Stefan Gerd Uhlmann

**Advanced Techniques for Classification of
Polarimetric Synthetic Aperture Radar Data**

Tampere 2014

Stefan Gerd Uhlmann

Advanced Techniques for Classification of Polarimetric Synthetic Aperture Radar Data

Thesis for the degree of Doctor of Science in Technology to be presented with due permission for public examination and criticism in Sähkötalo Building, Auditorium S3, at Tampere University of Technology, on the 1st of September 2014, at 12 noon.

Supervisors:

Prof. Dr. Serkan Kiranyaz (Custos),
Department of Signal Processing,
Faculty of Computing and Electrical Engineering,
Tampere University of Technology,
Tampere, FINLAND.

Academy Prof. Moncef Gabbouj,
Department of Signal Processing,
Faculty of Computing and Electrical Engineering,
Tampere University of Technology,
Tampere, FINLAND.

Pre-examiner:

Prof. Dr. Björn Waske,
Institute of Geographical Sciences,
Remote Sensing and Geoinformatics,
Freie Universität Berlin,
Berlin, GERMANY.

Pre-examiner and opponent:

Prof. Dr. Vladimir Lukin,
Department of Signal Reception, Transmission and Processing 504,
National Aerospace University ("Kharkov Aviation Institute"),
Kharkov, UKRAINE.

Opponent:

Prof. Dr. José Manuel Bioucas Dias,
Instituto de Telecomunicações,
Instituto Superior Técnico,
Lisbon, PORTUGAL.

Abstract

With various remote sensing technologies to aid Earth Observation, radar-based imaging is one of them gaining major interests due to advances in its imaging techniques in form of synthetic aperture radar (SAR) and polarimetry. The majority of radar applications focus on monitoring, detecting, and classifying local or global areas of interests to support humans within their efforts of decision-making, analysis, and interpretation of Earth's environment.

This thesis focuses on improving the classification performance and process particularly concerning the application of land use and land cover over polarimetric SAR (PolSAR) data. To achieve this, three contributions are studied related to superior feature description and advanced machine-learning techniques including classifiers, principles, and data exploitation.

First, this thesis investigates the application of *color* features within PolSAR image classification to provide additional discrimination on top of the conventional scattering information and texture features. The color features are extracted over the visual presentation of fully and partially polarimetric SAR data by generation of pseudo color images. Within the experiments, the obtained results demonstrated that with the addition of the considered color features, the achieved classification performances outperformed results with common PolSAR features alone as well as achieved higher classification accuracies compared to the traditional combination of PolSAR and texture features.

Second, to address the large-scale learning challenge in PolSAR image classification with the utmost efficiency, this thesis introduces the application of an adaptive and data-driven supervised classification topology called Collective Network of Binary Classifiers, CNBC. This topology incorporates active learning to support human users with the analysis and interpretation of PolSAR data focusing on collections of images, where changes or updates to the existing classifier might be required frequently due to surface, terrain, and object changes as well as certain variations in capturing time and position. Evaluations demonstrated the capabilities of CNBC over an extensive set of experimental results regarding the adaptation and data-driven classification of single as well as collections of PolSAR images. The experimental results verified that the evolutionary classification topology, CNBC, did provide an efficient solution for the problems of scalability and dynamic adaptability allowing both feature space dimensions and the number of terrain classes in PolSAR image collections to vary dynamically.

Third, most PolSAR classification problems are undertaken by supervised machine learning, which require manually labeled ground truth data available. To reduce the manual labeling efforts, supervised and unsupervised learning approaches are combined into semi-

supervised learning to utilize the huge amount of unlabeled data. The application of semi-supervised learning in this thesis is motivated by ill-posed classification tasks related to the small training size problem. Therefore, this thesis investigates how much ground truth is actually necessary for certain classification problems to achieve satisfactory results in a supervised and semi-supervised learning scenario. To address this, two semi-supervised approaches are proposed by unsupervised extension of the training data and ensemble-based self-training. The evaluations showed that significant speed-ups and improvements in classification performance are achieved. In particular, for a remote sensing application such as PolSAR image classification, it is advantageous to exploit the location-based information from the labeled training data.

Each of the developed techniques provides its stand-alone contribution from different viewpoints to improve land use and land cover classification. The introduction of a new feature for better discrimination is independent of the underlying classification algorithms used. The application of the CNBC topology is applicable to various classification problems no matter how the underlying data have been acquired, for example in case of remote sensing data. Moreover, the semi-supervised learning approach tackles the challenge of utilizing the unlabeled data. By combining these techniques for superior feature description and advanced machine-learning techniques exploiting classifier topologies and data, further contributions to polarimetric SAR image classification are made. According to the performance evaluations conducted including visual and numerical assessments, the proposed and investigated techniques showed valuable improvements and are able to aid the analysis and interpretation of PolSAR image data. Due to the generic nature of the developed techniques, their applications to other remote sensing data will require only minor adjustments.

Preface

The research presented in this thesis has been carried out at the Department of Signal Processing as part of the Faculty of Computing and Electrical Engineering of the Tampere University of Technology, Tampere, Finland within the MUVIS team of the Multimedia Research Group during 2010-2014.

First and the foremost I wish to express my deepest gratitude to my supervisors, Prof. Serkan Kiranyaz and Prof. Moncef Gabbouj, for their guidance, support, patience, and belief during all these years, whenever needed. I am in debt to Prof. Moncef Gabbouj for giving me the opportunity to work with him starting in 2006 as a Research Assistant, and to Prof. Serkan Kiranyaz for his technical guidance and support besides the physical distances between us. Furthermore, thanks go to Prof. Turker Ince introducing the fields of remote sensing and SAR to use and me.

I would like to thank Prof. Vladimir Lukin and Prof. Björn Waske for serving as pre-examiners of this thesis and for their constructive feedback and helpful comments. I also would like to thank Prof. Vladimir Lukin and Prof. Josè Bioucas for agreeing to serve as opponents during the public defense.

Thanks also go to past and present MUVIS members and SGN colleagues for the pleasant and multicultural working environment. I especially extend my thanks to my office mates from TC413 (Guanqun Cao, Honlgei Zhang, Kaveh Samiee, Jenni Pulkkinen) for putting up with me the last few years and providing this friendly office atmosphere. Special thanks go to our lunch/coffee group particularly Uygar Tuna and Murat Birinci for our memorable conversations and discussions.

I would like to thank Prof. Reiner Creutzburg from my home university (FH Brandenburg), without him I would not have found my way to Tampere and TUT. Moreover, I want to thank Dr. Alan Payne and Dr. Peter Fry for giving me the opportunity to work alongside them during my stay with Kodak European Research in Cambridge, UK and later in their spin-off DeepVisual Ltd. It had been a great pleasure and experience, I met so many wonderful people, and I would say that the time spent there, change my life.

