv.raster <-
raster("/home/bernard/sharefolder/qgis/shapefiles/cameroon1_samples_50
50/Cameroon1_training_samples_5050_raster_clipped.envi")
38   c.test.raster <-
raster("/home/bernard/sharefolder/qgis/shapefiles/cameroon1_samples_50
50/Cameroon1_test_samples_5050_raster_clipped.envi")
39   cam1.dry.stack <-
brick("/home/bernard/sharefolder/A_Data_PhD/A_Alldata/step2_after_geoc
ode/g4c_large_stack_resampled/Cameroon1/Cameroon1_utm_db_stack_dr
y.vrt")
40   cam1.dry.stack <- addLayer(cam1.dry.stack,c.trv_t.raster)
41   cam1.dry.stack
42
43   #save(c.trv.samples3000,c.test.samples3000,file="cameroon1_training_test_
samples3000.rda") #note from higher up in the code
44   #load the 3000 samples per class.
45   load("cameroon1_training_test_samples3000.rda")
46   #cameroon1 training
47   cam1_dry_all_trv3000 <-
extract(cam1.dry.stack,c.trv.samples3000,layer=1,nl=dim(cam1.dry.stack)[3],
sp=TRUE)
48   save(cam1_dry_all_trv3000,file="cam1_dry_all_trv3000.rda")
49   #cameroon1 test
50   cam1_dry_all_test3000 <-
extract(cam1.dry.stack,c.test.samples3000,layer=1,nl=dim(cam1.dry.stack)[3]
,sp=TRUE)
51   save(cam1_dry_all_test3000,file="cam1_dry_all_test3000.rda")
52   save(cam1_dry_all_trv3000,cam1_dry_all_test3000,
file="cam1_dry_training_test3000.rda")
53   #write to csv

```

```

54
55 write.csv(cam1_dry_all_trv3000,"/home/bernard/sharefolder/RProjects/sign
56 ature_extract/cam1_dry_all_trv3000.csv")
57
58 write.csv(cam1_dry_all_test3000,"/home/bernard/sharefolder/RProjects/sig
59 nature_extract/cam1_dry_all_test3000.csv")

```

*Script A3.3 Define input feature combinations*

```

1  cam1_dry_colnames <- c("ID","cell","x","y","class2", "C1DAVHG",
2  "C1DAVHS","C1DAVVG","C1DAVHdivVV_linearG","C1DAVHdivVV_rdBG",
3  "C1DAVH5meanG","C1DAVH5ContrG","C1DAVH5CorrG","C1DAVH5EntrG",
4  "C1DAVH5VarG","C1DAVH7meanG","C1DAVH7ContrG","C1DAVH7CorrG",
5  "C1DAVH7EntrG",
6  #... columns omitted here
7  "C1DTHV15EntrG","C1DTHV15VarG","C1DTHV9meanG","C1DTHV9ContrG",
8  "C1DTHV9CorrG","C1DTHV9EntrG","C1DTHV9VarG","C1Daspect","C1Delev",
9  "C1Dslope",
10 "class","x1","y1")
11  length(cam1_dry_colnames)
12
13  cd.lcx.dB.S <-
14  c("C1DAVHS","C1DAVVS","C1DPHHS","C1DTHHS","C1DTHVS","C1DTVHS","C1
15  DTVVS")
16
17  cd.l.dB.S = cd.lcx.dB.S[substr(cd.lcx.dB.S,1,4) == "C1DP"]
18  cd.c.dB.S = cd.lcx.dB.S[substr(cd.lcx.dB.S,1,4) == "C1DA"]
19  cd.x.dB.S = cd.lcx.dB.S[substr(cd.lcx.dB.S,1,4) == "C1DT"]
20
21  cd.lcx.i.S <- c("C1DAVHdivVV_linearS",
22  "C1DAVHdivVV_rdBS","C1DAVVminVH_linearS",
23  "C1DAVVmVHdivVVpVH_NDIS","C1DAVVplusVH_linearS",
24  "C1DTHHmHVdivHHpHV_NDIS","C1DTHHminHV_linearS",
25  "C1DTHHplusHV_linearS","C1DTHVdivHH_linearS",
26  "C1DTHVdivHH_rdBS","C1DTVHdivVV_linearS","C1DTVHdivVV_rdBS",
27  "C1DTVVminVH_linearS","C1DTVVmVHdivVVpVH_NDIS",
28  "C1DTVVplusVH_linearS")
29
30  cd.l.i.S = cd.lcx.i.S[substr(cd.lcx.i.S,1,4) == "C1DP"]
31  cd.c.i.S = cd.lcx.i.S[substr(cd.lcx.i.S,1,4) == "C1DA"]
32  cd.x.i.S = cd.lcx.i.S[substr(cd.lcx.i.S,1,4) == "C1DT"]
33
34  cd.lcx.t.S <- c("C1DAVH3meanS","C1DAVH3ContrS","C1DAVH3CorrS",
35  "C1DAVH3EntrS","C1DAVH3VarS","C1DAVH5meanS","C1DAVH5ContrS",
36

```

```

37 "C1DAVH5CorrS", "C1DAVH5EntrS", "C1DAVH5VarS", "C1DAVH7meanS",
38 "C1DAVH7ContrS", "C1DAVH7CorrS", "C1DAVH7EntrS",
39 "C1DAVH7VarS", "C1DAVH9meanS", "C1DAVH9ContrS", "C1DAVH9CorrS",
40 "C1DAVH9EntrS", "C1DAVH9VarS", "C1DAVV3meanS",
41 "C1DAVV3ContrS", "C1DAVV3CorrS", "C1DAVV3EntrS", "C1DAVV3VarS",
42 "C1DAVV5meanS", "C1DAVV5ContrS", "C1DAVV5CorrS",
43 "C1DAVV5EntrS", "C1DAVV5VarS", "C1DAVV7meanS", "C1DAVV7ContrS",
44 "C1DAVV7CorrS", "C1DAVV7EntrS", "C1DAVV7VarS"
45 #, ... columns omitted here
46 )
47
48   cd.l.t.S = cd.lcx.t.S[substr(cd.lcx.t.S,1,4) == "C1DP"]
49   cd.c.t.S = cd.lcx.t.S[substr(cd.lcx.t.S,1,4) == "C1DA"]
50   cd.x.t.S = cd.lcx.t.S[substr(cd.lcx.t.S,1,4) == "C1DT"]
51
52 cd.elev <- c("C1Daspect", "C1Delev", "C1Dslope")
53 cd.all.S <- c(cd.lcx.dB.S, cd.lcx.i.S, cd.lcx.t.S, cd.elev)
54
55 cd.lc.dB.S <- c(cd.l.dB.S, cd.c.dB.S)
56   cd.lx.dB.S <- c(cd.l.dB.S, cd.x.dB.S)
57   cd.cx.dB.S <- c(cd.c.dB.S, cd.x.dB.S)
58   cd.lcx.dB.S <- c(cd.l.dB.S, cd.c.dB.S, cd.x.dB.S)
59   cd.lc.i.S <- c(cd.l.i.S, cd.c.i.S)
60   cd.lx.i.S <- c(cd.l.i.S, cd.x.i.S)
61   cd.cx.i.S <- c(cd.c.i.S, cd.x.i.S)
62   cd.lcx.i.S <- c(cd.l.i.S, cd.c.i.S, cd.x.i.S)
63   cd.lc.t.S <- c(cd.l.t.S, cd.c.t.S)
64   cd.lx.t.S <- c(cd.l.t.S, cd.x.t.S)
65   cd.cx.t.S <- c(cd.c.t.S, cd.x.t.S)
66   cd.lcx.t.S <- c(cd.l.t.S, cd.c.t.S, cd.x.t.S)
67   cd.elev
68 cd.l.dBite.S <- c(cd.l.dB.S, cd.l.i.S, cd.l.t.S, cd.elev)
69 cd.c.dBite.S <- c(cd.c.dB.S, cd.c.i.S, cd.c.t.S, cd.elev)
70 cd.x.dBite.S <- c(cd.x.dB.S, cd.x.i.S, cd.x.t.S, cd.elev)
71 cd.lc.dBite.S <- c(cd.lc.dB.S, cd.lc.i.S, cd.lc.t.S, cd.elev)
72 cd.lx.dBite.S <- c(cd.lx.dB.S, cd.lx.i.S, cd.lx.t.S, cd.elev)
73 cd.cx.dBite.S <- c(cd.cx.dB.S, cd.cx.i.S, cd.cx.t.S, cd.elev)
74 cd.lcx.dBite.S <- c(cd.lcx.dB.S, cd.lcx.i.S, cd.lcx.t.S, cd.elev)
75
76 cd.l.dBi.S <- c(cd.l.dB.S, cd.l.i.S)
77 cd.c.dBi.S <- c(cd.c.dB.S, cd.c.i.S)
78 cd.x.dBi.S <- c(cd.x.dB.S, cd.x.i.S)
79 cd.lc.dBi.S <- c(cd.l.dB.S, cd.l.i.S, cd.c.dB.S, cd.c.i.S)
80 cd.lx.dBi.S <- c(cd.l.dB.S, cd.l.i.S, cd.x.dB.S, cd.x.i.S)
81 cd.cx.dBi.S <- c(cd.c.dB.S, cd.c.i.S, cd.x.dB.S, cd.x.i.S)
82 cd.lcx.dBi.S <- c(cd.l.dB.S, cd.l.i.S, cd.c.dB.S, cd.c.i.S, cd.x.dB.S, cd.x.i.S)
83

```

```

84 cd.l.dBe.S <- c(cd.l.dB.S,cd.elev)
85 cd.c.dBe.S <- c(cd.c.dB.S,cd.elev)
86 cd.x.dBe.S <- c(cd.x.dB.S,cd.elev)
87 cd.lc.dBe.S <- c(cd.l.dB.S,cd.c.dB.S,cd.elev)
88 cd.lx.dBe.S <- c(cd.l.dB.S,cd.x.dB.S,cd.elev)
89 cd.cx.dBe.S <- c(cd.c.dB.S,cd.x.dB.S,cd.elev)
90 cd.lcx.dBe.S <- c(cd.l.dB.S,cd.c.dB.S,cd.x.dB.S,cd.elev)
91
92 cd.l.dBt.S <- c(cd.l.dB.S,cd.l.t.S)
93 cd.c.dBt.S <- c(cd.c.dB.S,cd.c.t.S)
94 cd.x.dBt.S <- c(cd.x.dB.S,cd.x.t.S)
95 cd.lc.dBt.S <- c(cd.l.dB.S,cd.l.t.S,cd.c.dB.S,cd.c.t.S)
96 cd.lx.dBt.S <- c(cd.l.dB.S,cd.l.t.S,cd.x.dB.S,cd.x.t.S)
97 cd.cx.dBt.S <- c(cd.c.dB.S,cd.c.t.S,cd.x.dB.S,cd.x.t.S)
98 cd.lcx.dBt.S <- c(cd.l.dB.S,cd.l.t.S,cd.c.dB.S,cd.c.t.S,cd.x.dB.S,cd.x.t.S)
99
100 cd.l.dBit.S <- c(cd.l.dB.S,cd.l.i.S,cd.l.t.S)
101 cd.c.dBit.S <- c(cd.c.dB.S,cd.c.i.S,cd.c.t.S)
102 cd.x.dBit.S <- c(cd.x.dB.S,cd.x.i.S,cd.x.t.S)
103 cd.lc.dBit.S <- c(cd.l.dB.S,cd.l.i.S,cd.l.t.S,cd.c.dB.S,cd.c.i.S,cd.c.t.S)
104 cd.lx.dBit.S <- c(cd.l.dB.S,cd.l.i.S,cd.l.t.S,cd.x.dB.S,cd.x.i.S,cd.x.t.S)
105 cd.cx.dBit.S <- c(cd.c.dB.S,cd.c.i.S,cd.c.t.S,cd.x.dB.S,cd.x.i.S,cd.x.t.S)
106 cd.lcx.dBit.S <-
107 c(cd.l.dB.S,cd.l.i.S,cd.l.t.S,cd.c.dB.S,cd.c.i.S,cd.c.t.S,cd.x.dB.S,cd.x.i.S,cd.x.t.S)
108
109 cd3000.train <-
110 as.data.frame(read.csv("/home/bernard/sharefolder/RProjects/signature_ex
111 tract/cam1_dry_all_trv3000.csv"))
112 cd3000.test <-
113 as.data.frame(read.csv("/home/bernard/sharefolder/RProjects/signature_ex
114 tract/cam1_dry_all_test3000.csv"))
115 dim(cd3000.train)
116 #summary(cd3000.train)
117 cd3000.train[cd3000.train== -999]<-NA
118 cd3000.test[cd3000.test== -999]<-NA
119 table(cd3000.train$Cameroon1_utm_db_stack_dry.162,useNA=c("always"))
120 #?table
121 cd3000.train <- replace(cd3000.train, is.na(cd3000.train), 0)
122 cd3000.test <- replace(cd3000.test, is.na(cd3000.test), 0)
123 table(cut2(cd3000.train$Cameroon1_utm_db_stack_dry.132,g=24),
124 cd3000.train$Cameroon1_training_samples_5050_raster_clipped,
125 useNA='always')
126
127 colnames(cd3000.train) <- cam1_dry_colnames
128 dim(cd3000.train)
129 summary(cd3000.train)

```

```

130 colnames(cd3000.test) <- cam1_dry_colnames
131 summary(cd3000.test)
132 head(cd3000.train)
133 #3000 dry
134 cd3000.train$classname[cd3000.train$class == 1 ] <- "Bare"
135 cd3000.train$classname[cd3000.train$class == 3 ] <- "DenseTrees"
136 cd3000.train$classname[cd3000.train$class == 4 ] <- "ThinnerDenseTrees"
137 cd3000.train$classname[cd3000.train$class == 5 ] <- "SparseVeg"
138 cd3000.train$classname[cd3000.train$class == 6 ] <- "Water"
139 cd3000.train$classname[cd3000.train$class == 7 ] <- "Agriculture"
140 cd3000.train$classname[cd3000.train$class == 10 ] <- "Settlement"
141 table(cd3000.train$classname)
142
143 cd3000.test$classname[cd3000.test$class == 1 ] <- "Bare"
144 cd3000.test$classname[cd3000.test$class == 3 ] <- "DenseTrees"
145 cd3000.test$classname[cd3000.test$class == 4 ] <- "ThinnerDenseTrees"
146 cd3000.test$classname[cd3000.test$class == 5 ] <- "SparseVeg"
147 cd3000.test$classname[cd3000.test$class == 6 ] <- "Water"
148 cd3000.test$classname[cd3000.test$class == 7 ] <- "Agriculture"
149 cd3000.test$classname[cd3000.test$class == 10 ] <- "Settlement"
150 table(cd3000.test$classname)
151
152 cd3000.train[,c("ID","cell","x","y","class2","class","x1","y1")] <- list(NULL)
153 summary(cd3000.train)
154 cd3000.test[,c("ID","cell","x","y","class2","class","x1","y1")] <- list(NULL)
155 summary(cd3000.test)
156
157 #make the response variable a factor
158 cd3000.train$classname <- as.factor(cd3000.train$classname)
159 cd3000.test$classname <- as.factor(cd3000.test$classname)
160 table(cd3000.train$classname)
161 table(cd3000.test$classname)
162
163 cd.colnames.all = names(cd3000.train)
164 cd.colnames.all.G <- cd.lcx.dBite.G
165 cd.colnames.all.S <- cd.lcx.dBite.S
166 cd.colnames.all.G2 <- c(cd.colnames.all.G,'classname')
167 cd.colnames.all.S2 <- c(cd.colnames.all.S,'classname')
168
169 cd3000.train.G <- subset(cd3000.train,select=cd.colnames.all.G2)
170 cd3000.train.S <- subset(cd3000.train,select=cd.colnames.all.S2)
171 cd3000.test.G <- subset(cd3000.test,select=cd.colnames.all.G2)
172 cd3000.test.S <- subset(cd3000.test,select=cd.colnames.all.S2)
173
174 #select columns for each scenario
175 #train cd dB
176 cd.l.dB.S.train = subset(cd3000.train, select = c(cd.l.dB.S,'classname'))

```

```

177 cd.c.dB.S.train = subset(cd3000.train, select = c(cd.c.dB.S,'classname'))
178 cd.x.dB.S.train = subset(cd3000.train, select = c(cd.x.dB.S,'classname'))
179 cd.lc.dB.S.train = subset(cd3000.train, select = c(cd.lc.dB.S,'classname'))
180 cd.lx.dB.S.train = subset(cd3000.train, select = c(cd.lx.dB.S,'classname'))
181 cd.cx.dB.S.train = subset(cd3000.train, select = c(cd.cx.dB.S,'classname'))
182 cd.lcx.dB.S.train = subset(cd3000.train, select = c(cd.lcx.dB.S,'classname'))
183 #test cd dB
184 cd.l.dB.S.test = subset(cd3000.test, select = c(cd.l.dB.S,'classname'))
185 cd.c.dB.S.test = subset(cd3000.test, select = c(cd.c.dB.S,'classname'))
186 cd.x.dB.S.test = subset(cd3000.test, select = c(cd.x.dB.S,'classname'))
187 cd.lc.dB.S.test = subset(cd3000.test, select = c(cd.lc.dB.S,'classname'))
188 cd.lx.dB.S.test = subset(cd3000.test, select = c(cd.lx.dB.S,'classname'))
189 cd.cx.dB.S.test = subset(cd3000.test, select = c(cd.cx.dB.S,'classname'))
190 cd.lcx.dB.S.test = subset(cd3000.test, select = c(cd.lcx.dB.S,'classname'))
191 #select columns for each scenario
192 #train cd i
193 cd.l.i.S.train = subset(cd3000.train, select = c(cd.l.i.S,'classname'))
194 cd.c.i.S.train = subset(cd3000.train, select = c(cd.c.i.S,'classname'))
195 cd.x.i.S.train = subset(cd3000.train, select = c(cd.x.i.S,'classname'))
196 cd.lc.i.S.train = subset(cd3000.train, select = c(cd.lc.i.S,'classname'))
197 cd.lx.i.S.train = subset(cd3000.train, select = c(cd.lx.i.S,'classname'))
198 cd.cx.i.S.train = subset(cd3000.train, select = c(cd.cx.i.S,'classname'))
199 cd.lcx.i.S.train = subset(cd3000.train, select = c(cd.lcx.i.S,'classname'))
200 #test cd i
201 cd.l.i.S.test = subset(cd3000.test, select = c(cd.l.i.S,'classname'))
202 cd.c.i.S.test = subset(cd3000.test, select = c(cd.c.i.S,'classname'))
203 cd.x.i.S.test = subset(cd3000.test, select = c(cd.x.i.S,'classname'))
204 cd.lc.i.S.test = subset(cd3000.test, select = c(cd.lc.i.S,'classname'))
205 cd.lx.i.S.test = subset(cd3000.test, select = c(cd.lx.i.S,'classname'))
206 cd.cx.i.S.test = subset(cd3000.test, select = c(cd.cx.i.S,'classname'))
207 cd.lcx.i.S.test = subset(cd3000.test, select = c(cd.lcx.i.S,'classname'))
208 #select columns for each scenario
209 #train cd t
210 cd.l.t.S.train = subset(cd3000.train, select = c(cd.l.t.S,'classname'))
211 cd.c.t.S.train = subset(cd3000.train, select = c(cd.c.t.S,'classname'))
212 cd.x.t.S.train = subset(cd3000.train, select = c(cd.x.t.S,'classname'))
213 cd.lc.t.S.train = subset(cd3000.train, select = c(cd.lc.t.S,'classname'))
214 cd.lx.t.S.train = subset(cd3000.train, select = c(cd.lx.t.S,'classname'))
215 cd.cx.t.S.train = subset(cd3000.train, select = c(cd.cx.t.S,'classname'))
216 cd.lcx.t.S.train = subset(cd3000.train, select = c(cd.lcx.t.S,'classname'))
217 #test cd t
218 cd.l.t.S.test = subset(cd3000.test, select = c(cd.l.t.S,'classname'))
219 cd.c.t.S.test = subset(cd3000.test, select = c(cd.c.t.S,'classname'))
220 cd.x.t.S.test = subset(cd3000.test, select = c(cd.x.t.S,'classname'))
221 cd.lc.t.S.test = subset(cd3000.test, select = c(cd.lc.t.S,'classname'))
222 cd.lx.t.S.test = subset(cd3000.test, select = c(cd.lx.t.S,'classname'))
223 cd.cx.t.S.test = subset(cd3000.test, select = c(cd.cx.t.S,'classname'))

```