Special thanks go to Noora Rotola-Pukkila, Susanna Anttila, Johanna Pernu, Vivre Larmila, Päivi Salo, Elina Orava, and Ulla Siltaloppi for their kind help with practical and administrative matters.

Work is not everything, so warmest thanks go to the many friends and people, too many to mention, I had the fortune to meet during this time. I shall cherish the moments we spent together in Tampere, Finland, and abroad.

Last but not least, I have no English words to express my thanks to my family. Ich danke von ganzem Herzen meiner Familie, insbesondere meinem Vater, meiner Mutter, und meinem Bruder für euren andauernden Beistand, Rückhalt und Liebe über all die Jahre trotz der Entfernungen. Eure Unterstützung und Motivation in manch schwieriger Entscheidung haben mir stets geholfen und mich wieder auf den richtigen Weg geführt. Ohne euch wäre ich heute nicht hier und der Mensch, der ich bin.

Ich widme diese Arbeit meiner Familie und in Andenken an meine Oma Marie und meinen Opa Kurt.

Tampere, July 2014   Stefan Uhlmann.

Contents

List of Acronyms

$\langle[C]\rangle$	Covariance Matrix
$\langle[T]\rangle$	Coherency Matrix
$[S]$	Scattering Matrix
AIRSAR	NASA/Jet Propulsion Laboratory (JPL) Airborne SAR
AL	Active Learning
ANN	Artificial Neural Network
BP	Backpropagation
BC	Binary Classifier
CLD	Color Layout Descriptor
CNBC	Collective Network of Binary Classifier
COSMO-SKYMED	Italian Space Agency COnstellation of small Satellites for the Mediterranean basin Observation (Space-borne SAR)
CSD	Color Structure Descriptor
CTD	Coherent Target Decomposition
CV	Class Vector
DCD	Dominant Color Descriptor
DCT	Discrete Cosine Transform
DT	Decision Tree
EANN	Evolutionary Artificial Neural Network
EHD	Edge Histogram Descriptor
EO	Earth Observation
FS	Feature Set
FV	Feature Vector
GLCM	Gray Level Co-occurrence Matrix
GW	Gabor Wavelets
H	Horizontal linear (basis polarization)
HSV	Hue, Saturation, Value (=Luminance, Intensity)
ICTD	Incoherent Target Decomposition
LBP	Local Binary Pattern

LULC	Land Use Land Cover
MD-PSO	Multi-Dimensional PSO
MLP	Multi-Layer Perceptron
MLPH	Multiple Local Pattern Histogram
MPEG	Moving Picture Experts Group
NBC	Network of Binary Classifier
OCM	Ordinal Co-occurrence Matrix
OvA	One-versus-All
OvO	One-versus-One
PNR	Positive-to-Negative Ratio
PolSAR	Polarimetric Synthetic Aperture Radar
PSO	Particle Swarm Optimization
Radar	Radio Detection And Ranging
RADARSAT-2	Canadian Space Agency Space-borne SAR
RBF	Radial Basis Function
RF	Random Forest
RGB	Red-Green-Blue Color Space
RS	Remote Sensing
SAR	Synthetic Aperture Radar
SL	Supervised Learning
SNH	Search Neighborhood
SSL	Semi-Supervised Learning
SVM	Support Vector Machine
TERRASAR-X	German Aerospace Center Space-borne SAR
TD	Target Decomposition
TS	Training Set
UL	Unsupervised Learning
V	Vertical linear (basis polarization)

List of Tables

List of Figures

Chapter 1

Introduction

I n the past, capturing pictures of our planet Earth from space has fascinated people, and still does. These days images of the earth are constantly captured and acquired, and scientific communities use them to understand better and improve management of the earth itself and its environment. Such images capturing the earth enable us to see the world through a wide frame to witness large-scale phenomena with an accuracy and entirety, which could barely be captured by human efforts on the ground. This is highly advantageous when it comes to data acquisition around the world particularly places too remote or otherwise inaccessible. Recent examples are monitoring man-made and natural disasters, where a single satellite image has the potential to show the Deepwater Horizon oil spill spread in April 2010, the precise damage assessment after the 2010 Haiti earthquake or the tsunami that hit Japan in March 2011, the entire span of a hurricane or typhoon such as Typhoon Haiyan in November 2013, or help in search of a missing airplane in March 2014.

The gathering of information about planet Earth's physical, chemical, and biological systems falls within the discipline of Earth observation (EO). Hence, one of the core objectives of EO is to monitor and assess the status and changes of the natural and man-made environments via remote sensing technologies. Remote sensing (RS) in a broader sense involves collecting data and information about the physical world surrounding us. This is achieved by detecting and measuring signals, which can be composed of radiation or particles originating from any kind of object that is not in the close vicinity of the sensor. Such sensor can be carried by various platforms for instance satellite, aircraft, spacecraft, buoy, ship, and helicopter. The collected data are helpful for city planning, archaeological investigations, military observation, and geomorphological surveying by analyzing and comparing observations such as vegetation rates, soil erosion, air pollution, forestry, weather, and land use.

To initiate the remote sensing process, the first requirement is an energy source that illuminates or provides electromagnetic energy to the target of interest. As the energy travels from its source to the target and then to a sensor, it will interact with the atmosphere it passes through. Once the energy reaches the target, it interacts with it depending on the properties of the target and atmosphere. After the energy has been scattered by or emitted from the target, a sensor is used to collect and record the electromagnetic radiation. The data recorded by the

sensor is generally transmitted to a ground station where it is processed into images. These processed images can then be used to extract information about the target for analysis and interpretation. Usually the obtained data are available in digital form, so that the raw data can be processed in many ways, as the end user requires. The final stage of the remote sensing process is the application of the extracted information for better understanding, obtained new information, or solving a particular problem. The entire remote sensing process is illustrated in Figure 1.

Remote sensing for EO can be divided into two main sensor categories, which is related to the energy source used to illuminate the target. *Passive* sensors utilize an external energy source such as the sun to emit the initial signal whereas *active* sensors are capable of transmitting the electromagnetic energy signal itself besides being able to receive scattered electromagnetic radiation from targets. Common passive RS sensors are multispectral (e.g., Landsat [9], Quick Bird [55], WorldView-1 and 2 [55], GeoEye-1 [55], Pleiades [14]) and hyperspectral (e.g., AVIRIS [75], Hyperion [157], HYDICE [140], CASI [12], DAIS [145]) systems depending on the number of different electromagnetic wavelengths captured. Widely used active RS sensors are AIRSAR [129], EMISAR [37], E-/F-SAR [89], [90], Pi-SAR [206], and RAMSES [59] as airborne systems; ENVISAT-ASAR [93], ALOS-PALSAR [167], RADARSAT-2 [141], TerraSAR-X [112], and COSMO-SkyMed [183] as space-borne systems. With the advances in such remote sensing systems, particularly the amount of data available in shorter periods of time with much higher image resolutions, gathering critical information proved highly beneficial for interpreting and analyzing while observing the earth [142].
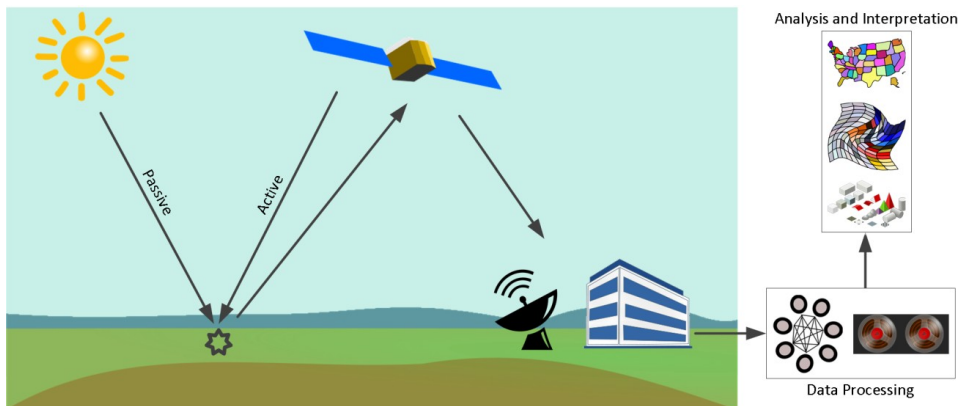
Figure 1 – Illustration of the remote sensing process – transmitting, receiving, processing, and analyzing.

1.1.   SYNTHETIC APERTURE RADAR

A commonly used active remote sensing system is *RAdio Detection And Ranging* widely known as *radar*. In its simplest form, it operates by broadcasting a pulse of electromagnetic energy and if that pulse encounters an object and interacts with it, then some of that energy

scatters back to the radar antenna. Generally, radar works within the microwave region of the electromagnetic spectrum from P-Band (0.3 GHz) up to K_a-Band (40 GHz) [142], as illustrated in Figure 2. Radar systems have the advantage over their optical or spectral counterparts regarding backscatter sensitivity to terrain and object characteristics (e.g., pass through tree canopy, dry surface deposits, snow) and the ability to operate in all-weather conditions (i.e., pass through clouds, fog, rain, smokes) ([152], Chapter 4.1.1). Particularly, lower frequencies are the least affected by clouds and rain compared to higher frequencies at X-Band [205]. Furthermore, as an active remote sensing system, it is capable of operating day and night, as it does not rely on the sun as an external transmitting source.

Wavelength (cm)

0.3	0.8	2.5	3.8	7.5	15	60	120

W	V	Q	K_a	K	K_u	X	C	S	L	P

110	40	12.5	8	4	2	0.5	0.25

Frequency (GHz)
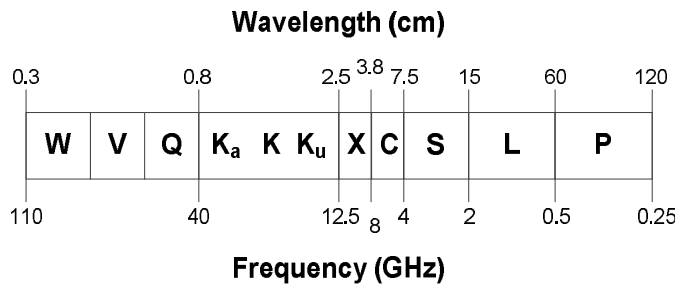
Figure 2 – Electromagnetic spectrum of microwaves used within radar.

The spatial resolution of a (real aperture) radar system is determined by, among other things, the size of its antenna. To focus the transmitted and received energy into a sharp beam (pulse of energy), a physically large antenna is needed which is similar for optical systems lenses or mirror sizes as they also require large apertures to obtain high image resolution. However, to accomplish high image resolutions using radar, such remote sensing system would require an antenna physically larger than the one that can be practically carried by a sensor platform, i.e., antenna lengths of several hundred meters would be needed. To overcome this physical limitation, radar systems utilize the flight path of the platform to simulate an extremely large antenna or aperture electronically. The distance the system covers during its flight in synthesizing the antenna is known as the *synthetic aperture*. A coherent radar imaging system that applies the technique of synthetic aperture to generate high-resolution remote sensing imagery is referred to as *synthetic aperture radar* (SAR).

Generally, SAR works based on the same principles as a phased array with many parallel antenna elements. SAR simulates this phased array principle in a time-multiplex approach, where the different positions of the antenna elements are imitated by the moving sensor platform. While moving along its flight path, the SAR processor continuously stores all its received radar signals (amplitudes and phases) over a time T for positions P_t. With the information from P_t, it is possible to reconstruct the signal, which would have been generated by an antenna of length $v \cdot T$, with v being the platform speed. As the line of sight direction changes along the platform flight path, synthetic aperture is produced by signal processing which results in simulating a lengthy antenna. The size of the synthetic aperture is associated to time T used to collect the returned radar signals, so that with a larger T higher resolution

can be realized. The achievable resolution of a SAR sensor is related to the length of the real antenna, yet is independent of the platform altitude. Therefore, SAR is intrinsically the only viable and practical radar imaging technique that can achieve high spatial resolution, also from space platforms ([152], Chapter 1.2.6).
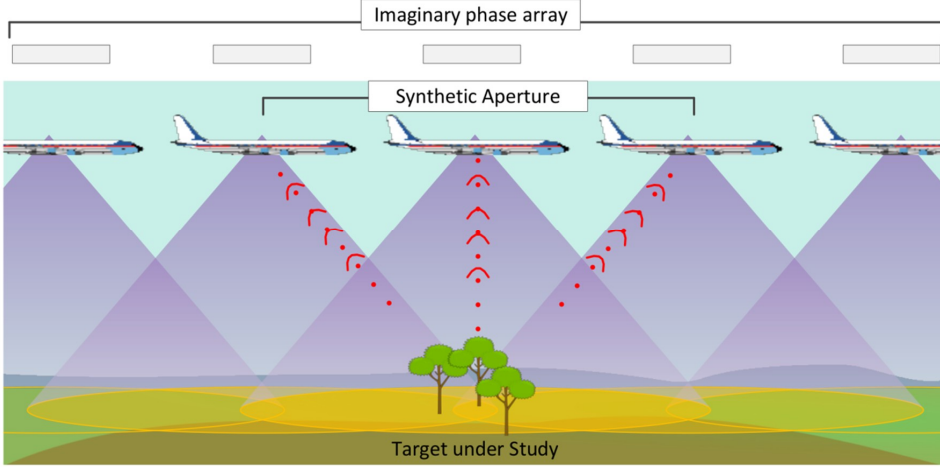
Figure 3 – Illustration of generating synthetic aperture with a moving platform.