```

224 cd.lcx.t.S.test = subset(cd3000.test, select = c(cd.lcx.t.S,'classname'))
225 #select columns for each scenario
226 #train cd e
227 cd.l.e.S.train = subset(cd3000.train, select = c(cd.elev,'classname'))
228 #test cd e
229 cd.l.e.S.test = subset(cd3000.test, select = c(cd.elev,'classname'))
230
231 #=====#
232 #cd dBite train
233 cd.l.dBite.S.train <- subset(cd3000.train, select = c(cd.l.dBite.S,
'classname'))
234 cd.c.dBite.S.train <- subset(cd3000.train, select = c(cd.c.dBite.S,
'classname'))
235 cd.x.dBite.S.train <- subset(cd3000.train, select = c(cd.x.dBite.S,
'classname'))
236 cd.lc.dBite.S.train <- subset(cd3000.train, select = c(cd.lc.dBite.S,
'classname'))
237 cd.lx.dBite.S.train <- subset(cd3000.train, select = c(cd.lx.dBite.S,
'classname'))
238 cd.cx.dBite.S.train <- subset(cd3000.train, select = c(cd.cx.dBite.S,
'classname'))
239 cd.lcx.dBite.S.train <- subset(cd3000.train, select = c(cd.lcx.dBite.S,
'classname'))
240
241 #cd dBite test
242 cd.l.dBite.S.test <- subset(cd3000.test, select = c(cd.l.dBite.S,
'classname'))
243 cd.c.dBite.S.test <- subset(cd3000.test, select = c(cd.c.dBite.S,
'classname'))
244 cd.x.dBite.S.test <- subset(cd3000.test, select = c(cd.x.dBite.S,
'classname'))
245 cd.lc.dBite.S.test <- subset(cd3000.test, select = c(cd.lc.dBite.S,
'classname'))
246 cd.lx.dBite.S.test <- subset(cd3000.test, select = c(cd.lx.dBite.S,
'classname'))
247 cd.cx.dBite.S.test <- subset(cd3000.test, select = c(cd.cx.dBite.S,
'classname'))
248 cd.lcx.dBite.S.test <- subset(cd3000.test, select = c(cd.lcx.dBite.S,
'classname'))
249
250 #train cd dBi
251 cd.l.dBi.S.train = subset(cd3000.train, select = c(cd.l.dBi.S,'classname'))
252 cd.c.dBi.S.train = subset(cd3000.train, select = c(cd.c.dBi.S,'classname'))
253 cd.x.dBi.S.train = subset(cd3000.train, select = c(cd.x.dBi.S,'classname'))
254 cd.lc.dBi.S.train = subset(cd3000.train, select = c(cd.lc.dBi.S,'classname'))
255 cd.lx.dBi.S.train = subset(cd3000.train, select = c(cd.lx.dBi.S,'classname'))
256 cd.cx.dBi.S.train = subset(cd3000.train, select = c(cd.cx.dBi.S,'classname'))

```

```

257 cd.lcx.dBi.S.train = subset(cd3000.train, select = c(cd.lcx.dBi.S,'classname'))
258 #test cd dBi
259 cd.l.dBi.S.test = subset(cd3000.test, select = c(cd.l.dBi.S,'classname'))
260 cd.c.dBi.S.test = subset(cd3000.test, select = c(cd.c.dBi.S,'classname'))
261 cd.x.dBi.S.test = subset(cd3000.test, select = c(cd.x.dBi.S,'classname'))
262 cd.lc.dBi.S.test = subset(cd3000.test, select = c(cd.lc.dBi.S,'classname'))
263 cd.lx.dBi.S.test = subset(cd3000.test, select = c(cd.lx.dBi.S,'classname'))
264 cd.cx.dBi.S.test = subset(cd3000.test, select = c(cd.cx.dBi.S,'classname'))
265 cd.lcx.dBi.S.test = subset(cd3000.test, select = c(cd.lcx.dBi.S,'classname'))
266
267 #train cd dBe
268 cd.l.dBe.S.train = subset(cd3000.train, select = c(cd.l.dBe.S,'classname'))
269 cd.c.dBe.S.train = subset(cd3000.train, select = c(cd.c.dBe.S,'classname'))
270 cd.x.dBe.S.train = subset(cd3000.train, select = c(cd.x.dBe.S,'classname'))
271 cd.lc.dBe.S.train = subset(cd3000.train, select = c(cd.lc.dBe.S,'classname'))
272 cd.lx.dBe.S.train = subset(cd3000.train, select = c(cd.lx.dBe.S,'classname'))
273 cd.cx.dBe.S.train = subset(cd3000.train, select = c(cd.cx.dBe.S,'classname'))
274 cd.lcx.dBe.S.train = subset(cd3000.train, select = c(cd.lcx.dBe.S,'classname'))
275 #test cd dBe
276 cd.l.dBe.S.test = subset(cd3000.test, select = c(cd.l.dBe.S,'classname'))
277 cd.c.dBe.S.test = subset(cd3000.test, select = c(cd.c.dBe.S,'classname'))
278 cd.x.dBe.S.test = subset(cd3000.test, select = c(cd.x.dBe.S,'classname'))
279 cd.lc.dBe.S.test = subset(cd3000.test, select = c(cd.lc.dBe.S,'classname'))
280 cd.lx.dBe.S.test = subset(cd3000.test, select = c(cd.lx.dBe.S,'classname'))
281 cd.cx.dBe.S.test = subset(cd3000.test, select = c(cd.cx.dBe.S,'classname'))
282 cd.lcx.dBe.S.test = subset(cd3000.test, select = c(cd.lcx.dBe.S,'classname'))
283
284 #train cd dBt
285 cd.l.dBt.S.train = subset(cd3000.train, select = c(cd.l.dBt.S,'classname'))
286 cd.c.dBt.S.train = subset(cd3000.train, select = c(cd.c.dBt.S,'classname'))
287 cd.x.dBt.S.train = subset(cd3000.train, select = c(cd.x.dBt.S,'classname'))
288 cd.lc.dBt.S.train = subset(cd3000.train, select = c(cd.lc.dBt.S,'classname'))
289 cd.lx.dBt.S.train = subset(cd3000.train, select = c(cd.lx.dBt.S,'classname'))
290 cd.cx.dBt.S.train = subset(cd3000.train, select = c(cd.cx.dBt.S,'classname'))
291 cd.lcx.dBt.S.train = subset(cd3000.train, select = c(cd.lcx.dBt.S,'classname'))
292 #test cd dBt
293 cd.l.dBt.S.test = subset(cd3000.test, select = c(cd.l.dBt.S,'classname'))
294 cd.c.dBt.S.test = subset(cd3000.test, select = c(cd.c.dBt.S,'classname'))
295 cd.x.dBt.S.test = subset(cd3000.test, select = c(cd.x.dBt.S,'classname'))
296 cd.lc.dBt.S.test = subset(cd3000.test, select = c(cd.lc.dBt.S,'classname'))
297 cd.lx.dBt.S.test = subset(cd3000.test, select = c(cd.lx.dBt.S,'classname'))
298 cd.cx.dBt.S.test = subset(cd3000.test, select = c(cd.cx.dBt.S,'classname'))
299 cd.lcx.dBt.S.test = subset(cd3000.test, select = c(cd.lcx.dBt.S,'classname'))
300
301 #train cd dBit
302 cd.l.dBit.S.train = subset(cd3000.train, select = c(cd.l.dBit.S,'classname'))
303 cd.c.dBit.S.train = subset(cd3000.train, select = c(cd.c.dBit.S,'classname'))

```

```

304 cd.x.dBit.S.train = subset(cd3000.train, select = c(cd.x.dBit.S,'classname'))
305 cd.lc.dBit.S.train = subset(cd3000.train, select = c(cd.lc.dBit.S,'classname'))
306 cd.lx.dBit.S.train = subset(cd3000.train, select = c(cd.lx.dBit.S,'classname'))
307 cd.cx.dBit.S.train = subset(cd3000.train, select = c(cd.cx.dBit.S,'classname'))
308 cd.lcx.dBit.S.train = subset(cd3000.train, select = c(cd.lcx.dBit.S,'classname'))
309 #test cd dBit
310 cd.l.dBit.S.test = subset(cd3000.test, select = c(cd.l.dBit.S,'classname'))
311 cd.c.dBit.S.test = subset(cd3000.test, select = c(cd.c.dBit.S,'classname'))
312 cd.x.dBit.S.test = subset(cd3000.test, select = c(cd.x.dBit.S,'classname'))
313 cd.lc.dBit.S.test = subset(cd3000.test, select = c(cd.lc.dBit.S,'classname'))
314 cd.lx.dBit.S.test = subset(cd3000.test, select = c(cd.lx.dBit.S,'classname'))
315 cd.cx.dBit.S.test = subset(cd3000.test, select = c(cd.cx.dBit.S,'classname'))
316 cd.lcx.dBit.S.test = subset(cd3000.test, select = c(cd.lcx.dBit.S,'classname'))

```

### A3.2.2. R functions for modelling

*Script A3.4 model\_and\_test\_rf function and application example in R*

```

1  model_and_test_rf <-function(training,test) {
2    rf.model <- randomForest(classname~., data=training,
3 importance=TRUE, keep.forest=TRUE,ntree=500)
4    rf.test <- predict(rf.model, newdata=test)
5    rf.confusion <- confusionMatrix(rf.test,test$classname,
6 dnn=c('Prediction','Reference')) #from caret package
7    return(rf.confusion)
8  }
9
10 #apply the model_and_test_rf_model
11 cd.l.dBite.S.conf <- model_and_test_rf(cd.l.dBite.S.train,cd.l.dBite.S.test)
12 cd.c.dBite.S.conf <- model_and_test_rf(cd.c.dBite.S.train,cd.c.dBite.S.test)
13 cd.x.dBite.S.conf <- model_and_test_rf(cd.x.dBite.S.train,cd.x.dBite.S.test)
14 cd.lc.dBite.S.conf <- model_and_test_rf(cd.lc.dBite.S.train,cd.lc.dBite.S.test)
15 cd.lx.dBite.S.conf <- model_and_test_rf(cd.lx.dBite.S.train,cd.lx.dBite.S.test)
16 cd.cx.dBite.S.conf <- model_and_test_rf(cd.cx.dBite.S.train,cd.cx.dBite.S.test)
17 cd.lcx.dBite.S.conf <-
18 model_and_test_rf(cd.lcx.dBite.S.train,cd.lcx.dBite.S.test)
19 save(cd.l.dBite.S.conf,cd.c.dBite.S.conf,cd.x.dBite.S.conf,cd.lc.dBite.S.conf,cd.
20 lx.dBite.S.conf,cd.cx.dBite.S.conf,cd.lcx.dBite.S.conf,
21 file="cd_dBite_S_confusion.rda")

```

*Script A3.5 model\_rf function in R*

```
1 model_rf <-function(training) {
2   rf.model <- randomForest(classname~., data=training, importance=TRUE,
3 keep.forest=TRUE,ntree=500)
4   #rf.test <- predict(rf.model, newdata=test)
5   #rf.confusion <- confusionMatrix(rf.test,test$classname,
6 dnn=c('Prediction','Reference')) #from caret package
7   return(rf.model)
8 }
```

*Script A3.6 varimp\_rf function and application example in R*

```
1 varimp_rf <- function(training, test) {
2   rf.model <- randomForest(classname~., data=training,
3 importance=TRUE, keep.forest=TRUE,ntree=500)
4   rf.test <- predict(rf.model, newdata=test)
5   rf.confusion <- confusionMatrix(rf.test,test$classname,
6 dnn=c('Prediction','Reference')) #from caret package
7   imp.rf.model <-importance(rf.model)
8   #varImpPlot(rf.model)
9   #write.csv(imp.rf.cd100,paste(training,"_imp.csv")
10  imp.rf.model.ordered <-
11  imp.rf.model[order(imp.rf.model[,dim(imp.rf.model)[2]], decreasing=TRUE),]
12  imp.rf.model.ordered <- as.data.frame(imp.rf.model.ordered)
13  imp.rf.model.ordered$variables <- rownames(imp.rf.model.ordered)
14  #cdl.varimp.rf.ordered$variables[1]
15  return(imp.rf.model.ordered)
16  }
17  #importance(obj, type=1, scale=FALSE)
18  #varimp(obj)
19
20  #apply the varimp_rf function
21  cd.l.dBite.S.varimp.rf <- varimp_rf(cd.l.dBite.S.train,cd.l.dBite.S.test)
22  cd.c.dBite.S.varimp.rf <- varimp_rf(cd.c.dBite.S.train,cd.c.dBite.S.test)
23  cd.x.dBite.S.varimp.rf <- varimp_rf(cd.x.dBite.S.train,cd.x.dBite.S.test)
24  cd.lc.dBite.S.varimp.rf <- varimp_rf(cd.lc.dBite.S.train,cd.lc.dBite.S.test)
25  cd.lx.dBite.S.varimp.rf <- varimp_rf(cd.lx.dBite.S.train,cd.lx.dBite.S.test)
26  cd.cx.dBite.S.varimp.rf <- varimp_rf(cd.cx.dBite.S.train,cd.cx.dBite.S.test)
27  cd.lcx.dBite.S.varimp.rf <- varimp_rf(cd.lcx.dBite.S.train,cd.lcx.dBite.S.test)
28 }
```

Script A3.7 *varimp\_rf\_unscaled* function and application example in R

```
1 varimp_rf_unscaled <- function(training, test) {
2   rf.model <- randomForest(classname~., data=training, importance=TRUE,
3 keep.forest=TRUE, ntree=500)
4   rf.test <- predict(rf.model, newdata=test)
5   rf.confusion <- confusionMatrix(rf.test, test$classname,
6 dnn=c('Prediction', 'Reference')) #from caret package
7   imp.rf.model <- importance(rf.model, scale=FALSE)
8   #varImpPlot(rf.model)
9   #write.csv(imp.rf.cd100, paste(training, "_imp.csv"))
10  imp.rf.model.ordered <-
11  imp.rf.model[order(imp.rf.model[, dim(imp.rf.model)[2]-1],
12 decreasing=TRUE),]
13  imp.rf.model.ordered <- as.data.frame(imp.rf.model.ordered)
14  imp.rf.model.ordered$variables <- rownames(imp.rf.model.ordered)
15  #cdl.varimp.rf.ordered$variables[1]
16  return(imp.rf.model.ordered)
17 }
18
19 #apply the varimp_rf_unscaled function
20 cd.lcx.dBite.S.varimp.rf.uns <-
21 varimp_rf_unscaled(cd.lcx.dBite.S.train, cd.lcx.dBite.S.test)
```

Script A3.8 *model\_and\_test\_svm* function in R

```
1 # SVM model
2 model_and_test_svm <- function(training, test) {
3   svm.model <- svm(classname~., data=training)
4   svm.test <- predict(svm.model, newdata=test)
5   svm.confusion <- confusionMatrix(svm.test, test$classname,
6 dnn=c('Prediction', 'Reference')) #from caret package
7   return(svm.confusion)
8 }
```

Script A3.9 *model\_svm* function in R

```
1 model_svm <- function(training) {
2   svm.model <- svm(classname~., data=training)
3   #svm.test <- predict(svm.model, newdata=test)
4   #svm.confusion <- confusionMatrix(svm.test, test$classname,
5 dnn=c('Prediction', 'Reference')) #from caret package
6   return(svm.model)
7 }
```

```

1 rf.topX.conf <- function(varimp_all,train,test,mda_mdg_ind,top_x){
2     if (mda_mdg_ind == 'MDG') {
3         varimp_all_ordered <- varimp_all[order(varimp_all[,8],
4 decreasing=TRUE),]
5         varimp_all_ordered <- as.data.frame(varimp_all_ordered)
6     }
7     if (mda_mdg_ind == 'MDA') {
8         varimp_all_ordered <- varimp_all[order(varimp_all[,7],
9 decreasing=TRUE),]
10        varimp_all_ordered <- as.data.frame(varimp_all_ordered)
11    }
12    #select top x (20) variables
13    varimp_all_ordered_topX <-
14    varimp_all_ordered$variables[1:top_x]
15    varimp_all_ordered_topX_with_classname <-
16    c(varimp_all_ordered_topX,"classname")
17    train.topX <-
18    subset(train,select=varimp_all_ordered_topX_with_classname)
19    test.topX <-
20    subset(test,select=varimp_all_ordered_topX_with_classname)
21    train.topX.conf <- model_and_test_rf(train.topX,test.topX)
22    return(train.topX.conf)
23 }
24
25 cd.dBit.rf.conf.mdg.top1 <-
26 rf.topX.conf(dBit_cd_uns,cd.lcx.dBite.S.train,cd.lcx.dBite.S.test,'MDG',1)
27 cd.dBit.rf.conf.mdg.top2 <-
28 rf.topX.conf(dBit_cd_uns,cd.lcx.dBite.S.train,cd.lcx.dBite.S.test,'MDG',2)
29 cd.dBit.rf.conf.mdg.top3 <-
30 rf.topX.conf(dBit_cd_uns,cd.lcx.dBite.S.train,cd.lcx.dBite.S.test,'MDG',3)
31 cd.dBit.rf.conf.mdg.top4 <-
32 rf.topX.conf(dBit_cd_uns,cd.lcx.dBite.S.train,cd.lcx.dBite.S.test,'MDG',4)
33 cd.dBit.rf.conf.mdg.top5 <-
34 rf.topX.conf(dBit_cd_uns,cd.lcx.dBite.S.train,cd.lcx.dBite.S.test,'MDG',5)

```

```

1 rf.topX.model <- function(varimp_all,train,test,mda_mdg_ind,top_x){
2     if (mda_mdg_ind == 'MDG') {
3         varimp_all_ordered <- varimp_all[order(varimp_all[,8],
4 decreasing=TRUE),]
5         varimp_all_ordered <- as.data.frame(varimp_all_ordered)
6     }

```

```

7         if (mda_mdg_ind == 'MDA') {
8             varimp_all_ordered <- varimp_all[order(varimp_all[,7],
9 decreasing=TRUE),]
10            varimp_all_ordered <- as.data.frame(varimp_all_ordered)
11        }
12        #select top x (20) variables
13        varimp_all_ordered_topX <-
14 varimp_all_ordered$variables[1:top_x]
15        varimp_all_ordered_topX_with_classname <-
16 c(varimp_all_ordered_topX,"classname")
17        train.topX <-
18 subset(train,select=varimp_all_ordered_topX_with_classname)
19        test.topX <-
20 subset(test,select=varimp_all_ordered_topX_with_classname)
21        train.topX.model <- model_rf(train.topX)
22        return(train.topX.model)
23    }

```

*Script A3.12 rf.topX.varimp function in R*

```

1 rf.topX.varimp <- function(varimp_all,train,test,mda_mdg_ind,top_x){
2     if (mda_mdg_ind == 'MDG') {
3         varimp_all_ordered <- varimp_all[order(varimp_all[,8],
4 decreasing=TRUE),]
5         varimp_all_ordered <- as.data.frame(varimp_all_ordered)
6     }
7     if (mda_mdg_ind == 'MDA') {
8         varimp_all_ordered <- varimp_all[order(varimp_all[,7],
9 decreasing=TRUE),]
10        varimp_all_ordered <- as.data.frame(varimp_all_ordered)
11    }
12    #select top x (20) variables
13    varimp_all_ordered_topX <-
14 varimp_all_ordered$variables[1:top_x]
15    varimp_all_ordered_topX_with_classname <-
16 c(varimp_all_ordered_topX,"classname")
17    train.topX <-
18 subset(train,select=varimp_all_ordered_topX_with_classname)
19    test.topX <-
20 subset(test,select=varimp_all_ordered_topX_with_classname)
21    topX.varimp <- varimp_rf_unscaled(train.topX,test.topX)
22    return(topX.varimp)
23 }

```

```

1 svm.topX.conf <- function(varimp_all,train,test,mda_mdg_ind,top_x){
2     if (mda_mdg_ind == 'MDG') {
3         varimp_all_ordered <- varimp_all[order(varimp_all[,8],
4 decreasing=TRUE),]
5         varimp_all_ordered <- as.data.frame(varimp_all_ordered)
6     }
7     if (mda_mdg_ind == 'MDA') {
8         varimp_all_ordered <- varimp_all[order(varimp_all[,7],
9 decreasing=TRUE),]
10        varimp_all_ordered <- as.data.frame(varimp_all_ordered)
11    }
12    #select top x (20) variables
13    varimp_all_ordered_topX <-
14    varimp_all_ordered$variables[1:top_x]
15    varimp_all_ordered_topX_with_classname <-
16    c(varimp_all_ordered_topX,"classname")
17    train.topX <-
18    subset(train,select=varimp_all_ordered_topX_with_classname)
19    test.topX <-
20    subset(test,select=varimp_all_ordered_topX_with_classname)
21    # svm top20 modelling
22    svm.topX.conf <- model_and_test_svm(train.topX,test.topX)
23    return(svm.topX.conf)
24 }
25
26 cd.dBit.svm.conf.mdg.top1 <-
27 svm.topX.conf(dBit_cd_uns,cd.lcx.dBite.S.train,cd.lcx.dBite.S.test,'MDG',1)
28 cd.dBit.svm.conf.mdg.top2 <-
29 svm.topX.conf(dBit_cd_uns,cd.lcx.dBite.S.train,cd.lcx.dBite.S.test,'MDG',2)
30 cd.dBit.svm.conf.mdg.top3 <-
31 svm.topX.conf(dBit_cd_uns,cd.lcx.dBite.S.train,cd.lcx.dBite.S.test,'MDG',3)
32 cd.dBit.svm.conf.mdg.top4 <-
33 svm.topX.conf(dBit_cd_uns,cd.lcx.dBite.S.train,cd.lcx.dBite.S.test,'MDG',4)
34 cd.dBit.svm.conf.mdg.top5 <-
35 svm.topX.conf(dBit_cd_uns,cd.lcx.dBite.S.train,cd.lcx.dBite.S.test,'MDG',5)

```