The idea of SAR was initially mentioned by Carl A. Wiley [214] within a Goodyear report June 1951 and put into operation by early 1952. For the next 25 years, experiments were primarily conducted using airborne SAR systems until 1978 when the SEASAT satellite carried the first space-borne SAR [98] into orbit for Earth observation. At this time, main emphasize was given to the obtained amplitude or intensity images while neglecting the available phase information by SAR processors. While the amplitude data offering the fundamental information for SAR image analysis, SEASAT-SAR provided the addition of *phase* and *polarization* information in the coherently processed complex images. Based on this additional phase information, the technology of interferometric SAR (InSAR) [74] was established and its main applications today are generation of digital elevation models as well as observations of crust movement and deformation caused by earthquakes, volcanic activities, and glacier flows.

The available information about polarization within complex SAR images led to another technology known as *polarimetric SAR* (PolSAR) [208], [224]. The concept of *radar polarimetry* (Polar: polarization, Metry: measure) is to acquire, process, and analyze polarization of transmitted and/or received electromagnetic waves. Generally, polarization refers to the orientation of the plane of the electric field, as opposed to the magnetic field of the earth's surface. Now, radar polarization is measured as the orientation of the electric vector in the electromagnetic waves, which can be achieved in different modes independent from the signal wavelength.
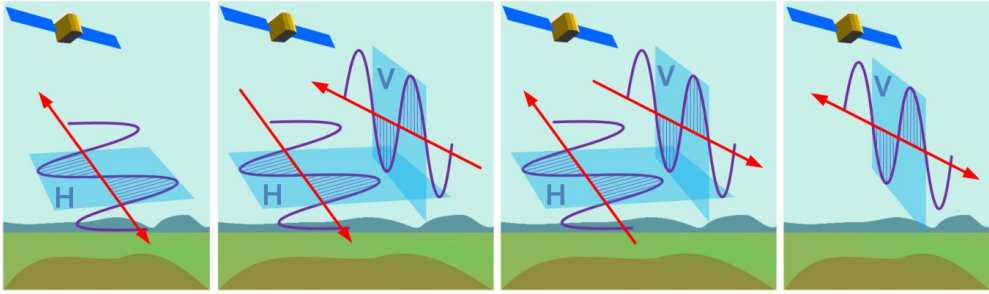
Figure 4 – Combinations for polarization of electromagnetic waves in H and V basis.

In conventional imaging radar systems, waves are polarized either in *horizontal* (H) or *vertical* (V) basis polarization with the lone electric vector oriented in the horizontal or vertical direction in antenna coordinates. Polarization is generated by the antenna, which may be adjusted to transmit and receive in different polarization modes. For example, the horizontal transmit and vertical receive polarization (HV) is where the electromagnetic waves of the electric field are oriented in the horizontal plane for signal transmission, and where the vertically polarized electric field of the backscattered energy is received by the radar antenna. This results in four possible combinations for transmitting and receiving the wave's polarization as

1. HH - for horizontal transmit and horizontal receive,
2. VV - for vertical transmit and vertical receive,
3. HV - for horizontal transmit and vertical receive, and
4. VH - for vertical transmit and horizontal receive,

as illustrated in Figure 4. HH and VV are referred to as co-polarized since transmitting and receiving polarizations are the same, whereas HV and VH are called cross-polarized as their transmitting and receiving polarizations are orthogonal to one another.

Now, depending on its antenna configuration, a system may support different polarization modes when obtaining the PolSAR data. Single polarized (single-pol) data are provided in the form of HH, VV, HV, or VH polarization mode with HH or VV being the most typical single-pol configurations. In case of dual polarized (dual-pol) configuration, data are commonly provided in one of the three combinations HH/HV, VV/VH, or HH/VV. When a system supports all four polarization (quad-pol) combinations, it is also called fully polarimetric. The majority of the aforementioned airborne (AIRSAR, EMISAR, E-/F-SAR, Pi-SAR, and RAMSES) and space-borne (ENVISAT-ASAR, ALOS-PALSAR, RADARSAT-2, TerraSAR-X, COSMO-SkyMed) SAR systems support all four polarization modes. This is because polarimetric data contain more information on the scattering from objects than the conventional single- and dual-polarization data. A review of polarimetry in the context of SAR can be found in [184].

1.2.   A BRIEF HISTORY ON SYNTHETIC APERTURE RADAR

After the conception of SAR in the earlier 1950s, the establishment of imaging radar as an essential remote sensing tool for Earth observation happened with the launch of the SEASAT-satellite. SEASAT-SAR was the first space-borne platform particularly targeting remote sensing of oceans and sea ice able to cover wide surface area operating with a single HH polarization. Additionally, applications to general terrain discrimination and target detection could also be considered. SEASAT-SAR operations ended after 105 days due to a massive electric system failure, however, it is regarded as the pioneering mission demonstrating the capabilities of SAR and imaging radar technology. It led to many follow-on space-borne SAR systems such as NASA's SIR-A (HH) in 1981 and SIR-B (HH) in 1984 on space shuttles, the European ERS-1, 2 (VV) in 1992 and 1995, the Japanese JERS-1 (HH) [148] in 1992, and the Canadian RADARSAT-1 (HH) [163] in 1995. Furthermore, this led to advances of dual-polarization and fully polarimetric imaging radar as extensions of such single polarization SAR.

Advances of polarimetric radar imaging picked up in 1985 with the first practical fully polarimetric airborne SAR system (AIRSAR) developed by the NASA Jet Propulsion Laboratory (JPL). For over 20 years, NASA/JPL flew AIRSAR able to operate in L-, C-, and P-Bands simultaneously making it the primary source for polarimetric radar imaging data. The availability of such data from campaigns all over the world stimulated intensive research in polarimetric radar imaging, its related polarimetric analysis techniques, and associated applications.

In the early 1990s, PolSAR research in Europe started to rise with the support of the European Space Agency (ESA) resulting in various airborne PolSAR systems. In particular, the German Aerospace Research Center built and operated Experimental SAR (E-SAR) with quad polarization in L- and P-Band providing higher spatial resolution than AIRSAR. E-SAR was followed by F-SAR in November 2006 providing a full modular system with X-, C-, S-, L- and P-Band with full polarimetric capabilities in all frequencies. Other systems were EMISAR from Denmark with quad-pol capabilities C- and L-Band with resolutions of up to three meters; RAMSES from France provided various bands (Ka-, X-, C-, S-, L-, P-); the CONVAIR 580 SAR from the Canadian Center for Remote Sensing with X-, C-, P-Band experimental SAR systems as well as the Pi-SAR from Japan with X-Band and L-Band PolSAR sensors.

Space-borne PolSAR systems effectively started with the successful launch of the SIR-C/X-SAR onboard Space Shuttles in April and October 1994. Even though the two missions were relatively short in time (10 days), SIR-C acquired digital SAR images of the earth with fully PolSAR at C-Band and L-Band, and a single polarization X-Band SAR simultaneously. The first space-borne fully polarimetric imaging platform was the Advanced Land Observing Satellite (ALOS-PALSAR, Japan) in January 2006 with L-Band PolSAR sensor onboard combined with two optical instruments. Since then various satellites have been launched providing full polarimetric imaging capabilities. In 2007, TerraSAR-X (Germany) and RADARSAT-2 (Canada) were launched in June and December, respectively. TerraSAR-X was

later extended with TerraSAR-X add-on for Digital Elevation Measurement (TanDEM-X) making it the only one dedicated to InSAR digital elevation model (DEM) generation. This is used as the basis to generate WorldDEM, the global digital elevation models. Within 2007-2010, the COnstellation of small Satellites for the Mediterranean basin Observation (COSMO-SkyMed, Italy) launched four identical medium-sized satellites equipped with X-Band SAR capabilities providing the opportunity to observe an area of interest several times a day. The majority of the later space-borne systems are "sole-SAR" systems with only SAR capabilities reducing their weight significantly compared to platforms combining multiple sensor types and providing higher spatial resolutions with different imaging modes than the previous systems.

Additional missions are being planned to deploy further SAR platforms in the future such as Sentinel-1A [10] as the first satellite of the Copernicus programme, ALOS-2 [174], and AstroSAR-Lite [88], which is particularly focused on extremely short revisit times (10-15 per day) and coverage with high image resolution for the regional use in the tropics and sub-tropics.

1.3. OBJECTIVES

Due to the development and technical advances of radar technologies over time, airborne and space-borne systems provide sufficient data for remote sensing the earth's environment. Applications are manifold due to the all-weather and day/night capabilities of radar such as in the fields of *geology* (e.g., topography, DEM production), *agriculture* (e.g., crop classification, soil moisture), *forestry* (e.g., tree biomass, height, species, plantation and deforestation, forest fire monitoring), *urban*, (e.g., urban structure and density, change detection), *disaster* (e.g., monitoring of damage and recovery), and *oceanography*, (e.g., waves-, wind-, ship-detection, oil spill, -slick monitoring) to name a few. An extended list of applications and detailed discussions can be found in [153].

A wide majority of applications is in the areas of monitoring, detecting, and classifying local or global regions and targets. These analyzing and interpretation tasks are generally undertaken with the help of automatic tools to support human experts. Such tools and techniques have been introduced from various disciplines such as signal/image processing and computational- and machine intelligence particularly machine learning. With the growing amount of available PolSAR data due to higher resolutions and shortened revisiting cycles, this obviously becomes a large-scale learning challenge where the applied tools and techniques should be able to process such huge amount of data efficiently and effectively. By enabling this, the gain to support human experts in analyzing and interpreting is critical.

This starts with efficient data processing techniques for information content extraction and description exploiting raw SAR data by polarimetric target decompositions (as covered in Section 2.2) and image processing techniques introducing various texture descriptions for analysis (as described in Section 2.3). Any technique able to provide better description helping in discrimination to ease the analysis and interpretation in automatic ways is useful partic-

ularly when combined with machine learning methods for classification, detection, and monitoring.

Such machine learning methods are especially valuable if they are capable of processing and learning from large amount of data. Additionally, it is highly beneficial if they are able to adapt to the task, which might exhibit significant variations in time due to the introduction of new data provided by the human expert. Furthermore, it is advantageous if tools are able to utilize *all* available data in supervised and/or unsupervised manner especially if the ground truth cannot be checked or verified by on-site visits, as it would be too cumbersome and expensive for large repositories containing imagery from inaccessible regions.

In this thesis, the focus is particularly drawn on the introduction and investigation of advanced techniques to improve PolSAR image classification performance especially on the application of land use and land cover (LULC). This thesis significantly contributes on PolSAR image classification regarding the following novel aspects:

- ❑ Introduction and investigation of new descriptions for information content extraction by employing image processing techniques for better discrimination,
- ❑ Application of an adaptive and data-driven classification topology for large-scale learning to adapt to new challenges and tasks in a simple way while keeping existing knowledge and providing support for the human expert,
- ❑ Utilization of all available data using semi-supervised learning combining approaches of supervised and unsupervised learning paradigms.

1.4. THESIS OVERVIEW

The remainder of this thesis is organized as follows. First, a general overview about SAR data management is presented in Chapter 2 considering the necessary steps up to the analytical and interpretation stage of the remote sensing process. The chapter starts with a brief description of the underlying data acquisition and SAR functionalities. Particular focus is then given to the information content extraction and processing from the raw SAR data over the scattering matrix utilizing the full polarimetric information. Moreover, the extraction of additional texture features over a SAR image in combination with traditional SAR features for an extended analysis procedure is described in this chapter. At the end, various SAR image data as well as a feature extraction description of the aforementioned SAR and texture features are presented. They are mainly utilized within the experiments of the following chapters in this thesis.

Chapter 3 proposes some novel features to the application of land use and land cover terrain classification. In particular, the integration of color features extracted over pseudo color images is presented over fully and partially polarimetric SAR data in order to improve terrain discrimination and classification accuracy. This covers the generation of various pseudo color images as well as describing the underlying color features. The work presented in this chapter is based on the author's publications, [195], [196], [198], and [200].

Besides the novel feature integration into SAR image classification, a data-driven and adaptive classification approach is presented in Chapter 4, where the application of Collective Network of Binary Classifier (CNBC) topology is considered over PolSAR data. The CNBC design allows dynamic adaptation to changes of the classification task at hand such as integration of new ground truth or feature information. Furthermore, this topology provides the capabilities to adapt to the data-driven classification over a large collection of available PolSAR images to reduce learning and training complexities. This chapter is mainly based on the author's original publications [108], [197], [198], [199], and [201].