```

1 svm.topX.model <- function(varimp_all,train,test,mda_mdg_ind,top_x){
2     if (mda_mdg_ind == 'MDG') {
3         varimp_all_ordered <- varimp_all[order(varimp_all[,8],
4 decreasing=TRUE),]
5         varimp_all_ordered <- as.data.frame(varimp_all_ordered)

```



```

6         }
7         if (mda_mdg_ind == 'MDA') {
8             varimp_all_ordered <- varimp_all[order(varimp_all[,7],
9 decreasing=TRUE),]
10            varimp_all_ordered <- as.data.frame(varimp_all_ordered)
11        }
12        #select top x (20) variables
13        varimp_all_ordered_topX <-
14        varimp_all_ordered$variables[1:top_x]
15        varimp_all_ordered_topX_with_classname <-
16        c(varimp_all_ordered_topX,"classname")
17        train.topX <-
18        subset(train,select=varimp_all_ordered_topX_with_classname)
19        test.topX <-
20        subset(test,select=varimp_all_ordered_topX_with_classname)
21        svm.topX.model <- model_svm(train.topX)
22        return(svm.topX.model)
23    }

```

Script A3.15 apply\_KL\_divergence function

```

1  apply_KL_divergence <- function(dB_training, class1_vector, class2_vector,
2                                iterations,colname1,colname2) {
3      library(FNN)
4      temp_subset <- subset(dB_training,
5                             subset = classname %in% c(class1_vector[1]))
6      class1_colnames <- colnames(temp_subset)
7      a <- length(class1_vector)
8      b <- length(class2_vector)
9      e <- length(class1_colnames)
10     d <- a * b * (e-1)
11     kld.df <- as.data.frame(matrix(nrow = d, ncol = 4))
12     for (n in 1:(length(class1_colnames)-1)) {
13         for (i in 1:length(class1_vector)) {
14             class1_subset <- subset(dB_training,
15                                    subset = classname %in% c(class1_vector[i]))
16             #class1_colnames <- colnames(class1_subset)
17             for (j in 1:length(class2_vector)) {
18                 class2_subset <- subset(dB_training,
19                                        subset = classname %in% c(class2_vector[j]))
20                 kld <- KL.divergence(as.matrix(class1_subset[n]),
21                                     as.matrix(class2_subset[n]), k = iterations, algorithm=c("kd_tree"))
22                 #a
23                 #kld.df

```

```

24     m = ((n-1)*(length(class1_vector)*length(class2_vector)))+((i-
25 1)*length(class1_vector))+j)
26     kld.df[m,1] <- class1_vector[i]
27     kld.df[m,2] <- class2_vector[j]
28     kld.df[m,3] <- class1_colnames[n]
29     kld.df[m,4] <- round(mean(kld[is.finite(kld)]),2)
30   }
31 }
32 }
33 print(n)
34 print(i)
35 print(j)
36 kld.df$V1 <- as.factor(kld.df$V1)
37 kld.df$V2 <- as.factor(kld.df$V2)
38 colnames(kld.df) <- c(colname1,colname2,"Band","KLD")
39 return(kld.df)
40 }
41
42 # apply_KL_divergence
43 td_class_vector <- c("TD_Agriculture", "TD_Bare", "TD_DenseTrees",
44                     "TD_Settlement", "TD_SparseVeg")
45 tw_class_vector <- c("TW_Agriculture", "TW_Bare", "TW_DenseTrees",
46                     "TW_Settlement", "TW_SparseVeg")
47 kld.td.tw.dB <- apply_KL_divergence(td.tw.lcx.dB.rf.mdg.30b[[1]],
48                                   td_class_vector, tw_class_vector, 2000,
49                                   "TD_Train", "TW_Train")
50 kld.td.tw.t <- apply_KL_divergence(td.tw.lcx.t.rf.mdg.30b[[1]],
51                                   td_class_vector, tw_class_vector, 2000, "TD_Train", "TW_Train")

```

### A3.2.3. R functions for plotting graphs

The multiplot function is reproduced in Script A3.16 for reference from [www.cookbook-r.com](http://www.cookbook-r.com).

*Script A3.16 multiplot function from [www.cookbook-r.com](http://www.cookbook-r.com)*

```

1 #multiplot function reference:
2 #http://www.cookbook-
3 r.com/Graphs/Multiple_graphs_on_one_page_(ggplot2)/
4 # Multiple plot function
5 # ggplot objects can be passed in ..., or to plotlist (as a list of ggplot objects)
6 # - cols: Number of columns in layout

```

```

7 # - layout: A matrix specifying the layout. If present, 'cols' is ignored.
8 #
9 # If the layout is something like matrix(c(1,2,3,3), nrow=2, byrow=TRUE),
10 # then plot 1 will go in the upper left, 2 will go in the upper right, and
11 # 3 will go all the way across the bottom.
12 #
13 multiplot <- function(..., plotlist=NULL, file, cols=1, layout=NULL) {
14   library(grid)
15   # Make a list from the ... arguments and plotlist
16   plots <- c(list(...), plotlist)
17
18   numPlots = length(plots)
19
20   # If layout is NULL, then use 'cols' to determine layout
21   if (is.null(layout)) {
22     # Make the panel
23     # ncol: Number of columns of plots
24     # nrow: Number of rows needed, calculated from # of cols
25     layout <- matrix(seq(1, cols * ceiling(numPlots/cols)),
26                       ncol = cols, nrow = ceiling(numPlots/cols))
27   }
28
29   if (numPlots==1) {
30     print(plots[[1]])
31
32   } else {
33     # Set up the page
34     grid.newpage()
35     pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))
36
37     # Make each plot, in the correct location
38     for (i in 1:numPlots) {
39       # Get the i,j matrix positions of the regions that contain this subplot
40       matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))
41
42       print(plots[[i]], vp = viewport(layout.pos.row = matchidx$row,
43                                       layout.pos.col = matchidx$col))
44     }
45   }
46 }

```

The `plot_overall_accuracy_with_class_accuracy` function in Script A3.17 can be modified to include only selected input variables, that are with no X-band, no C-band and for only L-band data.

*Script A3.17 `plot_overall_accuracy_with_class_accuracy` function and application example.*

```

1 plot_overall_accuracy_with_class_accuracy <-
2 function(inputl.conf,inputc.conf,inputx.conf,inputlc.conf,
3           inputlx.conf,inputcx.conf,inputlcx.conf,
4           site,season,GS,vardescrip,legendYN) {
5   # inputl.conf <- cd.l.dBite.S.conf
6   # inputc.conf <- cd.c.dBite.S.conf
7   # inputx.conf <- cd.x.dBite.S.conf
8   # inputlc.conf <- cd.lc.dBite.S.conf
9   # inputlx.conf <- cd.lx.dBite.S.conf
10  # inputcx.conf <- cd.cx.dBite.S.conf
11  # inputlcx.conf <- cd.lcx.dBite.S.conf
12  # site <- "Cameroon"
13  # season <- "Dry"
14  # GS <- "S"
15  # vardescrip <- "All variables"
16  # legendYN <- "Y"
17
18  input.byClass <- rbind(inputl.conf$byClass[1],
19                        inputc.conf$byClass[1],
20                        inputx.conf$byClass[1],
21                        inputlc.conf$byClass[1],
22                        inputlx.conf$byClass[1],
23                        inputcx.conf$byClass[1],
24                        inputlcx.conf$byClass[1])
25
26  input.overall <- c(inputl.conf$overall[1],
27                  inputc.conf$overall[1],
28                  inputx.conf$overall[1],
29                  inputlc.conf$overall[1],
30                  inputlx.conf$overall[1],
31                  inputcx.conf$overall[1],
32                  inputlcx.conf$overall[1])
33  #input.overall
34  lcx <- c('L','C','X','LC','LX','CX','LCX')
35
36  #input.overall.new <- input.overall[order(input.overall)]
37  input.overall2 <- cbind(lcx,input.overall)
38  input.overall.order <- input.overall2[order(input.overall)]

```

```

39 input.byClass <- round(input.byClass * 100,2)
40 input.overall <- round(input.overall * 100,2)
41 input.byClass1 <- cbind(input.byClass,input.overall)
42
43 if (site == 'Cameroon') {
44   site_columnnames <- c("BareSoil", "DenseTrees", "Settlement",
45     "SparseVeg", "ThinnerDenseTrees","Water","Overall")
46 }
47 if (site == 'Tanzania') {
48   site_columnnames <- c("Agriculture","BareSoil", "DenseTrees",
49     "Settlement", "SparseVeg","ThinnerDenseTrees","Overall")
50 }
51 #input.byClass1
52 colnames(input.byClass1) <- site_columnnames
53 rownames(input.byClass1) <- lcx
54 input.byClass2 <- cbind(lcx,as.data.frame(input.byClass1))
55 input.byClass.long <- melt(input.byClass2,id.vars=c("lcx"))
56 input.byClass.long$lcx <- as.factor(input.byClass.long$lcx)
57 #input.byClass.long$lcx <- ordered(input.byClass.long$lcx,
58   levels = c('L','C','X','LC','LX','CX','LCX'))
59 input.byClass.long$lcx <- ordered(input.byClass.long$lcx,
60   levels =c(input.overall.order))
61
62 if (site == 'Cameroon') {
63   site_colours <- c("#FF5500", "#007139", "#FF0000", "#FB3AFF",
64     "#00D362", "#0000FF", "#000000")
65 }
66 if (site == 'Tanzania') {
67   site_colours <- c("#55AAFF", "#FF5500", "#007139",
68     "#FF0000", "#FB3AFF", "#00D362", "#000000")
69 }
70
71 if (GS == "S" && legendYN == "Y") {
72   pinput <- ggplot(input.byClass.long,
73     aes(x=lcx, y=value, colour=variable, group=variable)) +
74     geom_line(size=1.2) +
75     ggtitle(bquote(. (site) ~ . (season) )) +
76     ylab("Classification Accuracy on Test Data") +
77     xlab("") +
78     scale_color_manual(values=site_colours) + ylim(0,100) +
79     annotate("text",x=lcx,y=(input.overall+5),
80       label=paste0(round(input.overall,"%"),size=8) +
81 theme(plot.title=element_text(size=rel(3.0),lineheight=2.0,vjust=2),
82   legend.title=element_text(size=rel(1.5),color="white"),
83   legend.text=element_text(size=rel(2.0),lineheight=2.0),
84   legend.key.height=unit(1,"cm"),
85   axis.title=element_text(size=rel(1.8)),

```

```

86     axis.text=element_text(size=rel(1.8),colour='black'),
87
88     axis.ticks=element_line(size=rel(2.0)),strip.text=element_text(size=rel(2.0)),
89     plot.background=element_rect(fill = "transparent", colour = NA)
90   )
91   } else if (GS == "G" && legendYN == "Y") {
92     pinplot <- ggplot(input.byClass.long, aes(x=lcx, y=value, colour=variable,
93     group=variable)) + geom_line(size=1.2) +
94     ggtitle(bquote(. (site) ~ . (season) )) +
95     ylab("Classification Accuracy on Test Data") +
96     xlab("") +
97     scale_color_manual(values=site_colours) + ylim(0,100) +
98     annotate("text",x=lcx,y=(input.overall+5),
99     label=paste0(round(input.overall,"%"),size=8) +
100   theme(plot.title=element_text(size=rel(3.0),lineheight=2.0,vjust=2),
101     legend.title=element_text(size=rel(1.5),color="white"),
102     legend.text=element_text(size=rel(2.0),lineheight=2.0),
103     legend.key.height=unit(1,"cm"),
104     axis.title=element_text(size=rel(1.8)),
105     axis.text=element_text(size=rel(1.8),colour='black'),
106
107     axis.ticks=element_line(size=rel(2.0)),strip.text=element_text(size=rel(2.0)),
108     plot.background=element_rect(fill = "transparent", colour = NA)
109   )
110   } else if (GS == "S" && legendYN == "N") {
111     pinplot <- ggplot(input.byClass.long, aes(x=lcx, y=value, colour=variable,
112     group=variable)) + geom_line(size=1.2) +
113     ggtitle(bquote(. (site) ~ . (season) )) +
114     ylab("Classification Accuracy on Test Data") +
115     xlab("") +
116     scale_color_manual(values=site_colours) + ylim(0,100) +
117     annotate("text",x=lcx,y=(input.overall+5),
118     label=paste0(round(input.overall,"%"),size=8) +
119   theme(plot.title=element_text(size=rel(3.0),lineheight=2.0,vjust=2),
120     legend.title=element_text(size=rel(1.5),color="white"),
121     legend.text=element_text(size=rel(2.0),lineheight=2.0),
122     legend.key.height=unit(1,"cm"),
123     legend.position="none",
124     axis.title=element_text(size=rel(1.8)),
125     axis.text=element_text(size=rel(1.8),colour='black'),
126     axis.ticks=element_line(size=rel(2.0)),strip.text=element_text(size=rel(2.0)),
127     plot.background=element_rect(fill = "transparent", colour = NA)
128   )
129   } else if (GS == "G" && legendYN == "N") {
130     pinplot <- ggplot(input.byClass.long,
131     aes(x=lcx, y=value, colour=variable, group=variable)) +
132     geom_line(size=1.2) +

```

```

133     ggtitle(bquote(.(site) ~ .(season) )) +
134     ylab("Classification Accuracy on Test Data") +
135     xlab("") +
136     scale_color_manual(values=site_colours) + ylim(0,100) +
137     annotate("text",x=lcx,y=(input.overall+5),
138     label=paste0(round(input.overall,"%"),size=8) +
139 theme(plot.title=element_text(size=rel(3.0),lineheight=2.0,vjust=2),
140     legend.title=element_text(size=rel(1.5),color="white"),
141     legend.text=element_text(size=rel(2.0),lineheight=2.0),
142     legend.key.height=unit(1,"cm"),
143     legend.position="none",
144     axis.title=element_text(size=rel(1.8)),
145     axis.text=element_text(size=rel(1.8),colour='black'),
146 axis.ticks=element_line(size=rel(2.0)),strip.text=element_text(size=rel(2.0)),
147     plot.background=element_rect(fill = "transparent", colour = NA)
148   )
149 }
150 return(pinput)
151 }
152
153 #all4 S dBite - with legend
154 png("all4_cd_cw_td_tw_S_dBite_overallaccuracy_legendY900h.png",height=
155 900,width=1980, bg="transparent")
156 p1 <- plot_overall_accucary_with_class_accuracy(cd.l.dBite.S.conf,
157 cd.c.dBite.S.conf, cd.x.dBite.S.conf, cd.lc.dBite.S.conf, cd.lx.dBite.S.conf,
158 cd.cx.dBite.S.conf, cd.lcx.dBite.S.conf, "Cameroon",
159 "dry", "S", "All variables", "Y")
160 p2 <- plot_overall_accucary_with_class_accuracy(cw.l.dBite.S.conf,
161 cw.c.dBite.S.conf, cw.x.dBite.S.conf, cw.lc.dBite.S.conf, cw.lx.dBite.S.conf,
162 cw.cx.dBite.S.conf, cw.lcx.dBite.S.conf, "Cameroon", "wet", "S",
163 "All variables", "Y")
164 p3 <- plot_overall_accucary_with_class_accuracy(td.l.dBite.S.conf,
165 td.c.dBite.S.conf, td.x.dBite.S.conf, td.lc.dBite.S.conf,
166 td.lx.dBite.S.conf, td.cx.dBite.S.conf, td.lcx.dBite.S.conf,
167 "Tanzania", "dry", "S", "All variables", "Y")
168 p4 <- plot_overall_accucary_with_class_accuracy_no_x(tw.l.dBite.S.conf,
169 tw.c.dBite.S.conf, tw.lc.dBite.S.conf, "Tanzania", "wet", "S", "All variables", "Y")
170 multiplot(p1,p3,p2,p4,cols=2)
171 dev.off()

```

Script A3.18 *plot\_overall\_accucary\_with\_class\_accuracy\_elevation\_only* function in R.

```

1 plot_overall_accucary_with_class_accuracy_elevation_only <-
2 function(inpute.conf, site,season, GS,vardescrip,legendYN) {
3   input.byClass <- rbind(#inputl.conf$byClass[,1],
4     inpute.conf$byClass[,1]
5     #,inputx.conf$byClass[,1],

```

```

6      #inputlc.conf$byClass[,1]
7      #,inputlx.conf$byClass[,1],
8      #inputcx.conf$byClass[,1],
9      #inputlcx.conf$byClass[,1]
10 )
11
12 input.overall <- c(#inputl.conf$overall[1],
13   inpute.conf$overall[1]
14   #,inputx.conf$overall[1],
15   #inputlc.conf$overall[1]
16   #,inputlx.conf$overall[1],
17   #inputcx.conf$overall[1],
18   #inputlcx.conf$overall[1]
19 )
20 input.overall
21 #lcx <- c('L','C','X','LC','LX','CX','LCX')
22 lcx <- c('SRTM')
23
24 input.overall2 <- cbind(lcx,input.overall)
25
26 input.overall.order <- input.overall2[order(input.overall)]
27 input.byClass <- round(input.byClass * 100,2)
28 input.overall <- round(input.overall * 100,2)
29 input.byClass1 <- cbind(input.byClass,input.overall)
30
31 if (site == 'Cameroon') {
32   site_columnnames <- c("BareSoil", "DenseTrees", "Settlement",
33     "SparseVeg", "ThinnerDenseTrees","Water","Overall")
34 }
35 if (site == 'Tanzania') {
36   site_columnnames <- c("Agriculture","BareSoil", "DenseTrees",
37     "Settlement","SparseVeg","ThinnerDenseTrees","Overall")
38 }
39 #input.byClass1
40 colnames(input.byClass1) <- site_columnnames
41 rownames(input.byClass1) <- lcx
42 input.byClass2 <- cbind(lcx,as.data.frame(input.byClass1))
43 input.byClass.long <- melt(input.byClass2,id.vars=c("lcx"))
44 input.byClass.long$lcx <- as.factor(input.byClass.long$lcx)
45 #input.byClass.long$lcx <- ordered(input.byClass.long$lcx,
46 #   levels = c('L','C','X','LC','LX','CX','LCX'))
47 input.byClass.long$lcx <- ordered(input.byClass.long$lcx,
48   levels =c(input.overall.order))
49
50 if (site == 'Cameroon') {
51   site_colours <- c("#FF5500", "#007139", "#FF0000", "#FB3AFF",
52     "#00D362", "#0000FF", "#000000")

```



```

53 }
54 if (site == 'Tanzania') {
55   site_colours <- c("#55AAFF", "#FF5500", "#007139", "#FF0000",
56                     "#FB3AFF", "#00D362", "#000000")
57 }
58 if (legendYN == "Y") {
59   pinput <- ggplot(input.byClass.long, aes(x=lcx, y=value, colour=variable,
60 group=variable)) + geom_point(size=4.0) +
61     ggtitle(bquote(. (site) ~ . (season) )) +
62     ylab("Classification Accuracy on Test Data") +
63     xlab("") +
64     scale_color_manual(values=site_colours) + ylim(0,100) +
65     annotate("text",x=lcx,y=(input.overall+5),
66 label=paste0(round(input.overall),"%"),size=8) +
67 theme(plot.title=element_text(size=rel(3.0),lineheight=2.0,vjust=2),
68       legend.title=element_text(size=rel(1.5),color="white"),
69       legend.text=element_text(size=rel(2.0),lineheight=2.0),
70       legend.key.height=unit(1,"cm"),
71       axis.title=element_text(size=rel(1.8)),
72       axis.text=element_text(size=rel(1.8),colour='black'),
73 axis.ticks=element_line(size=rel(2.0)),strip.text=element_text(size=rel(2.0)),
74       plot.background=element_rect(fill = "transparent", colour = NA)
75   )
76 }
77 if (legendYN == "N") {
78   pinput <- ggplot(input.byClass.long, aes(x=lcx, y=value, colour=variable,
79 group=variable)) + geom_point(size=4.0) +
80     ggtitle(bquote(. (site) ~ . (season) )) +
81     ylab("Classification Accuracy on Test Data") +
82     xlab("") +
83     scale_color_manual(values=site_colours) + ylim(0,100) +
84     annotate("text",x=lcx,y=(input.overall+5),label=paste0(round(input.overall),"
85 %"),size=8) +
86 theme(plot.title=element_text(size=rel(3.0),lineheight=2.0,vjust=2),
87       legend.title=element_text(size=rel(1.5),color="white"),
88       legend.text=element_text(size=rel(2.0),lineheight=2.0),
89       legend.key.height=unit(1,"cm"),
90       legend.position="none", #for legendYN="N"
91       axis.title=element_text(size=rel(1.8)),
92       axis.text=element_text(size=rel(1.8),colour='black'),
93 axis.ticks=element_line(size=rel(2.0)),strip.text=element_text(size=rel(2.0)),
94       plot.background=element_rect(fill = "transparent", colour = NA)
95   )
96 }
97 return(pinput)
98 }
99

```

The variable were named P for PALSAR, A for ASAR, and T for TerraSAR-X. For showing the variable names instead as L, C and X, a function `convert_pat_to_lcx_dBite` was written, and shown in Script A3.19.

*Script A3.19 convert\_pat\_to\_lcx\_dBite function and example application in R*