Chapter 5 presents the application of the semi-supervised learning (SSL) paradigm within land use and land cover classification over PolSAR images. The focus is on addressing ill-posed classification where large numbers of classes are to be discriminated and the number of training samples per class is noticeably smaller in relation to the feature dimension. With the rather limited amount of training data available, this is also referred to the *small sample size problem,* as an underlying classifier will lack discrimination and generalization capabilities. To address this, two SSL approaches are proposed by extension of the training data and ensemble-based self-training. Furthermore, spatial techniques exploiting the pixel neighborhoods and superpixel segmentation are presented to select reliable and informative samples from the pool of unlabeled data improving speed and accuracy of the self-training process. The work presented in this chapter is based on the author's publications, [194] and [202].

Due to the variety of covered subjects, introduction and experimental results along with conclusive remarks are presented in each chapter. Conclusions of the thesis are finally drawn in Chapter 6.

1.5. AUTHOR'S CONTRIBUTIONS

The author's contributions to the field of land use and land cover classification over polarimetric SAR images are collected within the Chapters 3, 4, and 5. The main contributions can be summarized in the following points:

- Application of CNBC over polarimetric SAR data, in [108] and [198].
- Data-driven classification of a single polarimetric SAR image via incremental evolution, in [199].
- Adaptive classification of multiple polarimetric SAR images via incremental evolution, in [197] and [201].
- Integration of color features into classification of fully polarimetric SAR images, in [196], [198], and [200].
- Classification of partially polarimetric SAR images by integrating visual features, in [195].
- Semi-supervised learning of ill-posed polarimetric SAR image classification via ensemble-based self-training, in [194] and [202].

The initial idea of CNBC originates from Serkan Kiranyaz and Turker Ince with the latter introducing the field of PolSAR image classification [108]. The author is responsible for implementation of the graphical user interface application and parts of the feature extraction, as well as helping with conducting the experiments and contributing parts to the writing. In publications [197], [199], and [201], design and implementation of the proposed CNBC extension and running the experiments was performed by the author. The publications were mainly written by the author with the help of co-authors who also provided certain parts of the text during the review processes.

The general idea of applying color features to PolSAR images in [198] comes from Serkan Kiranyaz and the author, who are also responsible for major parts of the writing. The remaining co-authors provided parts of the text. Furthermore, The author conducted the experiments and produced the results. For publications [195], [196], and [200], the author is responsible for design, implementation, and performing the experiments. Moreover, the writing was undertaken mainly by the author while the co-authors provided parts of the text in the review processes.

The works in [194] and [202] were designed, implemented, and evaluated by the author. Writing the text was done by the author with the help of the co-authors.

Chapter 2

SAR Data Management

S AR data are obtained by means of a SAR sensor generally mounted to an airborne or space-borne platform. Such platform is also used to carry the energy source emitting electromagnetic microwaves to observe a target of interest. Once the energy makes its way to the target, it will interact with the target, the energy is scattered back, and the sensor on the SAR platform will collect electromagnetic radiation. Now this recorded energy may be transmitted to a receiving station on the earth for further data processing. However, in recent years, a lot more processing is already undertaken on board to form the final image representation. This image is then available to extract information about the target of interest, which can be used to solve a particular problem.

This chapter mainly covers the basic stages how a radar system works in generating radar imagery and briefly detail the parts commonly needed to further process and analyze. Section 2.1 outlines the steps of the radar data acquisition to obtain the radar images with their underlying content representation. Section 2.2 then describes the process of extracting and generating physical and geometrical information from PolSAR images we consider relevant for this thesis. Furthermore, well-known visual descriptors such as texture features have been employed for additional information during SAR data analysis and they are presented in Section 2.3. Previous interpretation approaches and analysis techniques are covered in Section 2.4. Within Sections 2.5 and 2.6, the PolSAR image data, polarimetric SAR- and texture features considered in the experiments and evaluations throughout the remaining chapters are introduced.

2.1. SAR IMAGE ACQUISITION AND INFORMATION CONTENT

Generally, an airborne or space-borne SAR system operates within the electromagnetic spectrum in a side-looking geometry, where a transmitted wave is perpendicular to the system's flight direction and images are obtained of a ground surface due to its backscattered wave signals ([116], Chapter 1.2). Thus, when a radar system transmits a pulse of energy to the ground, it scatters off the ground in all directions. A part of the scattered energy is directed back toward the radar receiver, and this part is referred to as *backscatter* as illustrated in Fig-

ure 5. Possible spatial and temporal fluctuation due to gains and losses from a SAR system are commonly corrected by calibration, which involves internal instrument calibration as well as external SAR calibration using targets of known reflectivity [67].
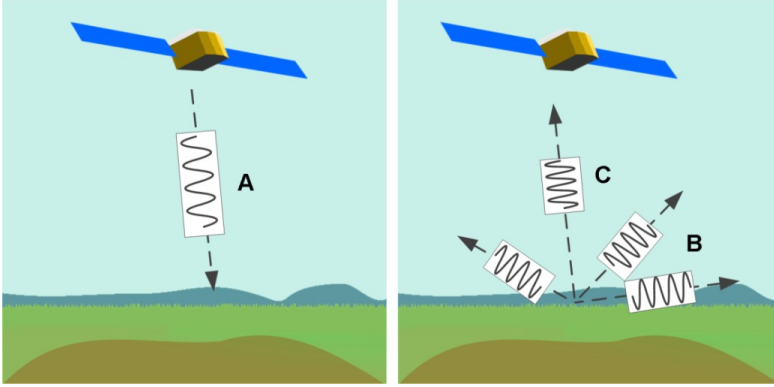
Figure 5 – Radar and backscatter (A) transmission, (B) scatter in various directions and parts directed and received called *backscatter* (C).

The received echo wave signal form a two-dimensional complex data matrix, where each complex sample represents amplitude and phase value with its real and imaginary part, respectively. Those raw data are formed into SAR images via signal processing techniques, which can simply be understood as two separate matched filter operations along the range and azimuth dimensions [142], ([165], Appendix D: Image Formation with Synthetic Aperture Radar). Now a synthesized radar image contains two basic information: (1) amplitude, which measures the magnitude of the radiation, or backscatter, received from the target; and (2) phase, which indicates how the dielectric or geometric properties of the target can modify the vibration state of the wave, thus causing a phase shift, ([136], Chapter 4.1.3; [152], Chapter 2.2.4). This surface reflectivity is a function of the radar system parameters, such as frequency, polarization, incidence angle of the emitted electromagnetic waves, and of the surface parameters, such as topography, local incidence angle, roughness, dielectric properties of the medium, and moisture ([116], Chapter 1.2.5).