```

1 convert_pat_to_lcx_dBite <- function(a) {
2   library(stringr)
3   library(dplyr)
4   a$type <- "notAssignedYet"
5   a$type <- with(a,ifelse(grepl("Entr",a$variables),"t",a$type))
6   a$type <- with(a,ifelse(grepl("mean",a$variables),"t",a$type))
7   a$type <- with(a,ifelse(grepl("Contr",a$variables),"t",a$type))
8   a$type <- with(a,ifelse(grepl("Var",a$variables),"t",a$type))
9   a$type <- with(a,ifelse(grepl("Corr",a$variables),"t",a$type))
10  a$type <- with(a,ifelse(grepl("elev",a$variables),"e",a$type))
11  a$type <- with(a,ifelse(grepl("slope",a$variables),"e",a$type))
12  a$type <- with(a,ifelse(grepl("aspect",a$variables),"e",a$type))
13  a$type <- with(a,ifelse(grepl("linear",a$variables),"i",a$type))
14  a$type <- with(a,ifelse(grepl("NDIS",a$variables),"i",a$type))
15  a$type <- with(a,ifelse(grepl("rdB",a$variables),"i",a$type))
16  a$type <- with(a,ifelse(grepl("rdB",a$variables),"i",a$type))
17  a$type <- with(a,ifelse(grepl("copol",a$variables),"i",a$type))
18  a$type <- with(a,ifelse(grepl("depol",a$variables),"i",a$type))
19  a$type <- with(a,ifelse(grepl("HVS",a$variables),"dB",a$type))
20  a$type <- with(a,ifelse(grepl("VHS",a$variables),"dB",a$type))
21  a$type <- with(a,ifelse(grepl("HHS",a$variables),"dB",a$type))
22  a$type <- with(a,ifelse(grepl("VVS",a$variables),"dB",a$type))
23  a$type <- with(a,ifelse(grepl("HVG",a$variables),"dB",a$type))
24  a$type <- with(a,ifelse(grepl("VHG",a$variables),"dB",a$type))
25  a$type <- with(a,ifelse(grepl("HHG",a$variables),"dB",a$type))
26  a$type <- with(a,ifelse(grepl("VVG",a$variables),"dB",a$type))
27  a$e <- with(a,ifelse(a$type ==
28    "e",substring(a$variables,4,nchar(a$variables)),""))
29    library(stringr)
30    library(dplyr)
31    a$bcc <- ""
32    a$b <- ""
33    a$cc <- ""
34    a$freq <- ""
35    a$var <- ""
36    a$calc1 <- ""
37    a$calc2 <- ""

```

```

36         a$win1 <- ""
37         a$win2 <- ""
38         a$m <- ""
39         a$descrip <- ""
40         a$bcc <- with(a,ifelse((a$type == "t" | a$type == "i" | a$type
== "dB"), substring(a$variables,4,6),a$bcc))
41         a$b <- with(a,ifelse((a$type == "t" | a$type == "i" | a$type
== "dB"),substring(a$variables,4,4),a$b))
42         a$cc <- with(a,ifelse((a$type == "t" | a$type == "i" | a$type
== "dB"),substring(a$variables,5,6),a$cc))
43         a$freq <- with(a,ifelse(b == "P","L",
44                                 ifelse(b == "A","C",
45                                         ifelse(b == "T","X",""))))
46         a$var <- with(a,ifelse(a$type == "t" | a$type == "i" | a$type ==
"dB",paste0(a$freq,"-",a$cc),a$freq))
47         a$calc1 <- with(a,ifelse(a$type == "t" | a$type ==
"i",substring(a$variables,5,nchar(a$variables)),a$calc1))
48         a$calc2 <- with(a,ifelse(a$type == "t" | a$type ==
"i",paste0(a$freq,"-",a$calc1),a$calc2))
49         a$calc2 <- with(a,ifelse(a$type == "t" | a$type ==
"i",gsub("S","",a$calc2),a$calc2))
50         a$calc2 <- with(a,ifelse(a$type == "t" | a$type ==
"i",gsub("linear","lin",a$calc2),a$calc2))
51         a$calc2 <- with(a,ifelse(a$type == "t" | a$type ==
"i",gsub("plus","p",a$calc2),a$calc2))
52         a$calc2 <- with(a,ifelse(a$type == "t" | a$type ==
"i",gsub("min","m",a$calc2),a$calc2))
53         a$calc2 <- with(a,ifelse(a$type == "t" | a$type ==
"i",gsub("div","d",a$calc2),a$calc2))
54         a$calc2 <- with(a,ifelse(a$type == "t" | a$type ==
"i",gsub("_gamma","",a$calc2),a$calc2))
55         a$calc2 <- with(a,ifelse(a$type == "t" | a$type ==
"i",gsub("_delta","",a$calc2),a$calc2))
56         a$win1 <- with(a,ifelse(a$type ==
"t",substring(a$calc2,5,6),a$win1))
57         a$win2 <- with(a,ifelse(substring(win1,1,1) ==
"1",substring(win1,1,2),substring(win1,1,1)))
58         a$m <- with(a,ifelse(a$type == "t",
                                ifelse(nchar(win2) ==
2,substring(calc2,7,nchar(calc2)),substring(calc2,6,nchar(calc2))),a$m))
59         a$m <- with(a,ifelse(a$type == "t",gsub("1","",a$m),a$m))
60         a$m <- with(a,ifelse(a$type == "t",gsub("2","",a$m),a$m))
61         a$m <- with(a,ifelse(a$type ==
"t",gsub("mean","Mean",a$m),a$m))
62         a$descrip <- with(a,ifelse(a$type == "dB",var,a$descrip))
63         a$descrip <- with(a,ifelse(a$type == "i",calc2,a$descrip))
64         a$descrip <- with(a,ifelse(a$type == "t",calc2,a$descrip))

```

```

65         a$descrip <- with(a,ifelse(a$type == "e",e,a$descrip))
66     return(a)
67 }
68
69 load("dBite_dBit_t_unscaled.rda")
70 dBite_cd_uns <- convert_pat_to_lcx_dBite(cd.lcx.dBite.S.varimp.rf.uns)
71 dBit_cd_uns <- convert_pat_to_lcx_dBite(cd.lcx.dBit.S.varimp.rf_uns)
72 t_cd_uns <- convert_pat_to_lcx_dBite(cd.lcx.t.S.varimp.rf_uns)
73
74 db_cd_uns_select <- dBite_cd_uns[dBite_cd_uns$type=="dB",]
75 i_cd_uns_select <- dBite_cd_uns[dBite_cd_uns$type=="i",]
76 t_cd_uns_select <- dBite_cd_uns[dBite_cd_uns$type=="t",]
77 e_cd_uns_select <- dBite_cd_uns[dBite_cd_uns$type=="e",]

```

*Script A3.20 corrplot function and example application in R*

```

1 library(corrplot)
2 #cd mdg
3 ppi <- 300
4 png("cd.lcx.dBite.S.train_top30_mdg_cor.png",height=8*ppi,width=7*ppi,
5 res=ppi,bg="transparent")
6 corrplot(cd.lcx.dBite.S.train_top30_mdg_newcolnames_cor,
7 method='ellipse',mar=c(0,0,2,0),order="hclust",tl.col="black",
8 title="Cameroon dry top 30 MDG correlation, hclust sorting")
9 dev.off()

```

*Script A3.21 plot\_overall\_accucary\_with\_class\_accuracy\_top1\_to\_30\_in1s function*

```

1 plot_overall_accucary_with_class_accuracy_top1_to_30_in1s <-
function(input1.conf,input2.conf,input3.conf,input4.conf, input5.conf,
2 input6.conf,input7.conf,input8.conf,input9.conf,input10.conf,
3 input11.conf,input12.conf,input13.conf,input14.conf,input15.conf,
4 input16.conf,input17.conf,input18.conf,input19.conf,input20.conf,
5 input21.conf,input22.conf,input23.conf,input24.conf,input25.conf,
6 input26.conf,input27.conf,input28.conf,input29.conf,input30.conf,
7 site,season,GS,vardescrip,legendYN) {
8
9     input.byClass <- rbind(input1.conf$byClass[,1],input2.conf$byClass[,1],
10 input3.conf$byClass[,1],input4.conf$byClass[,1],input5.conf$byClass[,1],
11 input6.conf$byClass[,1], input7.conf$byClass[,1], input8.conf$byClass[,1],
12 input9.conf$byClass[,1],input10.conf$byClass[,1],input11.conf$byClass[,1],
13 input12.conf$byClass[,1], input13.conf$byClass[,1], input14.conf$byClass[,1],
14 input15.conf$byClass[,1], input16.conf$byClass[,1], input17.conf$byClass[,1],
15 input18.conf$byClass[,1],input19.conf$byClass[,1],input20.conf$byClass[,1],

```

```

16 input21.conf$byClass[,1], input22.conf$byClass[,1], input23.conf$byClass[,1],
17 input24.conf$byClass[,1], input25.conf$byClass[,1], input26.conf$byClass[,1],
18 input27.conf$byClass[,1], input28.conf$byClass[,1], input29.conf$byClass[,1],
19 input30.conf$byClass[,1] )
20
21   input.overall <- c(input1.conf$overall[1], input2.conf$overall[1],
22 input3.conf$overall[1], input4.conf$overall[1], input5.conf$overall[1],
23 input6.conf$overall[1], input7.conf$overall[1], input8.conf$overall[1],
24 input9.conf$overall[1], input10.conf$overall[1], input11.conf$overall[1],
25 input12.conf$overall[1], input13.conf$overall[1], input14.conf$overall[1],
26 input15.conf$overall[1], input16.conf$overall[1], input17.conf$overall[1],
27 input18.conf$overall[1], input19.conf$overall[1], input20.conf$overall[1],
28 input21.conf$overall[1], input22.conf$overall[1], input23.conf$overall[1],
29 input24.conf$overall[1], input25.conf$overall[1],
30 input26.conf$overall[1], input27.conf$overall[1], input28.conf$overall[1],
31 input29.conf$overall[1], input30.conf$overall[1] )
32   #input.overall
33
34   lcx <- c('1','2','3','4','5',
35           '6','7','8','9','10',
36           '11','12','13','14','15',
37           '16','17','18','19','20',
38           '21','22','23','24','25',
39           '26','27','28','29','30')
40
41   input.overall2 <- cbind(lcx, input.overall)
42   input.overall.order <- input.overall2[order(input.overall)]
43   input.byClass <- round(input.byClass * 100, 2)
44   input.overall <- round(input.overall * 100, 2)
45   input.byClass1 <- cbind(input.byClass, input.overall)
46
47   if (site == 'Cameroon') {
48     site_columnnames <- c("BareSoil", "DenseTrees", "Settlement",
49                           "SparseVeg", "ThinnerDenseTrees", "Water", "Overall")
50   }
51   if (site == 'Tanzania') {
52     site_columnnames <- c("Agriculture", "BareSoil", "DenseTrees",
53                           "Settlement", "SparseVeg", "ThinnerDenseTrees", "Overall")
54   }
55   #input.byClass1
56   colnames(input.byClass1) <- site_columnnames
57   rownames(input.byClass1) <- lcx
58   input.byClass2 <- cbind(lcx, as.data.frame(input.byClass1))
59   input.byClass.long <- melt(input.byClass2, id.vars=c("lcx"))
60   input.byClass.long$lcx <- as.factor(input.byClass.long$lcx)
61   input.byClass.long$lcx <- ordered(input.byClass.long$lcx,
62                                     levels = c('1','2','3','4','5',

```

```

63         '6','7','8','9','10',
64         '11','12','13','14','15',
65         '16','17','18','19','20',
66         '21','22','23','24','25',
67         '26','27','28','29','30'))
68     ##input.byClass.long$lcx <- ordered(input.byClass.long$lcx,
69     ## levels =c(input.overall.order)) #this line was commented out, which
70     ## was originally ordered in LCX
71
72     if (site == 'Cameroon') {
73         site_colours <- c("#FF5500", "#007139", "#FF0000", "#FB3AFF",
74         "#00D362", "#0000FF", "#000000")
75     }
76     if (site == 'Tanzania') {
77         site_colours <- c("#55AAFF", "#FF5500", "#007139", "#FF0000",
78         "#FB3AFF", "#00D362", "#000000")
79     }
80
81     if (GS == "S" && legendYN == "Y") {
82         pinplot <- ggplot(input.byClass.long, aes(x=lcx, y=value, colour=variable,
83         group=variable)) + geom_line(size=1.2) +
84         #ggtitle(bquote(. (site) ~ . (season) ~ sigma^0 ~ . (vardescrip) ~ "RF")) +
85         #ggtitle(bquote(. (site) ~ . (season) ~ sigma^0 ~ . (vardescrip) ~ "RF")) +
86         ggtitle(bquote(. (site) ~ . (season) ~ "")) +
87         #ggtitle(expression(paste0(site, " ", season, " ", sigma^0, " ", vardescrip, "
88         RF)))) +
89         ylab("Classification Accuracy on Test Data") +
90         #xlab("Sorted Sensor combinations") +
91         xlab("") +
92         scale_color_manual(values=site_colours) + ylim(0,100) +
93
94         annotate("text",x=5.2,y=(input.overall[5]+5),label=paste0(round(input.ove
95         rall[5]),"%"),size=8) +
96
97         annotate("text",x=10.2,y=(input.overall[10]+5),label=paste0(round(input.ove
98         rall[10]),"%"),size=8) +
99
100        annotate("text",x=15.2,y=(input.overall[15]+5),label=paste0(round(input.ove
101        rall[15]),"%"),size=8) +
102
103        annotate("text",x=20.2,y=(input.overall[20]+5),label=paste0(round(input.ove
104        rall[20]),"%"),size=8) +
105
106        annotate("text",x=25.2,y=(input.overall[25]+5),label=paste0(round(input.ove
107        rall[25]),"%"),size=8) +

```

```

103 annotate("text",x=29.2,y=(input.overall[30]+5),label=paste0(round(input.ove
104 rall[30]),"%"),size=8) +
105     theme(plot.title=element_text(size=rel(3.0),lineheight=2.0,vjust=2),
106           legend.title=element_text(size=rel(1.5),color="white"),
107           legend.text=element_text(size=rel(2.0),lineheight=2.0),
108           legend.key.height=unit(1,"cm"),
109           axis.title=element_text(size=rel(1.8)),
110           axis.text=element_text(size=rel(1.5),colour='black'),
111
112 axis.ticks=element_line(size=rel(2.0)),strip.text=element_text(size=rel(2.0)),
113   plot.background=element_rect(fill = "transparent", colour = NA)
114   )
115   } else if (GS == "G" && legendYN == "Y") {
116     pinplot <- ggplot(input.byClass.long, aes(x=lcx, y=value, colour=variable,
117 group=variable)) + geom_line(size=1.2) +
118     #ggtitle(bquote(.(site) ~ .(season) ~ gamma^0 ~ .(vardescrip) ~ "RF")) +
119     ggtitle(bquote(.(site) ~ .(season) ~ "")) +
120     #ggtitle(expression(paste(site,season,gamma^0,vardescrip," RF"))) +
121     #ggtitle(expression(paste0(site," ",season," ",gamma^0," ",vardescrip,"
122 RF")))) +
123     ylab("Classification Accuracy on Test Data") +
124     #xlab("Sorted Sensor combinations") +
125     xlab("") +
126     scale_color_manual(values=site_colours) + ylim(0,100) +
127
128     annotate("text",x=5.2,y=(input.overall[5]+5),label=paste0(round(input.ove
129 rall[5]),"%"),size=8) +
130
131     annotate("text",x=10.2,y=(input.overall[10]+5),label=paste0(round(input.ove
132 rall[10]),"%"),size=8) +
133
134     annotate("text",x=15.2,y=(input.overall[15]+5),label=paste0(round(input.ove
135 rall[15]),"%"),size=8) +
136
137     annotate("text",x=20.2,y=(input.overall[20]+5),label=paste0(round(input.ove
138 rall[20]),"%"),size=8) +
139
140     annotate("text",x=25.2,y=(input.overall[25]+5),label=paste0(round(input.ove
141 rall[25]),"%"),size=8) +
142
143     annotate("text",x=29.2,y=(input.overall[30]+5),label=paste0(round(input.ove
144 rall[30]),"%"),size=8) +
145     theme(plot.title=element_text(size=rel(3.0),lineheight=2.0,vjust=2),
146           legend.title=element_text(size=rel(1.5),color="white"),
147           legend.text=element_text(size=rel(2.0),lineheight=2.0),
148           legend.key.height=unit(1,"cm"),

```

```

141     axis.title=element_text(size=rel(1.8)),
142     axis.text=element_text(size=rel(1.5),colour='black'),
143
144     axis.ticks=element_line(size=rel(2.0)),strip.text=element_text(size=rel(2.0)),
145     plot.background=element_rect(fill = "transparent", colour = NA)
146   )
147   } else if (GS == "S" && legendYN == "N") {
148     pinplot <- ggplot(input.byClass.long, aes(x=lcx, y=value, colour=variable,
149     group=variable)) + geom_line(size=1.2) +
150     #ggtitle(bquote(. (site) ~ . (season) ~ sigma^0 ~ . (vardescrip) ~ "RF")) +
151     #ggtitle(bquote(. (site) ~ . (season) ~ sigma^0 ~ . (vardescrip) ~ "RF")) +
152     ggtitle(bquote(. (site) ~ . (season) ~ "")) +
153     #ggtitle(expression(paste0(site, " ",season, " ",sigma^0, " ",vardescrip, "
154     RF)))) +
155     ylab("Classification Accuracy on Test Data") +
156     #xlab("Sorted Sensor combinations") +
157     xlab("") +
158     scale_color_manual(values=site_colours) + ylim(0,100) +
159
160     #annotate("text",x=29.2,y=(max(input.overall)+5),label=paste0(round(max(in
161     put.overall)), "%"),size=8) +
162
163     annotate("text",x=5.2,y=(input.overall[5]+5),label=paste0(round(input.ove
164     rall[5]), "%"),size=8) +
165
166     annotate("text",x=10.2,y=(input.overall[10]+5),label=paste0(round(input.ove
167     rall[10]), "%"),size=8) +
168
169     annotate("text",x=15.2,y=(input.overall[15]+5),label=paste0(round(input.ove
170     rall[15]), "%"),size=8) +
171
172     annotate("text",x=20.2,y=(input.overall[20]+5),label=paste0(round(input.ove
173     rall[20]), "%"),size=8) +
174
175     annotate("text",x=25.2,y=(input.overall[25]+5),label=paste0(round(input.ove
176     rall[25]), "%"),size=8) +
177
178     annotate("text",x=29.2,y=(input.overall[30]+5),label=paste0(round(input.ove
179     rall[30]), "%"),size=8) +
180
181     theme(plot.title=element_text(size=rel(3.0),lineheight=2.0,vjust=2),
182     legend.title=element_text(size=rel(1.5),color="white"),
183     legend.text=element_text(size=rel(2.0),lineheight=2.0),
184     legend.key.height=unit(1,"cm"),
185     legend.position="none",
186     axis.title=element_text(size=rel(1.8)),
187     axis.text=element_text(size=rel(1.5),colour='black'),
188

```



```

179
180 axis.ticks=element_line(size=rel(2.0)),strip.text=element_text(size=rel(2.0)),
181   plot.background=element_rect(fill = "transparent", colour = NA)
182   )
    } else if (GS == "G" && legendYN == "N") {
183     pinplot <- ggplot(input.byClass.long, aes(x=lcx, y=value, colour=variable,
184 group=variable)) + geom_line(size=1.2) +
185       #ggtitle(bquote(.(site) ~ .(season) ~ gamma^0 ~ .(vardescrip) ~ "RF")) +
186       ggtitle(bquote(.(site) ~ .(season) ~ "")) +
187       #ggtitle(expression(paste(site,season,gamma^0,vardescrip," RF"))) +
188       #ggtitle(expression(paste0(site," ",season," ",gamma^0," ",vardescrip,"
RF")))) +
189       ylab("Classification Accuracy on Test Data") +
190       #xlab("Sorted Sensor combinations") +
191       xlab("") +
192       scale_color_manual(values=site_colours) + ylim(0,100) +

193 #annotate("text",x=29.2,y=(max(input.overall)+5),label=paste0(round(max(in
194 put.overall)), "%"),size=8) +

195 annotate("text",x=5.2,y=(input.overall[5]+5),label=paste0(round(input.ove
196 ral[5]), "%"),size=8) +

197 annotate("text",x=10.2,y=(input.overall[10]+5),label=paste0(round(input.ove
198 ral[10]), "%"),size=8) +

199 annotate("text",x=15.2,y=(input.overall[15]+5),label=paste0(round(input.ove
200 ral[15]), "%"),size=8) +

201 annotate("text",x=20.2,y=(input.overall[20]+5),label=paste0(round(input.ove
202 ral[20]), "%"),size=8) +

203 annotate("text",x=25.2,y=(input.overall[25]+5),label=paste0(round(input.ove
204 ral[25]), "%"),size=8) +

205 annotate("text",x=29.2,y=(input.overall[30]+5),label=paste0(round(input.ove
206 ral[30]), "%"),size=8) +
    theme(plot.title=element_text(size=rel(3.0),lineheight=2.0,vjust=2),
207         legend.title=element_text(size=rel(1.5),color="white"),
208         legend.text=element_text(size=rel(2.0),lineheight=2.0),
209         legend.key.height=unit(1,"cm"),
210         legend.position="none",
211         axis.title=element_text(size=rel(1.8)),
212         axis.text=element_text(size=rel(1.5),colour='black'),
213
214
215 axis.ticks=element_line(size=rel(2.0)),strip.text=element_text(size=rel(2.0)),
216   plot.background=element_rect(fill = "transparent", colour = NA)

```

```

217     )
218   }
219   return(pinput)
220 }
221
222 #apply plot_overall_accucary_with_class_accuracy_top1_to_30_in1s
223 png("top01to30_S_MDG_overallaccuracy_legendY900h.png",height=900,wid
224 th=1980, bg="transparent")
225 p1 <-
226 plot_overall_accucary_with_class_accuracy_top1_to_30_in1s(cd.dBit.rf.conf.
227 mdg.top1, cd.dBit.rf.conf.mdg.top2, cd.dBit.rf.conf.mdg.top3,
228 cd.dBit.rf.conf.mdg.top4, cd.dBit.rf.conf.mdg.top5, cd.dBit.rf.conf.mdg.top6,
229 cd.dBit.rf.conf.mdg.top7, cd.dBit.rf.conf.mdg.top8, cd.dBit.rf.conf.mdg.top9,
230 cd.dBit.rf.conf.mdg.top10, cd.dBit.rf.conf.mdg.top11,
231 cd.dBit.rf.conf.mdg.top12, cd.dBit.rf.conf.mdg.top13,
232 cd.dBit.rf.conf.mdg.top14, cd.dBit.rf.conf.mdg.top15,
233 cd.dBit.rf.conf.mdg.top16, cd.dBit.rf.conf.mdg.top17,
234 cd.dBit.rf.conf.mdg.top18, cd.dBit.rf.conf.mdg.top19,
235 cd.dBit.rf.conf.mdg.top20, cd.dBit.rf.conf.mdg.top21,
236 cd.dBit.rf.conf.mdg.top22, cd.dBit.rf.conf.mdg.top23,
237 cd.dBit.rf.conf.mdg.top24, cd.dBit.rf.conf.mdg.top25,
238 cd.dBit.rf.conf.mdg.top26, cd.dBit.rf.conf.mdg.top27,
239 cd.dBit.rf.conf.mdg.top28, cd.dBit.rf.conf.mdg.top29,
240 cd.dBit.rf.conf.mdg.top30, "Cameroon","dry","S","MDG","Y")
241 p2 <-
242 plot_overall_accucary_with_class_accuracy_top1_to_30_in1s(cw.dBit.rf.conf
243 .mdg.top1, cw.dBit.rf.conf.mdg.top2, cw.dBit.rf.conf.mdg.top3,
244 cw.dBit.rf.conf.mdg.top4, cw.dBit.rf.conf.mdg.top5,
245 cw.dBit.rf.conf.mdg.top6, cw.dBit.rf.conf.mdg.top7,
246 cw.dBit.rf.conf.mdg.top8, cw.dBit.rf.conf.mdg.top9,
247 cw.dBit.rf.conf.mdg.top10, cw.dBit.rf.conf.mdg.top11,
248 cw.dBit.rf.conf.mdg.top12, cw.dBit.rf.conf.mdg.top13,
249 cw.dBit.rf.conf.mdg.top14, cw.dBit.rf.conf.mdg.top15,
250 cw.dBit.rf.conf.mdg.top16, cw.dBit.rf.conf.mdg.top17,
251 cw.dBit.rf.conf.mdg.top18, cw.dBit.rf.conf.mdg.top19,
252 cw.dBit.rf.conf.mdg.top20, cw.dBit.rf.conf.mdg.top21,
253 cw.dBit.rf.conf.mdg.top22, cw.dBit.rf.conf.mdg.top23,
254 cw.dBit.rf.conf.mdg.top24, cw.dBit.rf.conf.mdg.top25,
255 cw.dBit.rf.conf.mdg.top26, cw.dBit.rf.conf.mdg.top27,
256 cw.dBit.rf.conf.mdg.top28, cw.dBit.rf.conf.mdg.top29,
257 cw.dBit.rf.conf.mdg.top30, "Cameroon","wet","S","MDG","Y")
258 p3 <-
259 plot_overall_accucary_with_class_accuracy_top1_to_30_in1s(td.dBit.rf.conf.
260 mdg.top1, td.dBit.rf.conf.mdg.top2, td.dBit.rf.conf.mdg.top3,
261 td.dBit.rf.conf.mdg.top4, td.dBit.rf.conf.mdg.top5, td.dBit.rf.conf.mdg.top6,
262 td.dBit.rf.conf.mdg.top7, td.dBit.rf.conf.mdg.top8, td.dBit.rf.conf.mdg.top9,
263 td.dBit.rf.conf.mdg.top10, td.dBit.rf.conf.mdg.top11,

```

```

264 td.dBit.rf.conf.mdg.top12, td.dBit.rf.conf.mdg.top13,
265 td.dBit.rf.conf.mdg.top14, td.dBit.rf.conf.mdg.top15,
266 td.dBit.rf.conf.mdg.top16, td.dBit.rf.conf.mdg.top17,
267 td.dBit.rf.conf.mdg.top18, td.dBit.rf.conf.mdg.top19,
268 td.dBit.rf.conf.mdg.top20, td.dBit.rf.conf.mdg.top21,
269 td.dBit.rf.conf.mdg.top22, td.dBit.rf.conf.mdg.top23,
270 td.dBit.rf.conf.mdg.top24, td.dBit.rf.conf.mdg.top25,
271 td.dBit.rf.conf.mdg.top26, td.dBit.rf.conf.mdg.top27,
272 td.dBit.rf.conf.mdg.top28, td.dBit.rf.conf.mdg.top29,
273 td.dBit.rf.conf.mdg.top30, "Tanzania", "dry", "S", "MDG", "Y")
274 p4 <-
275 plot_overall_accucary_with_class_accuracy_top1_to_30_in1s(tw.dBit.rf.conf.
276 mdg.top1, tw.dBit.rf.conf.mdg.top2, tw.dBit.rf.conf.mdg.top3,
277 tw.dBit.rf.conf.mdg.top4, tw.dBit.rf.conf.mdg.top5, tw.dBit.rf.conf.mdg.top6,
278 tw.dBit.rf.conf.mdg.top7, tw.dBit.rf.conf.mdg.top8, tw.dBit.rf.conf.mdg.top9,
279 tw.dBit.rf.conf.mdg.top10, tw.dBit.rf.conf.mdg.top11,
280 tw.dBit.rf.conf.mdg.top12, tw.dBit.rf.conf.mdg.top13,
281 tw.dBit.rf.conf.mdg.top14, tw.dBit.rf.conf.mdg.top15,
282 tw.dBit.rf.conf.mdg.top16, tw.dBit.rf.conf.mdg.top17,
283 tw.dBit.rf.conf.mdg.top18, tw.dBit.rf.conf.mdg.top19,
284 tw.dBit.rf.conf.mdg.top20, tw.dBit.rf.conf.mdg.top21,
285 tw.dBit.rf.conf.mdg.top22, tw.dBit.rf.conf.mdg.top23,
286 tw.dBit.rf.conf.mdg.top24, tw.dBit.rf.conf.mdg.top25,
287 tw.dBit.rf.conf.mdg.top26, tw.dBit.rf.conf.mdg.top27,
288 tw.dBit.rf.conf.mdg.top28, tw.dBit.rf.conf.mdg.top29,
289 tw.dBit.rf.conf.mdg.top30, "Tanzania", "wet", "S", "MDG", "Y")
290 multiplot(p1,p3,p2,p4,cols=2)
dev.off()

```

*Script A3.22 apply\_Model1\_on\_Site2\_conf\_newsite\_cw\_traintest*

```

1 apply_Model1_on_Site2_conf_newsite_cw_traintest <- function(train1,
2 train2, test1, test2, same_other_ind, overlap_colnames,
3                     classname_site_ind,
4                     LCX_ind, dBit_ind, elevYN,
5                     model_ind, mda_mdg_ind, top_x) {
6   # train1 <- cw3000.train
7   # train2 <- cd3000.train
8   # test1 <- cw3000.test
9   # test2 <- cd3000.test
10  # same_other_ind <- "wetdry"
11  # overlap_colnames <- F_overlap_CW_CD_S
12  # classname_site_ind <- "classname_wetdry_site" #"classname_tt_site"
13  # LCX_ind <- c("L", "C", "X")
14  # dBit_ind <- c("dB", "i", "t")
15  # elevYN <- "Y"

```

```

16 # model_ind <- "RF"
17 # mda_mdg_ind <- "MDG"
18 # top_x <- 30
19 #
20 colnames2 <- return_LCX(overlap_colnames,LCX_ind)
21 colnames3 <- return_dBit(colnames2,dBit_ind)
22 if (elevYN == "Y") { colnames4 <- c(colnames3,"elev","slope","aspect")
23 } else if (elevYN == "N") { colnames4 <- colnames3 }
24
25 b_d_se_sp_w <-
26 c("Bare","DenseTrees","Settlement","SparseVeg","Water")
27 b_d_se_sp_w_site <- c("CW_Bare", "CW_DenseTrees", "CW_Settlement",
28 "CW_SparseVeg", "CW_Water", "DW_Bare", "DW_DenseTrees",
29 "DW_Settlement", "DW_SparseVeg", "DW_Water")
30 b_d_se_sp_w_site_only.cw <- c("CW_Bare", "CW_DenseTrees",
31 "CW_Settlement", "CW_SparseVeg", "CW_Water")
32 b_d_se_sp_w_site_only.dw <- c("DW_Bare", "DW_DenseTrees",
33 "DW_Settlement", "DW_SparseVeg", "DW_Water")
34 #train vs test comparison
35 #b_d_se_sp_w <- c("Bare", "DenseTrees", "Settlement", "SparseVeg",
36 "Water")
37 b_d_se_sp_w_site_cw_tt <- c("CW_Train_Bare", "CW_Train_DenseTrees",
38 "CW_Train_Settlement", "CW_Train_SparseVeg", "CW_Train_Water",
39 "CW_Test_Bare", "CW_Test_DenseTrees", "CW_Test_Settlement",
40 "CW_Test_SparseVeg", "CW_Test_Water")
41 b_d_se_sp_w_site_only.cw_train <- c("CW_Train_Bare",
42 "CW_Train_DenseTrees", "CW_Train_Settlement", "CW_Train_SparseVeg",
43 "CW_Train_Water")
44 b_d_se_sp_w_site_only.cw_test <- c("CW_Test_Bare",
45 "CW_Test_DenseTrees", "CW_Test_Settlement", "CW_Test_SparseVeg",
46 "CW_Test_Water")
47 #b_d_se_sp_w_site_cw_cd <- c("CW_Train_Bare",
48 #"CW_Train_DenseTrees", "CW_Train_Settlement",
49 #"CW_Train_SparseVeg", "CW_Train_Water",
50 #"CD_Train_Bare", "CD_Train_DenseTrees", "CD_Train_Settlement",
51 #"CD_Train_SparseVeg", "CD_Train_Water")
52
53 b_d_se_sp_w_site_cw_cd <- c("CW_Bare", "CW_DenseTrees",
54 "CW_Settlement", "CW_SparseVeg", "CW_Water", "CD_Bare",
55 "CD_DenseTrees", "CD_Settlement", "CD_SparseVeg", "CD_Water")
56
57 if (classname_site_ind == "classname") {
58 classes <- b_d_se_sp_w
59 } else if (classname_site_ind == "classname_site" & same_other_ind ==
60 "cw") {
61 classes <- b_d_se_sp_w_site_only.cw

```

```

61   } else if (classname_site_ind == "classname_site" & same_other_ind ==
    "dw") {
62     classes <- b_d_se_sp_w_site_only.dw
63   } else if (classname_site_ind == "classname_site" & same_other_ind ==
    "other") {
64     classes <- b_d_se_sp_w_site
65   }
66
67   if (classname_site_ind == "classname_tt_site") {
68     classes <- b_d_se_sp_w_site_cw_tt
69   }
70
71   if (classname_site_ind == "classname_wetdry_site") {
72     classes <- b_d_se_sp_w_site_cw_cd
73   }
74
75   #=====cw train vs cw test - same classes=====
76   if (same_other_ind == "cw" | same_other_ind == "dw" ) {
77     if (classname_site_ind == "classname") {
78       a.train1 = subset(train1, classname %in% classes,
79                         select = c(colnames4,'classname'))
80       a.test1 = subset(test1, classname %in% classes,
81                        select = c(colnames4,'classname'))
82       a.train1$classname <- factor(a.train1$classname,levels=classes)
83       a.test1$classname <- factor(a.test1$classname,levels=classes)
84       a.train <- a.train1
85       a.test <- a.test1
86       #same_other_ind == "cw" | same_other_ind == "dw"
87       #classname_site_ind == "classname"
88     } else
89       #=====cw train vs cw test - sitespecific classes=====
90       if (classname_site_ind == "classname_site") {
91         a.train1 = subset(train1, classname_site %in% classes,
92                           select = c(colnames4,'classname_site'))
93         a.test1 = subset(test1, classname_site %in% classes,
94                           select = c(colnames4,'classname_site'))
95         a.train1$classname_site <- factor(a.train1$classname_site,
96                                           levels=classes)
97         a.test1$classname_site <- factor(a.test1$classname_site,
98                                           levels=classes)
99         names(a.train1)[names(a.train1) == 'classname_site'] <- 'classname'
100        names(a.test1)[names(a.test1) == 'classname_site'] <- 'classname'
101        a.train <- a.train1
102        a.test <- a.test1
103        #same_other_ind == "cw" | same_other_ind == "dw"
104        #classname_site_ind == "classname_site"
105      }

```

```

106
107   } else if(same_other_ind == "other"){
108     #cw train vs dw train
109     if (classname_site_ind == "classname") {
110       a.train1 = subset(train1, classname %in% classes,
111                         select = c(colnames4,'classname'))
112       a.test1 = subset(test1, classname %in% classes,
113                       select = c(colnames4,'classname'))
114       a.train2 = subset(train2, classname %in% classes,
115                        select = c(colnames4,'classname'))
116       a.test2 = subset(test2, classname %in% classes,
117                       select = c(colnames4,'classname'))
118       a.train1$classname <- factor(a.train1$classname,levels=classes)
119       a.test1$classname <- factor(a.test1$classname,levels=classes)
120       a.train2$classname <- factor(a.train2$classname,levels=classes)
121       a.test2$classname <- factor(a.test2$classname,levels=classes)
122       #a.train <- rbind(a.train1,a.train2)
123       #a.test <- rbind(a.test1,a.test2)
124       a.train <- a.train1 #cw.train
125       a.test <- a.train2 #dw.train
126       #same_other_ind == "other"
127       #classname_site_ind == "classname"
128     } else
129     if (classname_site_ind == "classname_site") {
130       a.train1 = subset(train1, classname_site %in% classes,
131                        select = c(colnames4,'classname_site'))
132       a.test1 = subset(test1, classname_site %in% classes,
133                       select = c(colnames4,'classname_site'))
134       a.train1$classname_site <- factor(a.train1$classname_site,
135                                       levels=classes)
136       a.test1$classname_site <- factor(a.test1$classname_site,
137                                       levels=classes)
138       names(a.train1)[names(a.train1) == 'classname_site'] <- 'classname'
139       names(a.test1)[names(a.test1) == 'classname_site'] <- 'classname'
140       a.train2 = subset(train2, classname_site %in% classes,
141                        select = c(colnames4,'classname_site'))
142       a.test2 = subset(test2, classname_site %in% classes,
143                       select = c(colnames4,'classname_site'))
144       a.train2$classname_site <- factor(a.train2$classname_site,
145                                       levels=classes)
146       a.test2$classname_site <- factor(a.test2$classname_site,
147                                       levels=classes)
148       names(a.train2)[names(a.train2) == 'classname_site'] <- 'classname'
149       names(a.test2)[names(a.test2) == 'classname_site'] <- 'classname'
150       #a.train <- rbind(a.train1,a.train2)
151       #a.test <- rbind(a.test1,a.test2)
152       a.train <- rbind(a.train1,a.train2)

```

```

153     a.test <- rbind(a.test1,a.test2)
154     #same_other_ind == "other"
155     #classname_site_ind == "classname"
156   }
157 } else if(same_other_ind == "tt"){
158   if (classname_site_ind == "classname_tt_site") {
159     a.train1 = subset(train1, classname_tt_site %in% classes,
160                       select = c(colnames4,'classname_tt_site'))
161     a.test1 = subset(test1, classname_tt_site %in% classes,
162                     select = c(colnames4,'classname_tt_site'))
163     a.train1$classname_tt_site <- factor(a.train1$classname_tt_site,
164                                         levels=classes)
165     a.test1$classname_tt_site <- factor(a.test1$classname_tt_site,
166                                         levels=classes)
167     names(a.train1)[names(a.train1) == 'classname_tt_site'] <- 'classname'
168     names(a.test1)[names(a.test1) == 'classname_tt_site'] <- 'classname'
169     a.train2 = subset(train2, classname_tt_site %in% classes,
170                       select = c(colnames4,'classname_tt_site'))
171     a.test2 = subset(test2, classname_tt_site %in% classes,
172                     select = c(colnames4,'classname_tt_site'))
173     a.train2$classname_tt_site <- factor(a.train2$classname_tt_site,
174                                         levels=classes)
175     a.test2$classname_tt_site <- factor(a.test2$classname_tt_site,
176                                         levels=classes)
177     names(a.train2)[names(a.train2) == 'classname_tt_site'] <- 'classname'
178     names(a.test2)[names(a.test2) == 'classname_tt_site'] <- 'classname'
179     a.train <- rbind(a.train1,a.train2)
180     a.test <- rbind(a.test1,a.test2)
181     #a.train <- a.train1
182     #a.test <- a.train2
183   }
184 } else if(same_other_ind == "wetdry"){
185   if (classname_site_ind == "classname_wetdry_site") {
186     a.train1 = subset(train1, classname_site %in% classes,
187                       select = c(colnames4,'classname_site'))
188     #head(a.train1)
189     a.test1 = subset(test1, classname_site %in% classes,
190                     select = c(colnames4,'classname_site'))
191     #head(a.test1)
192     a.train1$classname_site <- factor(a.train1$classname_site,
193                                         levels=classes)
194     a.test1$classname_site <- factor(a.test1$classname_site,
195                                         levels=classes)
196     names(a.train1)[names(a.train1) == 'classname_site'] <- 'classname'
197     names(a.test1)[names(a.test1) == 'classname_site'] <- 'classname'
198     a.train2 = subset(train2, classname_site %in% classes,
199                       select = c(colnames4,'classname_site'))

```

```

200     a.test2 = subset(test2, classname_site %in% classes,
201                     select = c(colnames4,'classname_site'))
202     a.train2$classname_site <- factor(a.train2$classname_site,
203                                     levels=classes)
204     a.test2$classname_site <- factor(a.test2$classname_site,
205                                    levels=classes)
206     names(a.train2)[names(a.train2) == 'classname_site'] <- 'classname'
207     names(a.test2)[names(a.test2) == 'classname_site'] <- 'classname'
208     a.train <- rbind(a.train1,a.train2)
209     a.test <- rbind(a.test1,a.test2)
210     #a.train <- a.train1
211     #a.test <- a.train2
212   } else if (classname_site_ind == "classname") {
213     a.train1 = subset(train1, classname %in% classes,
214                       select = c(colnames4,'classname'))
215     a.test1 = subset(test1, classname %in% classes,
216                      select = c(colnames4,'classname'))
217     a.train2 = subset(train2, classname %in% classes,
218                       select = c(colnames4,'classname'))
219     a.test2 = subset(test2, classname %in% classes,
220                      select = c(colnames4,'classname'))
221     a.train1$classname <- factor(a.train1$classname,levels=classes)
222     a.test1$classname <- factor(a.test1$classname,levels=classes)
223     a.train2$classname <- factor(a.train2$classname,levels=classes)
224     a.test2$classname <- factor(a.test2$classname,levels=classes)
225     #a.train <- rbind(a.train1,a.train2)
226     #a.test <- rbind(a.test1,a.test2)
227     a.train <- a.train1
228     a.test <- a.train2
229     #same_other_ind == "other"
230     #classname_site_ind == "classname"
231   }
232 }
233 # head(a.train)
234 # head(a.test)
235 # table(a.train$classname)
236 # table(a.test$classname)
237
238 #return(a.train)
239 #return(a.test)
240 a.varimp <- varimp_rf_unscaled(a.train,a.test,mda_mdg_ind)
241
242 top_x <- min(top_x,length(a.varimp$variables)) #set top_x not more than
243 list of available vars
244 if (model_ind == "RF") {
245   a.conf <- rf.topX.conf(a.varimp,a.train,a.test,mda_mdg_ind,top_x)
246 } else if (model_ind == "SVM") {

```