However, the main drawback of SAR, as an active coherent imaging system, is the presence of *speckle*, a signal dependent granular noise that visually degrades the appearance of SAR images. Speckle is caused due to the presence of many elemental scatterers with a random distribution within a pixel, as it may represent various square meters on the ground. Yet the effect of speckle tends to weaken for very high-resolution systems, since the number of elemental scatterers within a pixel decreases [142]. The coherent sum of the many scatterers' amplitudes and phases results in strong fluctuations of the backscattering from pixel to pixel. To mitigate speckle a technique known as multi-look is commonly utilized, which is basically a non-coherent averaging of the SAR image ([152], Chapter 6.4). The number of looks, which is the number of independent samples included in the averaging, affects the evaluation of scattering mechanisms and characterizes radiometric resolution ([136], Chapter 3.13.3; [152],

Chapter 2.3). Although multi-looking causes a degradation in the image resolution, it greatly improves the interpretability of the SAR image. Speckle may severely diminish the performances of automated scene analysis and information extraction techniques. Therefore, despeckling or speckle reduction is generally applied, however, should be carefully designed to avoid spoiling useful information such as local mean of backscatter, point targets, linear features, and textures [6]. Commonly, there is a trade-off between either an efficient speckle reduction or the preservation of image structures and texture [149].
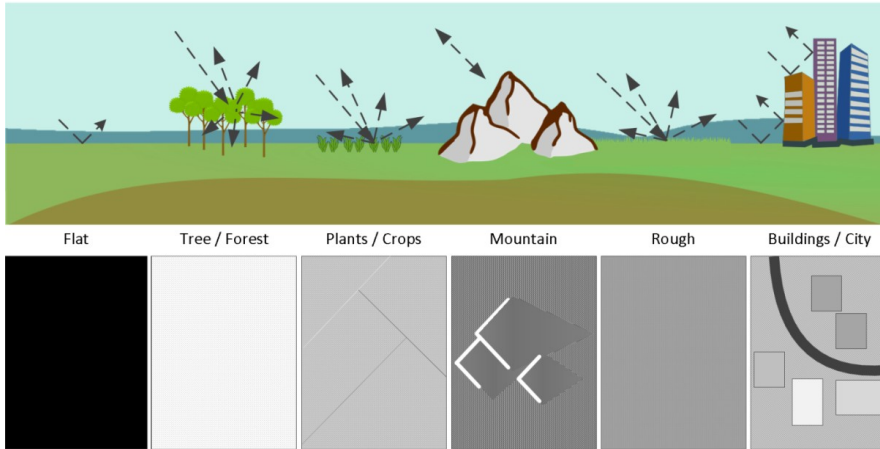
Figure 6 – Illustration of scattering mechanism over various different terrain types and their respective radar image appearances.

When reflected from the surface, the backscattered wave will feature different scattering mechanics due to diversity in surface characteristics. The main differentiation comes from the fact how the initiated waves interact with the underlying target surface and how much of the initially transmitted wave energy is reflected back ([165], Chapter 5). The more of the initial wave energy is reflected back to the receiver, the brighter it will appear in the generated radar image. In case of a flat and even surface (roads, water), the incident angle of the wave equals the angle of reflection so that the entire wave is directly reflected with a single bounce. Usually this is represented by a dark-tone in the radar image as no scatter is directed back to the radar sensor ([165], Chapters 5.3.1 and 5.7). When the wave hits a rough surface (e.g., grass, small crops), it is refracted due to the uneven surface and only a part of the initial wave is reflected back. This results in a mid-tone image representation ([165], Chapter 5.3.2). On man-made structures particularly buildings, the wave can be reflected more than once (double bounce) due to multiple flat and horizontal surfaces. Hence, most of the initial wave energy is scattered back, thus this generally is represented in bright-tones in the radar image ([136], Chapter 5.4.2). Obviously, the wave can further be reflected off multiple times (forest, urban- and city areas) either by trees and their branches or multiple buildings resulting in so-called volume scattering ([165], Chapter 5.4). Examples of the general scatter mechanics are illustrated in Figure 6.

Due to the complex nature of the surface under observation, the radar backscatter can be seen as a mixture of scattering mechanics ([165], Chapter 5.6), where their relative contributions are subject to the underlying surface roughness and the dielectric properties of the target. All these factors depend on radar wavelength, polarization, and incidence angle of the SAR transmitter ([165], Chapter 8.3). Particular low incidence angles maintain the coherence of the transmitted wave better than high incidence angles, which produce a more diffuse and less polarized signal ([136], Chapter 5.11). This and variations, due to topography or range position, can have influence on classification results ([116], Chapter 9.3).
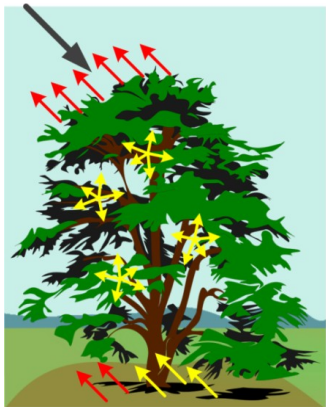
Figure 7 – Example of scattering types within tree canopy.
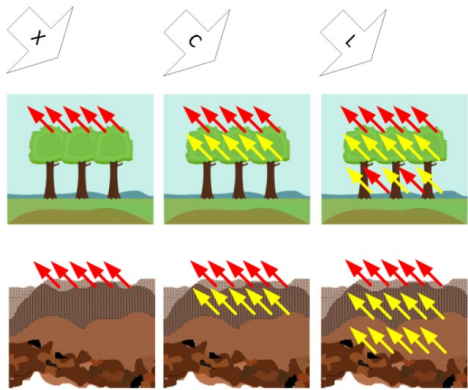
Figure 8 – Effects of different wavelength regarding penetration of various targets and surfaces.

The radar wavelength depends on the system and the used electromagnetic microwave frequency in the P-Band (0.3 GHz) up to K_a-Band (40 GHz) range. A lower frequency will result in a longer wavelength, which is capable of penetrating the target deeper. Due to interacting differently with the target, it generates random mixtures of scattering mechanics, which will result in different characteristics of a target for different microwave bands [38]. For example, in case of canopy (see Figure 7), the wavelength, and hence, penetration properties, will result in different possible interactions with the canopy such as surface scattering from the top, volume scattering from the inside as well as a combination of surface- and volume scattering from the ground. The radar backscatter at high frequencies such as K- to X-Bands is dominated by surface scattering from the top, whereas lower frequencies such as L-Band and P-Bands have relatively long penetration depth into vegetation and soil, enabling to extract information on the interior of the targets ([136], Chapter 1.3). Hence, different bands are considered for different applications. Figure 8 illustrates penetrations properties of different frequencies for a variety of surfaces.