```

247     a.conf <- svm.topX.conf(a.varimp,a.train,a.test,mda_mdg_ind,top_x)
248   }
249   b <- c(a.varimp$variables,"classname")
250   a.train <- a.train[c(b)]
251   a.test <- a.test[c(b)]
252   return(list(a.train,a.test,a.varimp,a.conf))
253 }
254
255 cw.cw.lq.lcx.dBite.rf.tt.mdg.30 <-
256 apply_Model1_on_Site2_conf_newsite_cw_traintest(cw3000.train,
257 cw3000.test, cw3000.train, cw3000.test,"tt",FW_overlap_CW_DW_Lquad_S,
258 "classname_tt_site", LCX_ind=c("L","C","X"),dBit_ind=c("dB","i","t"),
259 elevYN="Y", model_ind="RF", mda_mdg_ind="MDG", 30)
260
261 cw.cw.lq.lcx.dBite.rf.mdg.30 <-
262 apply_Model1_on_Site2_conf_newsite(cw3000.train,
263 cw3000.train,cw3000.test,cw3000.test,"cw",FW_overlap_CW_DW_Lquad_S,
264 "classname_site",LCX_ind=c("L","C","X"),dBit_ind=c("dB","i","t"), elevYN="Y",
265 model_ind="RF", mda_mdg_ind="MDG",30)
266
267 cw.cd.lcx.dBite.rf.mdg.30 <-
268 apply_Model1_on_Site2_conf_newsite_cw_traintest(cw3000.train,
269 cd3000.train, cw3000.test, cd3000.test, "wetdry", F_overlap_CW_CD_S,
270 "classname_wetdry_site", LCX_ind=c("L","C","X"),dBit_ind=c("dB","i","t"),
271 elevYN="Y", model_ind="RF", mda_mdg_ind="MDG",30)
272
273 cw.cd.classname.lcx.dBite.rf.mdg.30 <-
274 apply_Model1_on_Site2_conf_newsite_cw_traintest(cw3000.train,
275 cd3000.train,cw3000.test,cd3000.test,"wetdry",F_overlap_CW_CD_S,
276 "classname",LCX_ind=c("L","C","X"), dBit_ind=c("dB","i","t"), elevYN="Y",
277 model_ind="RF", mda_mdg_ind="MDG",30)
278
279 cw.dw.classname.lq.lcx.dBite.rf.mdg.30 <-
280 apply_Model1_on_Site2_conf_newsite_cw_traintest(cw3000.train,
281 dw3000.lq.train,cw3000.test,dw3000.lq.test,"other",FW_overlap_CW_DW_L
282 quad_S, "classname",LCX_ind=c("L","C","X"),dBit_ind=c("dB","i","t"),
283 elevYN="Y", model_ind="RF", mda_mdg_ind="MDG",30)
284
285 cw.dw.lq.lcx.dBite.rf.mdg.30 <-
286 apply_Model1_on_Site2_conf_newsite(cw3000.train,
287 dw3000.lq.train,cw3000.test,dw3000.lq.test,"other",
288 FW_overlap_CW_DW_Lquad_S, "classname_site", LCX_ind=c("L","C","X"),
289 dBit_ind=c("dB","i","t"), elevYN="Y",model_ind="RF",
290 mda_mdg_ind="MDG",30)
291

```

## Bibliography

- Ainsworth, T.L., Kelly, J.P. and Lee, J.-S. (2009) Classification comparisons between dual-pol, compact polarimetric and quad-pol SAR imagery. **ISPRS Journal of Photogrammetry and Remote Sensing** [online], 64 (5): 464–471. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0924271609000069> [Accessed 3 October 2013]
- Albalawi, E.K. and Kumar, L. (2013) Using remote sensing technology to detect, model and map desertification: A review. **Journal of Food, Agriculture and Environment**, 11 (2): 791–797
- Anderson, J.R., Hardy, E.E., Roach, J.T., et al. (1976) A Land Use and Land Cover Classification System for Use with Remote Sensor Data. **United States Geological Survey Circular**, (671): 1–28
- Anys, H. (1995) Evaluation of textural and multipolarization radar features for crop classification. **IEEE Transactions on Geoscience and Remote Sensing** [online], 33 (5): 1170–1181. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=469481>
- Arino, O., Bicheron, P., Achard, F., et al. (2008) GLOBCOVER: The most detailed portrait of Earth. **ESA Bulletin**, 136: 24–31
- Armitage, R.P., Alberto Ramirez, F., Mark Danson, F., et al. (2013) Probability of cloud-free observation conditions across Great Britain estimated using MODIS cloud mask. **Remote Sensing Letters** [online], 4 (5): 427–435. Available from: <http://www.tandfonline.com/doi/abs/10.1080/2150704X.2012.744486> [Accessed 14 April 2014]
- Aschbacher, J. and Milagro-Pérez, M.P. (2012) The European Earth monitoring (GMES) programme: Status and perspectives. **Remote Sensing of Environment** [online], 120 (2012): 3–8. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425712000612> [Accessed 12 April 2012]

- Asner, G.P. (2001) Cloud cover in Landsat observations of the Brazilian Amazon. **International Journal of Remote Sensing**, 22 (18): 3855–3862
- Bagan, H., Kinoshita, T. and Yamagata, Y. (2012) Combination of AVNIR-2, PALSAR and Polarimetric Parameters for Land Cover Classification. **IEEE Transactions on Geoscience and Remote Sensing**, 50 (4): 1318–1328
- Baghdadi, N., Aubert, M. and Zribi, M. (2012) Use of TerraSAR-X Data to Retrieve Soil Moisture Over Bare Soil Agricultural Fields. **IEEE Geoscience and Remote Sensing Letters** [online], 9 (3): 512–516. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6088081>
- Baghdadi, N., Cresson, R., Todoroff, P., et al. (2010) Multitemporal Observations of Sugarcane by TerraSAR-X Images. **Sensors** [online], 10 (10): 8899–8919. Available from: <http://www.mdpi.com/1424-8220/10/10/8899/> [Accessed 11 August 2011]
- Banks, S.N., Ullmann, T., Duffe, J., et al. (2012) Multi-frequency analysis of high resolution quad-pol Radarsat-2 and dual-pol TerraSAR-X data for land cover classification in Arctic Coastal Ecosystems, Mackenzie Delta, beaufort sea. **International Geoscience and Remote Sensing Symposium (IGARSS)**, pp. 3548–3551
- Barbosa, R.C. (2012) **Chinese Long March 2C lofts Huanjing-1C into orbit** [online]. Available from: <http://www.nasaspaceflight.com/2012/11/chinese-long-march-2c-huanjing-1c-into-orbit/>
- Barrett, B.W., Dwyer, E. and Whelan, P. (2009) Soil Moisture Retrieval from Active Spaceborne Microwave Observations: An Evaluation of Current Techniques. **Remote Sensing** [online], 1 (3): 210–242. Available from: <http://www.mdpi.com/2072-4292/1/3/210/> [Accessed 1 March 2012]
- Bartholomé, E. and Belward, S. (2005) GLC2000: a new approach to global land cover mapping from Earth observation data. **International Journal of Remote Sensing** [online], 26 (9): 1959–1977. Available from: <http://www.informaworld.com/openurl?genre=article&doi=10.1080/01431160412331291297&magic=crossref> | D404A21C5BB053405B1A640AFFD44AE3 [Accessed 26 July 2011]

- Blaschke, T. (2010) Object based image analysis for remote sensing. **ISPRS Journal of Photogrammetry and Remote Sensing** [online], 65 (1): 2–16. Available from: <http://dx.doi.org/10.1016/j.isprsjprs.2009.06.004>
- Blaschke, T., Hay, G.J., Kelly, M., et al. (2014) Geographic Object-Based Image Analysis - Towards a new paradigm. **ISPRS Journal of Photogrammetry and Remote Sensing** [online], 87: 180–191. Available from: <http://dx.doi.org/10.1016/j.isprsjprs.2013.09.014>
- Boland, M. V. (1999) **Quantitative Description and Automated Classification of Cellular Protein Localization Patterns in Fluorescence Microscope Images of Mammalian Cells** [online]. Carnegie Mellon. Available from: [http://murphylab.web.cmu.edu/publications/boland/boland\\_node26.html](http://murphylab.web.cmu.edu/publications/boland/boland_node26.html)
- Breiman, L. (1996) Bagging Predictors. **Machine Learning**, 24: 123–140
- Breiman, L. (2001) Random Forests. **Machine Learning**, 45: 5–32
- Breiman, L. and Cutler, A. (2001) **Random Forests** [online]. Available from: [http://www.stat.berkeley.edu/~breiman/RandomForests/cc\\_home.htm](http://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm) [Accessed 11 June 2015]
- Breiman, L., Friedman, J.H., Olshen, R.A., et al. (1984) **Classification and Regression Trees**. Belmont, CA: Wadsworth
- Brink, A.B. and Eva, H.D. (2011) The potential use of high-resolution Landsat satellite data for detecting land-cover change in the Greater Horn of Africa. **International Journal of Remote Sensing**, 32 (21): 5981–5995
- Buckley, J.R. (2002) “Environmental Change Detection in Prairie Landscapes with Simulated Radarsat 2 Imagery.” In **IEEE Geoscience and Remote Sensing Symposium (IGARSS)**. Toronto, Canada. 2002. pp. 3255–3257
- Carreiras, J.M.B., Vasconcelos, M.J. and Lucas, R.M. (2012) Understanding the relationship between aboveground biomass and ALOS PALSAR data in the forests of Guinea-Bissau (West Africa). **Remote Sensing of Environment** [online], 121: 426–442. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425712001058> [Accessed 27 March 2012]

- Chakraborty, M., Panigrahy, S., Rajawat, A.S., et al. (2013) Initial results using RISAT-1 C-band SAR data. **Current Science**, 104 (4): 490–501
- Chang, C.-C. and Lin, C.-J. (2011) LIBSVM: A Library for Support Vector Machines. **ACM Transactions on Intelligent Systems and Technology** [online], 2 (3): 1–27. Available from: <http://dl.acm.org/citation.cfm?doid=1961189.1961199> [Accessed 19 March 2014]
- Chen, K.S., Huang, W.P., Tsay, D.H., et al. (1996) Classification of Multifrequency Polarimetric SAR Imagery Using a Dynamic Learning Neural Network. **IEEE Transactions on Geoscience and Remote Sensing**, 34 (3): 814–820
- Clausi, D. a (2002) An analysis of co-occurrence texture statistics as a function of grey level quantization. **Canadian Journal of Remote Sensing** [online], 28 (1): 45–62. Available from: <http://pubs.casi.ca/doi/abs/10.5589/m02-004>
- CONAE (2011) **SAOCOM** [online]. Available from: <http://www.conae.gov.ar/eng/satelites/saocom.html>
- Corcoran, J., Knight, J. and Gallant, A. (2013) Influence of Multi-Source and Multi-Temporal Remotely Sensed and Ancillary Data on the Accuracy of Random Forest Classification of Wetlands in Northern Minnesota. **Remote Sensing** [online], 5 (7): 3212–3238. Available from: <http://www.mdpi.com/2072-4292/5/7/3212/> [Accessed 28 March 2014]
- Cortes, C. and Vapnik, V. (1995) Support-vector networks. **Machine Learning** [online], 20 (3): 273–297. Available from: <http://link.springer.com/10.1007/BF00994018>
- Cotlier, C., Brisco, B., Mondino, M.C., et al. (2011) Use of Radarsat-2 ultra-fine images in horticulture-intensive farming: Land use detection and crop discrimination. **Canadian Journal of Remote Sensing**, 37 (1): 37–44
- Covella, F., Battazza, F., Coletta, A., et al. (2010) COSMO-SkyMed an existing opportunity for observing the Earth. **Journal of Geodynamics** [online], 49 (3-4): 171–180. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0264370710000050> [Accessed 30 January 2012]
- Cruz, V.H. (2010) TERRASAR-X RAPID MAPPING FOR FLOOD EVENTS. **Photogrammetrie -**

- CSA (2011) **RADARSAT-2** [online]. Available from: <http://www.asc-csa.gc.ca/eng/satellites/radarsat2/>
- CSA (2013) **RADARSAT Constellation** [online]. Available from: <http://www.asc-csa.gc.ca/eng/satellites/radarsat/>
- Desnos, Y.-L., Buck, C., Guijarro, J., et al. (2000) "The ENVISAT Advanced Synthetic Aperture Radar System." In **IEEE International Geoscience and Remote Sensing Symposium. 2000.** pp. 1171–1173
- Díaz-Uriarte, R. and Alvarez de Andrés, S. (2006) Gene selection and classification of microarray data using random forest. **BMC bioinformatics**, 7: 3
- Dierking, W., Skriver, H. and Gudmandsen, P. (2003) "Sar polarimetry for sea ice classification." In **Proceedings Workshop Applications SAR Polarimetry Polarimetric Interferometry (POLinSAR). Frascati, Italy. 2003.** pp. 109–118
- Dietterich, T.G. (2000) "Ensemble Methods in Machine Learning." In Kittler, J. and Roli, F. (eds.). **1st International Workshop on Multiple Classifier Systems (MCS 2000).** **Cagliari, Italy. 2000.** Springer-Verlag Berlin. pp. 1–15
- Dobson, M.C., Pierce, L.E. and Ulaby, F.T. (1996) Knowledge-Based Land-Cover Classification Using ERS- 1/JERS- 1 SAR Composites. **IEEE Transactions on Geoscience and Remote Sensing**, 34 (1): 83–99
- Dronova, I. (2015) Object-Based Image Analysis in Wetland Research: A Review. **Remote Sensing** [online], 7 (5): 6380–6413. Available from: <http://www.mdpi.com/2072-4292/7/5/6380/>
- Drusch, M., Del Bello, U., Carlier, S., et al. (2012) Sentinel-2: ESA's Optical High-Resolution Mission for GMES Operational Services. **Remote Sensing of Environment** [online], 120: 25–36. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425712000636> [Accessed 4 December 2013]
- Duro, D.C., Coops, N.C., Wulder, M. a., et al. (2007) Development of a large area biodiversity monitoring system driven by remote sensing. **Progress in Physical Geography**, 31 (3):

- EC COM (2007) European Space Policy. **Communication from the Commission to the Council and the European Parliament**, 212
- Eckardt, R., Berger, C., Thiel, C., et al. (2013) Removal of Optically Thick Clouds from Multi-Spectral Satellite Images Using Multi-Frequency SAR Data. **Remote Sensing** [online], 5 (6): 2973–3006. Available from: <http://www.mdpi.com/2072-4292/5/6/2973/> [Accessed 11 October 2013]
- EEA (2007) **CLC2006 technical guidelines, EEA Technical Report No 17** [online]. Copenhagen. Available from: [http://www.eea.europa.eu/publications/technical\\_report\\_2007\\_17](http://www.eea.europa.eu/publications/technical_report_2007_17)
- EEA (2015) **Copernicus Land Monitoring Services - Corine Land Cover** [online]. Available from: <http://land.copernicus.eu/pan-european/corine-land-cover/#corine-land-cover-legend> [Accessed 1 September 2015]
- Eisfelder, C., Kuenzer, C. and Dech, S. (2012) Derivation of biomass information for semi-arid areas using remote-sensing data. **International Journal of Remote Sensing**, 33 (9): 2937–2984
- Elith, J., Phillips, S.J., Hastie, T., et al. (2011) A statistical explanation of MaxEnt for ecologists. **Diversity and Distributions** [online], 17 (1): 43–57. Available from: <http://doi.wiley.com/10.1111/j.1472-4642.2010.00725.x> [Accessed 11 December 2013]
- Englhart, S., Keuck, V. and Siegert, F. (2011) Aboveground biomass retrieval in tropical forests — The potential of combined X- and L-band SAR data use. **Remote Sensing of Environment** [online], 115 (5): 1260–1271. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425711000216> [Accessed 24 June 2011]
- Englhart, S., Keuck, V. and Siegert, F. (2012) Modeling Aboveground Biomass in Tropical Forests Using Multi-Frequency SAR Data — A Comparison of Methods. **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing**, 5 (1): 298–306

- ESA (2011) **RESPOND Final Report** [online]. Available from:  
[http://www.esa.int/Our\\_Activities/Observing\\_the\\_Earth/Copernicus/Final\\_reports](http://www.esa.int/Our_Activities/Observing_the_Earth/Copernicus/Final_reports)
- ESA (2012a) **ESA Declares end of mission for ENVISAT** [online]. Available from:  
[http://www.esa.int/Our\\_Activities/Observing\\_the\\_Earth/Envisat/ESA\\_declares\\_end\\_of\\_mission\\_for\\_Envisat](http://www.esa.int/Our_Activities/Observing_the_Earth/Envisat/ESA_declares_end_of_mission_for_Envisat) [Accessed 1 April 2015]
- ESA (2012b) "Sentinel data policy and access to data." **In Workshop on GMES Data and Information Policy, 12-13 January 2012. Brussels. 2012.** pp. 12–13
- ESA (2013a) **No 13-2013: ESA's next Earth Explorer Satellite** [online]. Available from:  
[http://www.esa.int/For\\_Media/Press\\_Releases/ESA\\_s\\_next\\_Earth\\_Explorer\\_satellite](http://www.esa.int/For_Media/Press_Releases/ESA_s_next_Earth_Explorer_satellite)
- ESA (2013b) **Sentinel-1** [online]. Available from:  
[http://www.esa.int/Our\\_Activities/Observing\\_the\\_Earth/Copernicus/Sentinel-1](http://www.esa.int/Our_Activities/Observing_the_Earth/Copernicus/Sentinel-1)
- ESA (2014) **Initial opening of the Sentinel-1 dataflow to the scihub.** [online]. Available from:  
<https://scihub.esa.int/twiki/do/view/SciHubNews/News00007> [Accessed 3 April 2015]
- EU 4th Space Council (2007) **Resolution on the European Space Policy** [online]. Brussels.  
 Available from:  
[http://www.consilium.europa.eu/ueDocs/cms\\_Data/docs/pressData/en/intm/94166.pdf](http://www.consilium.europa.eu/ueDocs/cms_Data/docs/pressData/en/intm/94166.pdf)
- EU Lisbon Treaty (2007) EU Lisbon Treaty. **Official Journal of the European Union**, 50 (C306)
- European Commission (2013) Copernicus: The EU Earth Observation Programme - good for jobs and the environment ! **European Commission Press Release.** (May 2013)
- European Commission (2013) **Copernicus, The European Earth Observation Programme** [online]. Available from: <http://www.copernicus.eu/>
- Evans, T.L. and Costa, M. (2013) Landcover classification of the Lower Nhecolândia subregion of the Brazilian Pantanal Wetlands using ALOS/PALSAR, RADARSAT-2 and ENVISAT/ASAR imagery. **Remote Sensing of Environment** [online], 128: 118–137.  
 Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425712003823>  
 [Accessed 12 March 2013]



- Evans, T.L., Costa, M., Telmer, K., et al. (2010) Using ALOS / PALSAR and RADARSAT-2 to Map Land Cover and Seasonal Inundation in the Brazilian Pantanal. **Earth**, 3 (4): 560–575
- FAO, UNDP and UNEP (2008) **UN collaborative programme on reducing emissions from deforestation and forest degradation in developing countries (UN-REDD)**
- Ferrazzoli, P., Paloscia, S., Pampaloni, P., et al. (1997) The Potential of Multifrequency Polarimetric SAR in Assessing Agricultural and Arboreous Biomass. **IEEE Transactions on Geoscience and Remote Sensing**, 35 (1): 5–17
- Fielding, A.H. and Bell, J.F. (1997) A review of methods for the assessment of prediction errors in conservation presence/absence models. **Environmental Conservation**, 24: 38–49
- Flament, P. (2004) Galileo: A new dimension in international satellite navigation. **Proceedings International Symposium Electronics in Marine**, (June): 6–14
- Flett, D., De Abreu, R., Arkett, M., et al. (2008) Initial evaluation of RADARSAT-2 for operational sea ice monitoring. **International Geoscience and Remote Sensing Symposium (IGARSS)**, 1 (1): 9–12
- Freeman, A. and Durden, S.L. (1998) A three-component scattering model for polarimetric SAR data. **IEEE Transactions on Geoscience and Remote Sensing**, 36 (3): 963–973
- Freund, Y. (1995) Boosting a weak learning algorithm by majority. **Information and Computation**, 121 (2): 256–285
- Freund, Y. and Schapire, R. (1997) A decision-theoretic generalization of on-line learning and an application to boosting. **Journal of Computer and System Sciences** [online], 55 (1): 119–139. Available from: [http://link.springer.com/chapter/10.1007/3-540-59119-2\\_166](http://link.springer.com/chapter/10.1007/3-540-59119-2_166)
- Freund, Y. and Schapire, R.E. (1996) “Experiments with a new boosting algorithm.” In **Machine Learning: Proceedings of the Thirteenth International Conference (ICML ’96)**. 1996. pp. 1–9
- Friedl, M.A., Sulla-Menashe, D., Tan, B., et al. (2010) MODIS Collection 5 global land cover: Algorithm refinements and characterization of new datasets. **Remote Sensing of Environment** [online], 114 (1): 168–182. Available from:

<http://linkinghub.elsevier.com/retrieve/pii/S0034425709002673> [Accessed 15 June 2011]

Fritz, T. and Eineder, M. (2010) **TerraSAR-X Ground Segment Basic Product Specification Document**

FSF (2014a) **GNU Bash**. [online]. Available from: <http://www.gnu.org/software/bash/> [Accessed 18 December 2015]

FSF (2014b) **The GNU Bash Reference Manual** [online]. Available from: <https://www.gnu.org/software/bash/manual/bashref.html> [Accessed 18 December 2015]

Gardini, B., Graf, G. and Ratier, G. (1995) The Instruments on ENVISAT. **Acta Astronautica**, 37 (95): 301–311

GCOS (2010) Implementation Plan for the Global Observing System for Climate in Support of the UNFCCC (2010 Update). **GCOS-138** [online], 138 (1523): 1–180. Available from: <http://eprints.soton.ac.uk/162953/>

GDAL (2013) **GDAL - Geospatial Data Abstraction Library**. [online]. Available from: <http://www.gdal.org>

Gibbs, H.K., Brown, S., Niles, J.O., et al. (2007) Monitoring and estimating tropical forest carbon stocks: making REDD a reality. **Environmental Research Letters** [online], 2: 045023(1–13). Available from: <http://stacks.iop.org/1748-9326/2/i=4/a=045023?key=crossref.4118e8af5a9a3ac02c1bb32f8a92c50f> [Accessed 17 July 2012]

Giudici, D., Maggi, L., Guarnieri, a. M., et al. (2010) Roll-steering for improving SAOCOM-SAR performances. **2010 IEEE Radar Conference** [online], pp. 752–755. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5494522>

De Grandi, G., Bouvet, A., Lucas, R.M., et al. (2011) The K&C PALSAR mosaic of the African continent: processing issues and first thematic results. **IEEE Transactions on Geoscience and Remote Sensing** [online], 49 (10): 3593–3610. Available from: [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=6022766](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6022766) [Accessed 16 November 2011]

- Di Gregorio, A. and Latham, J. (2002) AFRICOVER Land Cover Classification and Mapping Project. **Encyclopedica of Life Support Systems (EOLSS)**. I
- Di Grigorio, A. (2005) **Land Cover Classification System - Classification concepts and user manual for Software version 2**. Rome: FAO Environment and Natural Resources Service Series
- Gromski, P.S., Xu, Y., Correa, E., et al. (2014) A comparative investigation of modern feature selection and classification approaches for the analysis of mass spectrometry data. **Analytica Chimica Acta** [online], 829: 1–8. Available from: <http://dx.doi.org/10.1016/j.aca.2014.03.039>
- Hajnsek, I., Pottier, E. and Cloude, S.R. (2003) Inversion of Surface Parameters From Polarimetric SAR. **IEEE Transactions on Geoscience and Remote Sensing**, 41 (4): 727–744
- Hapfelmeier, A., Hothorn, T., Ulm, K., et al. (2014) A new variable importance measure for random forests with missing data. **Statistics and Computing**, 24 (1): 21–34
- Haralick, R.M., Shanmugam, K. and Dinstein, I. (1973) Textural Features for Image Classification. **IEEE Transactions on Systems, Man, and Cybernetics** [online], 3 (6): 610–621. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4309314>
- Hello, D. (2010) **GMES Emergency Management Service as developed by SAFER** [online]. Available from: [http://www.copernicus.eu/fileadmin/files/4. GMES Services/GMES\\_Emergency\\_Management\\_Service\\_Portfolio\\_19Nov10.pdf](http://www.copernicus.eu/fileadmin/files/4_GMES_Services/GMES_Emergency_Management_Service_Portfolio_19Nov10.pdf)
- Henry, J. -B., Chastanet, P., Fellah, K., et al. (2006) Envisat multi-polarized ASAR data for flood mapping. **International Journal of Remote Sensing** [online], 27 (10): 1921–1929. Available from: <http://www.tandfonline.com/doi/abs/10.1080/01431160500486724> [Accessed 16 November 2011]
- Herold, N.D., Haack, B.N. and Solomon, E. (2004) An evaluation of radar texture for land use/cover extraction in varied landscapes. **International Journal of Applied Earth Observation and Geoinformation** [online], 5 (2): 113–128. Available from:

<http://linkinghub.elsevier.com/retrieve/pii/S0303243404000170> [Accessed 13 December 2013]

Heumann, B.W. (2011) Satellite remote sensing of mangrove forests: Recent advances and future opportunities. **Progress in Physical Geography**, 35 (1): 87–108

Hisdesat (2013) **Paz** [online]. Available from: [http://www.hisdesat.es/eng/satelites\\_observ-paz.html](http://www.hisdesat.es/eng/satelites_observ-paz.html)

Hobbs, S.E., Ang, W. and Seynat, C. (1998) Wind and Rain Effects on SAR Backscatter from Crops. **Retrieval of Bio- and Geo-Physical Parameters from SAR Data for Land Applications Workshop**, 44 (0): 1–5

Hölbling, D., Füreder, P., Antolini, F., et al. (2012) A semi-automated object-based approach for landslide detection validated by persistent scatterer interferometry measures and landslide inventories. **Remote Sensing**, 4 (5): 1310–1336

Holecz, F., Barbieri, M., Cantone, A., et al. (2009) “SYNERGETIC USE OF MULTI - TEMPORAL ALOS PALSAR AND ENVISAT ASAR DATA II - 5.” In **IGARSS 2009**. 2009. pp. 5–8

Horn, R., Anton, N., Andreas, R., et al. (2009) F-SAR – DLR ’ S NEW MULTIFREQUENCY POLARIMETRIC AIRBORNE SAR. **IEEE International Geoscience and Remote Sensing Symposium (IGARSS)**, 2: 902–905

Huang, C., Davis, L.S. and Townshend, J.R.G. (2002) An assessment of support vector machines for land cover classification. **International Journal of Remote Sensing** [online], 23 (4): 725–749. Available from: <http://www.tandfonline.com/doi/abs/10.1080/01431160110040323> [Accessed 24 July 2012]

Huang, W., Chen, F., Yang, J., et al. (2010) “Chinese HJ-1C SAR and Its Wind Mapping Capability.” In [Http://earth.eo.esa.int/workshops/seasar2010/participants/276/pres\\_276\\_Huang.pdf](http://earth.eo.esa.int/workshops/seasar2010/participants/276/pres_276_Huang.pdf) (ed.). **The 3rd International Workshop on Advances in SAR Oceanography from Envisat, ERS and ESA third party missions. Frascati, Italy**. 2010. pp. 1–19

Ishwaran, H. (2015) The effect of splitting on random forests. **Machine Learning**, 99: 75–118

IUCN (2015) **The World Database on Protected Areas (WDPA)**. [online]. Available from:

www.protectedplanet.net [Accessed 17 May 2015]

Janitza, S., Strobl, C. and Boulesteix, A.-L. (2013) An AUC-based permutation variable importance measure for random forests. **BMC bioinformatics** [online], 14 (1): 119. Available from: <http://www.biomedcentral.com/1471-2105/14/119> \n<http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3626572&tool=pmcentrez&rendertype=abstract>

Janoth, J., Gantert, S., Koppe, W., et al. (2012) TERRASAR-X2 – MISSION OVERVIEW. **IEEE Geoscience and Remote Sensing Symposium (IGARSS)**, pp. 217–220

JAXA (2006) **ALOS - Overview and Objectives**

JAXA (2011) **DAICHI (ALOS) Operation Completion** [online]. Available from: [http://www.jaxa.jp/press/2011/05/20110512\\_daichi\\_e.html](http://www.jaxa.jp/press/2011/05/20110512_daichi_e.html)

JAXA (2013a) **ALOS-2** [online]. Available from: <https://directory.eoportal.org/web/eoportal/satellite-missions/a/alos-2>

JAXA (2013b) **ALOS-2 Launch Schedule** [online]. Available from: [http://www.jaxa.jp/projects/in\\_progress\\_e.html](http://www.jaxa.jp/projects/in_progress_e.html)

JAXA (2014a) **Advanced Land Observing Satellite-2 “DAICHI-2” (ALOS-2) Regular Provision of Observation Data** [online]. Available from: [http://global.jaxa.jp/press/2014/11/20141125\\_daichi2.html](http://global.jaxa.jp/press/2014/11/20141125_daichi2.html) [Accessed 3 April 2015]

JAXA (2014b) **Launch Result of H-IIA Launch Vehicle No. 24 with “DAICHI-2” (ALOS-2) onboard** [online]. Available from: [http://global.jaxa.jp/press/2014/05/20140524\\_h2af24.html](http://global.jaxa.jp/press/2014/05/20140524_h2af24.html) [Accessed 3 April 2015]

Jensen, J.R. (2005) **Introductory Digital Image Processing: A Remote Sensing Perspective**. Third Edit. Kaveney, D.E. and Challice, J. (eds.). NJ: Pearson Prentice Hall

Joyce, K.E., Belliss, S.E., Samsonov, S. V., et al. (2009) A review of the status of satellite remote sensing and image processing techniques for mapping natural hazards and disasters. **Progress in Physical Geography** [online], 33 (2): 183–207. Available from: <http://ppg.sagepub.com/cgi/doi/10.1177/0309133309339563> [Accessed 7 August 2011]

JPL (2014) **U.S. Releases Enhanced Shuttle Land Elevation Data** [online]. Available from:

<http://www.jpl.nasa.gov/news/news.php?release=2014-321>

- Ju, J. and Roy, D.P. (2008) The availability of cloud-free Landsat ETM + data over the conterminous United States and globally. **Remote Sensing of Environment**, 112: 1196–1211
- Kamwenda, G.J. (2002) Ngitili agrosilvipastoral systems in the United Republic of Tanzania. **Unasylva**, 53 (4): 46–50
- Karatzoglou, A., Meyer, D. and Hornik, K. (2006) Support Vector Machines in R. **Journal of Statistical Software**, 15 (9): 1–28
- Kouskoulas, Y., Ulaby, F.T. and Pierce, L.E. (2004) The Bayesian Hierarchical Classifier ( BHC ) and Its Application to Short Vegetation Using Multifrequency Polarimetric SAR. **IEEE Transactions on Geoscience and Remote Sensing**, 42 (2): 469–477
- Kullback, S. (1987) Letters to the Editor: The Kullback Leibler distance. **The American Statistician**, 41 (4): 340
- Kullback, S. and Leibler, R.A. (1951) On Information and Sufficiency. **The Annals of Mathematical Statistics**, 22 (1): 79–86
- Kumar, D., Rao, S. and Sharma, J.R. (2015) Radar Vegetation Index as an Alternative to NDVI for Monitoring of Soyabean and Cotton. **Proceedings of the XXXIII INCA International Congress (Indian Cartographer), 19-21 September, 2013, Jodhpur, India**, (September 2013): 91–96
- Kuntz, S. (2010) Potential of spaceborne SAR for monitoring the tropical environments. **Tropical Ecology**, 51 (1): 3–10
- Kuplich, T.M., Curran, P.J. and Atkinson, P.M. (2005) Relating SAR image texture to the biomass of regenerating tropical forests. **International Journal of Remote Sensing** [online], 26 (21): 4829–4854. Available from: <http://www.tandfonline.com/doi/abs/10.1080/01431160500239107> [Accessed 25 November 2013]
- Kwarteng, A.Y., Dobson, M.C., Kellendorfer, J., et al. (2008) SAR-based land cover classification of Kuwait. **International Journal of Remote Sensing** [online], 29 (23): 6739–6778. Available from:

<http://www.tandfonline.com/doi/abs/10.1080/01431160802209097> [Accessed 2 July 2012]

- Lardeux, C., Frison, P., Tison, C., et al. (2009) Support Vector Machine for Multifrequency SAR Polarimetric Data Classification. **IEEE Transactions on Geoscience and Remote Sensing**, 47 (12): 4143–4152
- Lee, J. and Ainsworth, T.L. (2010) “AN OVERVIEW OF RECENT ADVANCES IN POLARIMETRIC SAR INFORMATION EXTRACTION : ALGORITHMS AND APPLICATIONS.” In **Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International**. 2010. pp. 851–854
- Lewis, A.J., Henderson, F.M. and Holcomb, D.W. (1998) “Radar Fundamentals: The Geoscience Perspective.” In Henderson, F.M. and Lewis, A.J. (eds.) **Principles and Applications of Imaging Radar**. John Wiley & Sons, Inc. pp. 131–181
- Li, G., Lu, D., Moran, E., et al. (2012) A comparative analysis of ALOS PALSAR L-band and RADARSAT-2 C-band data for land-cover classification in a tropical moist region. **ISPRS Journal of Photogrammetry and Remote Sensing** [online], 70: 26–38. Available from: <http://dx.doi.org/10.1016/j.isprsjprs.2012.03.010>
- Li, S., Harner, E.J. and Adjeroh, D. a (2011) Random KNN feature selection - a fast and stable alternative to Random Forests. **BMC Bioinformatics** [online], 12 (1): 450. Available from: <http://www.biomedcentral.com/1471-2105/12/450>
- Liaw, A. and Wiener, M. (2002) Classification and Regression by randomForest. **R News** [online], 2 (3): 18–22. Available from: [http://www.r-project.org/doc/Rnews/Rnews\\_2002-3.pdf](http://www.r-project.org/doc/Rnews/Rnews_2002-3.pdf)
- Lillesand, T.M., Kiefer, R.W. and Chipman, J.W. (2004) **Remote Sensing and Image Interpretation**. Fifth Edit. Flahive, R., Powell, D. and Pigliucci, N.M. (eds.). New York: John Wiley & Sons, Inc.
- Longepe, N., Rakwatin, P., Isoguchi, O., et al. (2010) “ON THE USE OF SUPPORT VECTOR MACHINES FOR LAND COVER ANALYSIS WITH L-BAND SAR DATA.” In **IEEE Geoscience and Remote Sensing Symposium (IGARSS)**. Honolulu. 2010. pp. 3263–3266
- Lönnqvist, A., Rauste, Y., Molinier, M., et al. (2010) Polarimetric SAR Data in Land Cover

Mapping in Boreal Zone. **IEEE Transactions on Geoscience and Remote Sensing**, 48 (10): 3652–3662

Loosvelt, L., Peters, J., Skriver, H., et al. (2012) Random Forests as a tool for estimating uncertainty at pixel-level in SAR image classification. **International Journal of Applied Earth Observation and Geoinformation** [online], 19: 173–184. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0303243412001195> [Accessed 11 October 2013]

Louet, J. and Bruzzi, S. (1999) “ENVISAT Mission and System.” In **IEEE Geoscience and Remote Sensing Symposium (IGARSS). Hamburg, Germany. 1999.** pp. 1680–1682

Loveland, T.R. and Belward, A.S. (1997) The IGBP-DIS global 1km land cover data set , DISCover : First results. **International Journal of Remote Sensing**, 18 (15): 3289–3295

Lucas, R., Carreiras, J.M.B., Bunting, P., et al. (2007) “Pre-processing and geocoding of ALOS PALSAR data over Queensland , Australia.” In **The First Joint PI Symposium of ALPS Data Nodes for ALOS Science Program in Kyoto. Kyoto. 2007** [online]. pp. 1–3. Available from: [http://www.eorc.jaxa.jp/ALOS/en/conf/Proc\\_Plsymp2007/contents/proceedings/Kyoto\\_and\\_Carbon/KC07.pdf](http://www.eorc.jaxa.jp/ALOS/en/conf/Proc_Plsymp2007/contents/proceedings/Kyoto_and_Carbon/KC07.pdf)

Lucas, R.M., Armston, J. and Fairfax, R. (2010) An Evaluation of the ALOS PALSAR L-Band Backscatter — Above Ground Biomass Relationship Queensland , Australia : Impacts of Surface Moisture Condition and Vegetation Structure. **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing** [online], 3 (4): 576–593. Available from: [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=5658192](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5658192) [Accessed 2 December 2011]

Lucas, R.M. and Armston, J.D. (2007) ALOS PALSAR for characterizing wooded savannas in Northern Australia. **Geoscience and Remote Sensing Symposium, 2007. IGARSS 2007. IEEE International** [online], (AUGUST 2007): 3610–3613. Available from: <http://ieeexplore.ieee.org/ielx5/4422707/4422708/04423627.pdf?tp=&arnumber=4423627&isnumber=4422708>

Lucas, R.M., Cronin, N., Lee, A., et al. (2006) Empirical relationships between AIRSAR  
444



backscatter and LiDAR-derived forest biomass, Queensland, Australia. **Remote Sensing of Environment** [online], 100 (3): 407–425. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S003442570500355X> [Accessed 4 April 2014]

Mahmood, A. (2015) RADARSAT-1 Background Mission Implementation and Accomplishments. **Canadian Journal of Remote Sensing** [online], 40 (6): 385–395. Available from: <http://www.tandfonline.com/doi/abs/10.1080/07038992.2014.999913>

Mahmood, A., Cubero-Castan, E., Bequignon, J., et al. (2005) International Charter 'Space and Major Disasters' Status Report. **Proceedings. 2005 IEEE International Geoscience and Remote Sensing Symposium, 2005. IGARSS '05.** [online], 6: 4362–4365. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1525885>

Malenovsky, Z., Rott, H., Cihlar, J., et al. (2012) Sentinels for science: Potential of Sentinel-1, -2, and -3 missions for scientific observations of ocean, cryosphere, and land. **Remote Sensing of Environment** [online], 120: 91–101. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425712000648> [Accessed 16 December 2013]

Marti-Cardona, B., Lopez-Martinez, C., Dolz-Ripolles, J., et al. (2010) ASAR polarimetric, multi-incidence angle and multitemporal characterization of Doñana wetlands for flood extent monitoring. **Remote Sensing of Environment** [online], 114 (11): 2802–2815. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425710002051> [Accessed 23 June 2011]

Martinez, J. and Le Toan, T. (2007) Mapping of flood dynamics and spatial distribution of vegetation in the Amazon floodplain using multitemporal SAR data. **Remote Sensing of Environment** [online], 108 (3): 209–223. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425706004585> [Accessed 11 January 2011]

Martone, M., Rizzoli, P., Bräutigam, B., et al. (2013) First 2 years of TanDEM-X mission: 445

- Interferometric performance overview. **Radio Science**, 48 (5): 617–627
- Massonnet, D. and Feigl, K.L. (1998) Radar interferometry and its application to changes in the Earth's surface. **Reviews of Geophysics**, 36 (4): 441–500
- Mathiassen, J.R., Skavhaug, A. and Bø, K. (2002) Kullback-Leibler Divergence between Gamma Distributions. **ECCV 2002 - European Conference on Computer Vision Proceedings**, pp. 133–147
- Mayaux, P., De Grandi, G.F., Rauste, Y., et al. (2002) Large-scale vegetation maps derived from the combined L-band GRFM and C-band CAMP wide area radar mosaics of Central Africa. **International Journal of Remote Sensing**, 23 (7): 1261–1282
- McNairn, H. and Brisco, B. (2004) The application of C-band polarimetric SAR for agriculture: A review. **Canadian Journal of Remote Sensing**, 30 (3): 525–542
- Metternicht, G., Hurni, L. and Gogu, R. (2005) Remote sensing of landslides: An analysis of the potential contribution to geo-spatial systems for hazard assessment in mountainous environments. **Remote Sensing of Environment** [online], 98 (2-3): 284–303. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425705002506> [Accessed 12 August 2012]
- Mitchard, E.T. a., Saatchi, S.S., Lewis, S.L., et al. (2011) Measuring biomass changes due to woody encroachment and deforestation/degradation in a forest–savanna boundary region of central Africa using multi-temporal L-band radar backscatter. **Remote Sensing of Environment** [online], 115 (11): 2861–2873. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425711001337> [Accessed 23 June 2011]
- Mitchard, E.T.A., Saatchi, S.S., Woodhouse, I.H., et al. (2009) Using satellite radar backscatter to predict above-ground woody biomass: A consistent relationship across four different African landscapes. **Geophysical Research Letters** [online], 36 (23): 1–6. Available from: <http://www.agu.org/pubs/crossref/2009/2009GL040692.shtml> [Accessed 31 July 2012]

- Moran, M.S., Alonso, L., Moreno, J.F., et al. (2012) A RADARSAT-2 Quad-Polarized Time Series for Monitoring Crop and Soil Conditions in Barrax, Spain. **IEEE Transactions on Geoscience and Remote Sensing** [online], 50 (4): 1057–1070. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6034522>
- Morena, L.C., James, K. V. and Beck, J. (2004) An introduction to the RADARSAT-2 mission. **Canadian Journal of Remote Sensing**, 30 (3): 221–234
- Moser, G., Zerubia, J. and Serpico, S.B. (2006) Dictionary-Based Stochastic Expectation – Maximization for SAR Amplitude Probability Density Function Estimation. **IEEE Transactions on Geoscience and Remote Sensing**, 44 (1): 188–200
- Mountrakis, G., Im, J. and Ogole, C. (2011) Support vector machines in remote sensing: A review. **ISPRS Journal of Photogrammetry and Remote Sensing** [online], 66 (3): 247–259. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0924271610001140> [Accessed 19 March 2014]
- Nagendra, H., Lucas, R., Honrado, J.P., et al. (2013) Remote sensing for conservation monitoring: Assessing protected areas, habitat extent, habitat condition, species diversity, and threats. **Ecological Indicators** [online], 33: 45–59. Available from: <http://dx.doi.org/10.1016/j.ecolind.2012.09.014>
- Naidoo, L., Cho, M. a., Mathieu, R., et al. (2012) Classification of savanna tree species, in the Greater Kruger National Park region, by integrating hyperspectral and LiDAR data in a Random Forest data mining environment. **ISPRS Journal of Photogrammetry and Remote Sensing** [online], 69: 167–179. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0924271612000597> [Accessed 23 April 2012]
- Naidoo, L., Mathieu, R., Main, R., et al. (2015) Savannah woody structure modelling and mapping using multi-frequency ( X- , C- and L-band ) Synthetic Aperture Radar data. **ISPRS JOURNAL OF PHOTOGRAMMETRY AND REMOTE SENSING** [online], 105: 234–250. Available from: <http://dx.doi.org/10.1016/j.isprsjprs.2015.04.007>
- Nascimento, W.R., Souza-Filho, P.W.M., Proisy, C., et al. (2012) Mapping changes in the largest continuous Amazonian mangrove belt using object-based classification of

- multisensor satellite imagery. **Estuarine, Coastal and Shelf Science**, 117: 83–93
- Neteler, M., Bowman, M.H., Landa, M., et al. (2012) GRASS GIS: A multi-purpose open source GIS. **Environmental Modelling & Software** [online], 31: 124–130. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S1364815211002775> [Accessed 22 November 2013]
- Newton, A.C., Hill, R. a., Echeverría, C., et al. (2009) Remote sensing and the future of landscape ecology. **Progress in Physical Geography** [online], 33 (4): 528–546. Available from: <http://ppg.sagepub.com/cgi/content/abstract/33/4/528>
- Niu, X., Ban, Y. (2010) “Multitemporal RADARSAT-2 polarimetric SAR data for urban land-cover mapping.” In **ISPRS Symposium, Vienna, Austria, July 5-7 2010. 2010** [online]. pp. 175–180. Available from: <http://link.aip.org/link/?PSISDG/7841/78410N/1> [Accessed 25 November 2011]
- Okhimamhe, A.A. (2003) ERS SAR interferometry for land cover mapping in a savanna area in Africa. **International Journal of Remote Sensing** [online], 24 (18): 3583–3594. Available from: <http://www.tandfonline.com/doi/abs/10.1080/0143116021000024230> [Accessed 31 August 2011]
- Pacifici, F., Chini, M. and Emery, W.J. (2009) A neural network approach using multi-scale textural metrics from very high-resolution panchromatic imagery for urban land-use classification. **Remote Sensing of Environment** [online], 113 (6): 1276–1292. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425709000625> [Accessed 1 August 2012]
- Pal, M. and Foody, G.M. (2010) Feature selection for classification of hyperspectral data by SVM. **IEEE Transactions on Geoscience and Remote Sensing**, 48 (5): 2297–2307
- Paloscia, S. (2012) Combining L and X band SAR data for estimating biomass and soil moisture of agricultural fields. **European Journal of Remote Sensing** [online], 45 (1): 99–109. Available from: <http://www.aitjournal.com/articleView.aspx?ID=527> [Accessed 8 November 2013]
- Park, N. -W. and Chi, K. -H. (2008) Integration of multitemporal/polarization C-band SAR

- data sets for land-cover classification. **International Journal of Remote Sensing** [online], 29 (16): 4667–4688. Available from: <http://www.tandfonline.com/doi/abs/10.1080/01431160801947341> [Accessed 9 April 2014]
- Patenaude, G., Milne, R. and Dawson, T.P. (2005) Synthesis of remote sensing approaches for forest carbon estimation: reporting to the Kyoto Protocol. **Environmental Science & Policy** [online], 8 (2): 161–178. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S1462901105000158> [Accessed 13 March 2012]
- Peduzzi, P., Dao, H., Herold, C., et al. (2009) Assessing global exposure and vulnerability towards natural hazards: the Disaster Risk Index. **Natural Hazards and Earth System Science** [online], 9 (4): 1149–1159. Available from: <http://www.nat-hazards-earth-syst-sci.net/9/1149/2009/>
- Peng, X., Wang, J. and Zhang, Q. (2005) Deriving terrain and textural information from stereo RADARSAT data for mountainous land cover mapping. **International Journal of Remote Sensing**, 26 (22): 5029–5049
- Phillips, S. and Dudik, M. (2008) Modeling of species distributions with Maxent: new extensions and a comprehensive evaluation. **Ecography**, 31: 161–175
- Phillips, S.J., Avenue, P. and Park, F. (2004) “A Maximum Entropy Approach to Species Distribution Modeling.” In **Proceedings of the Twenty-First International Conference on Machine Learning**. Banff, Canada. 2004. pp. 655–662
- Piatetsky-Shapiro, G. (2007) **KDnuggets Data Mining Methodology Poll** [online]. Available from: [http://www.kdnuggets.com/polls/2007/data\\_mining\\_methodology.htm](http://www.kdnuggets.com/polls/2007/data_mining_methodology.htm) [Accessed 17 December 2013]
- Pierce, L.E., Bergen, K., Dobson, M.C., et al. (1995) Land-Cover Classification using SIR-C / X-SAR Data. **IEEE Geoscience and Remote Sensing Symposium (IGARSS)**, 2 (313): 8–10
- Pierce, L.E., Bergen, K.M., Dobson, M.C., et al. (1998) Multitemporal Land-Cover Classification Using SIR-C / X-SAR Imagery. **Remote Sensing of Environment**, 64 (1): 20–33

- Pierdicca, N., Pelliccia, F. and Chini, M. (2011) THEMATIC MAPPING AT REGIONAL SCALE USING SIASGE RADAR DATA AT X AND L BAND AND OPTICAL IMAGES. **IEEE Geoscience and Remote Sensing Symposium (IGARSS)**, pp. 1095–1098
- Platt, J.C. (1998) “Fast Training of Support Vector Machines Using Sequential Minimal Optimization.” In Smola, A., Bartlett, P., Scholkopf, B., et al. (eds.) **Advances in Kernel Methods - Support Vector Learning**. Cambridge, MA: MIT Press. pp. 185–208
- Prasad, A.M., Iverson, L.R. and Liaw, A. (2006) Newer classification and regression tree techniques: Bagging and random forests for ecological prediction. **Ecosystems**, 9 (2): 181–199
- Pulvirenti, L., Chini, M., Pierdicca, N., et al. (2011) Flood monitoring using multi-temporal COSMO-SkyMed data: Image segmentation and signature interpretation. **Remote Sensing of Environment** [online], 115 (4): 990–1002. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425710003408> [Accessed 23 August 2011]
- Qi, Z., Yeh, A., Li, X., et al. (2010) “Land use and land cover classification using RADARSAT-2 polarimetric SAR image.” In **Proceedings of the ISPRS Technical Commission VII Symposium: 100 Years ISPRS Advancing Remote Sensing Science. 2010** [online]. pp. 198–203. Available from: [http://www.isprs.org/proceedings/XXXVIII/part7/a/pdf/198\\_XXXVIII-part7A.pdf](http://www.isprs.org/proceedings/XXXVIII/part7/a/pdf/198_XXXVIII-part7A.pdf) [Accessed 15 November 2011]
- Qin, X., Zou, H., Zhou, S., et al. (2015) Region-Based Classification of SAR Images Using Kullback – Leibler Distance Between Generalized Gamma Distributions. **IEEE Geoscience and Remote Sensing Letters**, 12 (8): 1655–1659
- Quegan, S., Toan, T. Le, Skriver, H., et al. (2003) “Crop classification with multitemporal polarimetric sar data.” In **Proceedings Workshop Applications SAR Polarimetry Polarimetric Interferometry (POLinSAR). Frascati, Italy. 2003**. pp. 1–10
- Raney (1998) “Radar Fundamentals: Technical Perspective.” In Henderson, F.M. and Lewis, A.J. (eds.) **Principles and Applications of Imaging Radar**. 3rd ed. John Wiley & Sons, Inc. pp. 9–130

- R-Core-Team (2014) **R: A Language and Environment for Statistical Computing** [online].  
Available from: <http://www.r-project.org> [Accessed 11 June 2015]
- RESTEC (2014) **Announcement for ALOS AVNIR-2 and PALSAR Data Distribution** [online].  
Available from: <http://www.alos-restec.jp/en/article.php/20141121> [Accessed 16 June 2015]
- Riegler, G., Hennig, S.D. and Weber, M. (2015) WorldDEM – a Novel Global Foundation Layer. **ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences** [online], XL-3/W2 (March): 183–187. Available from: <http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-3-W2/183/2015/>
- Rodriguez-Galiano, V.F., Chica-Olmo, M., Abarca-Hernandez, F., et al. (2012a) Random Forest classification of Mediterranean land cover using multi-seasonal imagery and multi-seasonal texture. **Remote Sensing of Environment** [online], 121: 93–107. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425711004408> [Accessed 9 April 2014]
- Rodriguez-Galiano, V.F., Ghimire, B., Rogan, J., et al. (2012b) An assessment of the effectiveness of a random forest classifier for land-cover classification. **ISPRS Journal of Photogrammetry and Remote Sensing** [online], 67: 93–104. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0924271611001304> [Accessed 20 March 2012]
- Rosenqvist, a., Shimada, M., Suzuki, S., et al. (2014) Operational performance of the ALOS global systematic acquisition strategy and observation plans for ALOS-2 PALSAR-2. **Remote Sensing of Environment** [online], 155: 3–12. Available from: <http://dx.doi.org/10.1016/j.rse.2014.04.011>
- Rosenqvist, A. and De Grandi, F. (2009) “THE ALOS PALSAR MOSAIC OVER THE AFRICAN CONTINENT – A REFERENCE BASELINE DATASET FOR FOREST- AND LAND COVER CHANGE MONITORING.” In **IEEE International Geoscience and Remote Sensing Symposium. 2009.** pp. V–115 – V–117
- Rosenqvist, A., Shimada, M., Ito, N., et al. (2007) ALOS PALSAR: A Pathfinder Mission for Global-Scale Monitoring of the Environment. **IEEE Transactions on Geoscience and**

- Remote Sensing** [online], 45 (11): 3307–3316. Available from:  
<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4358862>
- Rowland, C.S., Balzter, H., Dawson, T.P., et al. (2008) Airborne SAR monitoring of tree growth in a coniferous plantation. **International Journal of Remote Sensing**, 29 (13): 3873–3889
- Ruecker, G. and Siegert, F. (2000) BURN SCAR MAPPING AND FIRE DAMAGE ASSESSMENT USING ERS-2 SAR IMAGES IN EAST KALIMANTAN , INDONESIA. **International Archives of Photogrammetry and Remote Sensing**, XXXIII: 1286–1293
- Saeys, Y., Inza, I. and Larrañaga, P. (2007) A review of feature selection techniques in bioinformatics. **Bioinformatics**, 23 (19): 2507–2517
- SAFER (2012) **GMES Emergency Response Service** [online]. Available from:  
[http://gmesdata.esa.int/web/gsc/core\\_services/emergency](http://gmesdata.esa.int/web/gsc/core_services/emergency)
- Schapire, R.E. (1990) The Strength of Weak Learnability. **Machine Learning**, 227 (5): 197–227
- Schmullius, C.C. and Evans, D.L. (1997) Synthetic aperture radar (SAR) frequency and polarization requirements for applications in ecology, geology, hydrology, and oceanography: A tabular status quo after SIR-C/X-SAR. **International Journal of Remote Sensing** [online], 18 (13): 2713–2722. Available from:  
<http://www.tandfonline.com/doi/abs/10.1080/014311697217297>
- Segal, M.R. (2004) Machine Learning Benchmarks and Random Forest Regression. **Biostatistics** [online], (MAY 2003): 1–14. Available from:  
<http://escholarship.org/uc/item/35x3v9t4.pdf>
- Sharma, S., Osei-Bryson, K.-M. and Kasper, G.M. (2012) Evaluation of an integrated Knowledge Discovery and Data Mining process model. **Expert Systems With Applications** [online], 39 (13): 11335–11348. Available from:  
<http://dx.doi.org/10.1016/j.eswa.2012.02.044>
- Shimada, M. and Ohtaki, T. (2010) Generating Large-Scale High-Quality SAR Mosaic Datasets: Application to PALSAR Data for Global Monitoring. **Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of** [online], 3 (4):



637–656. Available from:

[http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=5604331](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5604331) [Accessed 28 October 2011]

Simental, E., Guthrie, V. and Blundell, S.B. (2005) "POLARIMETRY BAND RATIOS, DECOMPOSITIONS, AND STATISTICS FOR TERRAIN CHARACTERIZATION." In **Pecora 16 "Global Priorities in Land Remote Sensing."** Sioux Falls, South Dakota. 2005. pp. 1–11

Skriver, H., Dierking, W., Gudmandsen, P., et al. (2003) "APPLICATIONS OF SYNTHETIC APERTURE RADAR POLARIMETRY." In **Proceedings Workshop Applications SAR Polarimetry Polarimetric Interferometry (POLinSAR).** Frascati, Italy. 2003. pp. 11–16

Solberg, A.H.S., Brekke, C. and Husøy, P.O. (2007) Oil Spill Detection in Radarsat and Envisat SAR Images. **IEEE Transactions on Geoscience and Remote Sensing**, 45 (3): 746–755

SSTL (2011) **Government investment brings low cost radar satellites to market** [online].

Available from: <http://www.sstl.co.uk/News-and-Events?story=1936>

Stolz, R. and Mauser, W. (1995) First evaluations of Shuttle SIR-C and X-SAR data. **IEEE Geoscience and Remote Sensing Symposium (IGARSS)**, 2: 1058–1060

Strobl, C., Boulesteix, A.-L., Kneib, T., et al. (2008) Conditional variable importance for random forests. **BMC bioinformatics** [online], 9 (1): 307. Available from: <http://www.biomedcentral.com/1471-2105/9/307> [Accessed 9 July 2014]

Strobl, C. and Zeileis, A. (2008) Danger: high power! - Exploring the statistical properties of a test for random forest variable importance. **University of Munich, Department of Statistics Technical Report** [online], 017 (017): 1–8. Available from: <http://epub.ub.uni-muenchen.de/2111/1/techreport.pdf>

Strozzi, T., Dammert, P.B.G., Wegmüller, U., et al. (2000) Landuse Mapping with ERS SAR Interferometry. **IEEE Transactions on Geoscience and Remote Sensing**, 38 (2): 766–775

Svoray, T. and Shoshany, M. (2002) SAR-based estimation of areal aboveground biomass ( AAB ) of herbaceous vegetation in the semi-arid zone : A modification of the water-

- cloud model. **International Journal of Remote Sensing**, 23 (19): 4089–4100
- Tanase, M.A., Pérez-cabello, F., Riva, J. De, et al. (2010a) TerraSAR-X Data for Burn Severity Evaluation in Mediterranean Forests on Sloped Terrain. **IEEE Transactions on Geoscience and Remote Sensing**, 48 (2): 917–929
- Tanase, M.A., Santoro, M., Riva, J. De, et al. (2010b) Sensitivity of X- , C- , and L-Band SAR Backscatter to Burn Severity in Mediterranean Pine Forests. **IEEE Transactions on GeoScience and Remote Sensing**, 48 (10): 3663–3675
- Tehrany, M.S., Pradhan, B. and Jebu, M.N. (2013) A comparative assessment between object and pixel-based classification approaches for land use/land cover mapping using SPOT 5 imagery. **Geocarto International** [online], 6049 (June 2015): 1–19. Available from: <http://dx.doi.org/10.1080/10106049.2013.768300>
- The International Charter (2000) **Space and Major Disasters** [online]. Available from: <https://www.disasterscharter.org/web/guest/home> [Accessed 9 December 2015]
- Thenkabail, P.S., Biradar, C.M., Noojipady, P., et al. (2009) Global irrigated area map (GIAM), derived from remote sensing, for the end of the last millennium. **International Journal of Remote Sensing**, 30 (14): 3679–3733
- Le Toan, T., Quegan, S., Davidson, M.W.J., et al. (2011) The BIOMASS mission: Mapping global forest biomass to better understand the terrestrial carbon cycle. **Remote Sensing of Environment** [online], 115 (11): 2850–2860. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425711001362> [Accessed 28 July 2011]
- Torres, R., Paul, S., Malcolm, D., et al. (2013) THE SENTINEL-1 MISSION AND ITS APPLICATION CAPABILITIES. **IEEE International Geoscience and Remote Sensing Symposium**, pp. 1703–1706
- Torres, R., Snoeij, P., Geudtner, D., et al. (2012) GMES Sentinel-1 mission. **Remote Sensing of Environment** [online], 120: 9–24. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425712000600> [Accessed 3 April 2012]
- Tsui, O.W., Coops, N.C., Wulder, M. a., et al. (2012) Using multi-frequency radar and

- discrete-return LiDAR measurements to estimate above-ground biomass and biomass components in a coastal temperate forest. **ISPRS Journal of Photogrammetry and Remote Sensing** [online], 69: 121–133. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0924271612000585> [Accessed 23 April 2012]
- Turkar, V., Deo, R., Rao, Y.S., et al. (2012) Classification Accuracy of Multi-Frequency and Multi-Polarization SAR Images for Various Land Covers. **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing** [online], 5 (3): 936–941. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6209449>
- Ulaby and Dobson (1989) **Handbook of Radar Scattering Statistics for Terrain**. Norwood, MA: Artech House
- UNFCCC (2011) **Report of the Conference of the Parties on its sixteenth session, held in Cancun from 29 November to 10 December 2010. Addendum. Part Two: Action taken by the Conference of the Parties at its sixteenth session. : UNFCCC.**
- Vapnik, V. (1979) **Estimation of Dependences Based on Empirical Data**. (Russian). Moscow: Nauka
- van der Velde, R., Su, Z., van Oevelen, P., et al. (2012) Soil moisture mapping over the central part of the Tibetan Plateau using a series of ASAR WS images. **Remote Sensing of Environment** [online], 120: 175–187. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0034425712000739> [Accessed 6 April 2012]
- Voigt, S. (2011) Rapid Damage Assessment and Situation Mapping : Learning from the 2010 Haiti Earthquake. **Photogrammetric Engineering and Remote Sensing**, pp. 1–8
- Voigt, S., Kemper, T., Riedlinger, T., et al. (2007) Satellite Image Analysis for Disaster and Crisis-Management Support. **IEEE Transactions on GeoScience and Remote Sensing**, 45 (6): 1520–1528
- Vrieling, A. (2006) Satellite remote sensing for water erosion assessment: A review. **Catena**, 65 (1): 2–18

- Waldram, M.S. (2014) **Characterising disturbance in tropical peat swamp forest using satellite imaging radar** [online]. Available from: <http://hdl.handle.net/2381/28631>
- Waske, B. (2007) **Classifying multisensor remote sensing data: Concepts, Algorithms and Applications**
- Waske, B. and Benediktsson, J.A. (2007) Fusion of Support Vector Machines for Classification of Multisensor Data. **IEEE Transactions on GeoScience and Remote Sensing**, 45 (12): 3858–3866
- Waske, B. and Braun, M. (2009) Classifier ensembles for land cover mapping using multitemporal SAR imagery. **ISPRS Journal of Photogrammetry and Remote Sensing** [online], 64 (5): 450–457. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S0924271609000070> [Accessed 1 October 2013]
- Wegmuller, U., Werner, C., Strozzi, T., et al. (2004) **FLASHING FIELDS ! A PRELIMINARY INVESTIGATION**
- Werner, C., Strozzi, T., Wiesmann, A., et al. (2012) **GAMMA SAR and Interferometry software**. Gumlingen, Switzerland
- Werner, C., Wegmüller, U., Strozzi, T., et al. (2000) “GAMMA SAR AND INTERFEROMETRIC PROCESSING SOFTWARE.” In **ERS-ENVISAT Symposium. Gothenburg, Sweden. 2000.** pp. 1–9
- Werninghaus, R. and Buckreuss, S. (2010) The TerraSAR-X Mission and System Design. **IEEE Transactions on Geoscience and Remote Sensing** [online], 48 (2): 606–614. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5339240>
- White, F. (1983a) **The vegetation of Africa: a descriptive memoir to accompany the UNESCO/AETFAT/UNSO vegetation map of Africa**
- White, F. (1983b) **Vegetation map of Africa**. [online]. Available from: [http://omap.africanmarineatlas.org/BIOSPHERE/pages/3\\_terrestrial vegetation.htm](http://omap.africanmarineatlas.org/BIOSPHERE/pages/3_terrestrial%20vegetation.htm)
- White, L., Brisco, B., Dabboor, M., et al. (2015) **A Collection of SAR Methodologies for Monitoring Wetlands** [online]. Available from: <http://www.mdpi.com/2072-4292/7/6/7615/>

- Woodhouse, I.H. (2006) **Introduction to Microwave Remote Sensing**. 1st Ed. Boca Raton, FL: Taylor & Francis Group
- WWF (2016a) **Central Africa: Eastern Democratic Republic of the Congo** [online]. Available from: <http://www.worldwildlife.org/ecoregions/at0101> [Accessed 26 July 2016]
- WWF (2016b) **Sahelian Acacia savanna** [online]. Available from: <http://www.worldwildlife.org/ecoregions/at0713> [Accessed 26 July 2016]
- WWF (2016c) **Western Africa: Western Cameroon extending into Nigeria** [online]. Available from: <http://www.worldwildlife.org/ecoregions/at0103> [Accessed 26 July 2016]
- Wylie, D., Jackson, D.L., Menzel, W.P., et al. (2005) Trends in Global Cloud Cover in Two Decades of HIRS Observations. **Journal of Climate**, 18: 3021–3031
- Zink, M. (2012) TanDEM-X mission status. **IEEE Geoscience and Remote Sensing Symposium (IGARSS)** [online], pp. 1896–1899. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6351136>
- Zribi, M., Kotti, F., Baghdadi, N., et al. (2012) Soil Texture Estimation Over a Semiarid Area Using TerraSAR-X Radar Data. **IEEE Geoscience and Remote Sensing Letters**, 9 (3): 353–357