A visual example of such penetration effects is shown for X- and L-Bands in Figure 9 over the Kojima district in Okayama, Japan. The area is mainly covered by fields with rice plants, which were almost fully-grown with a height approximately 60–90 cm at the time of the data

acquisition on 13 July 1999. Both L-Band and X-Band Pi-SAR images are decomposed into surface-, double-, and volume scattering. The L-Band image shows strong volume scattering for trees and bushes as the green dots as well as for the plants inside the Green houses. Double scattering is dominant for houses alongside roads and water channels with their respective banks. Lotus fields are a mixture of double- and volume scattering due to their respective height of around two meters. Rice fields are characterized by surface scattering due to the smaller and thinner plant size, thus, the longer wavelength penetrate them and being reflected from the ground. In the X-Band image, the dominant backscatter is volume scattering particularly for the rice and Lotus fields; and there are small amounts of double-bounce scattering from roads and houses compared to the L-Band.
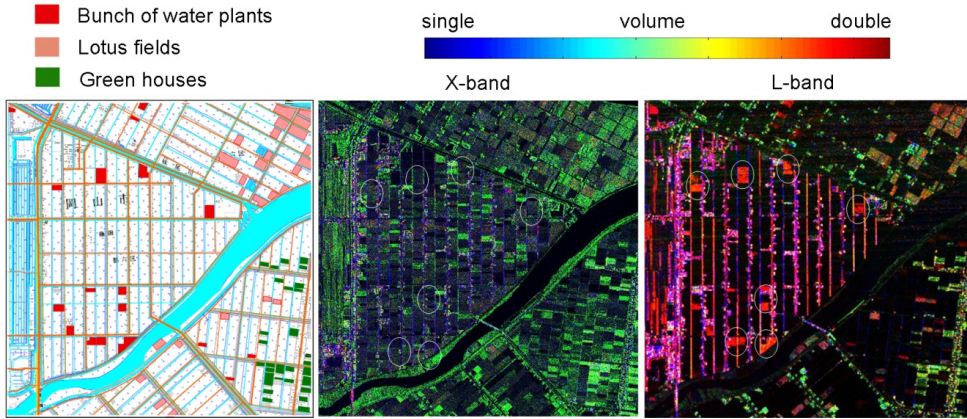
Figure 9 – Illustration of dominant backscattering for PiSAR L- and X-Band images. Reused with permission from Kazuo Ouchi [153].

In addition to the amplitude measuring the strength of the radar backscatter, polarimetric SAR (PolSAR) utilizes the polarimetry concept by measuring and interpreting the polarization of the transmitted and received electromagnetic waves [184]. The incident wave is transmitted with a particular polarization towards the target of interest. The wave is refracted by the target, which hereby transforms the incident wave into a scattered wave. Due to that interaction with the target, the scattered wave changes its polarization state and degree of the incident wave ([116], Chapter 3.1). The difference of the incident and backscattered waves contains the interesting information about the scattering properties and, eventually, the biophysical properties of the target itself. The transform of the incident wave into the scattered wave is described by the complex scattering matrix $[S]$:

$$\begin{bmatrix} E_H^r \\ E_V^r \end{bmatrix} = [S]\begin{bmatrix} E_H^t \\ E_V^t \end{bmatrix} = \begin{bmatrix} S_{HH} & S_{HV} \\ S_{VH} & S_{VV} \end{bmatrix}\begin{bmatrix} E_H^t \\ E_V^t \end{bmatrix}, \tag{1}$$

where E_H^t and E_V^t are the transmitted, E_H^r and E_V^r are the received electric fields of corresponding polarizations. $[S]$ is independent on the polarization of the incident wave and depends only on the physical and geometrical properties of the scatterer ([116], Chapter 3.1). The basic scattering process is illustrated in Figure 10.

This is by no means an exhaustive description rather than a brief overview about the underlying radar/SAR concepts regarding acquisition, backscattering, system properties, target characterization, and relations among them. More comprehensive summaries and details about microwaves, radar, SAR, and their interactions with surfaces can be found in [116], [165], [136], [38], [152], [203] covering in depth the math and physics behind them.
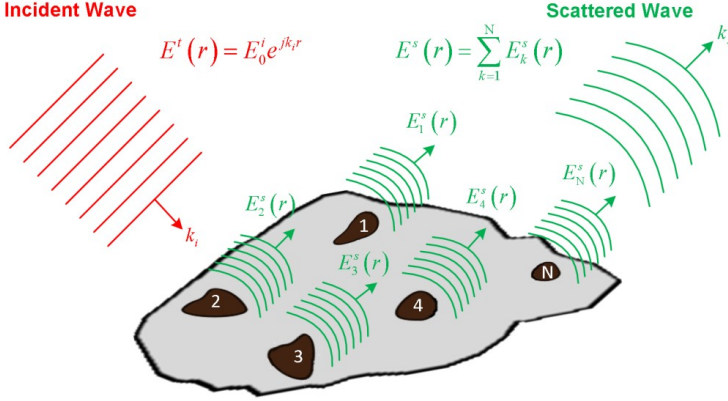
Figure 10 – Illustration of the incident wave interacting with a target resulting in the scattered wave.

2.2. POLSAR INFORMATION EXTRACTION

When acquiring data from radar systems, the effects of speckle noise and multiple scattering from the target surface require a multivariate statistical description. Therefore, finding a way of describing average or dominant scattering is beneficial for information extraction and tasks such as classification.

Information extraction process for PolSAR data can generally be divided into two feature categories. The first group belongs to features extracted directly from the polarimetric SAR data (scattering coefficients) and its unique representation as the scattering matrix $[S]$, from which the covariance matrix and the coherency matrix can be derived. The second group is based on the polarimetric target decomposition theorems, which try to describe the obtained average scattering as the sum of independent components, generally based on the covariance or coherency matrix, and can then be used as one source for information extraction. Particularly, SAR polarimetry is essential for extracting geo/bio-physical parameters for land-, snow and ice-, ocean-, and urban applications [142].

2.2.1. Scattering Matrix

From equation (1), the scattering matrix $[S]$ carries information generated by the interaction with the target. With linear horizontal (H) and vertical (H) polarizations for transmitting and receiving, $[S]$ can be expressed as

$$[S] = \begin{bmatrix} S_{HH} & S_{HV} \\ S_{VH} & S_{VV} \end{bmatrix}, \tag{2}$$

where reciprocity theorem applies in a monostatic system configuration (i.e., using a single antenna for both transmission and reception such as in airborne systems) yields $S_{HV} = S_{VH}$. The diagonal elements of $[S]$ are called *copolar* terms, since they relate to the same polarization for the incident and the scattered fields. The off-diagonal elements are known as *crosspolar* terms, as they relate to orthogonal polarization states ([116], Chapter 1.2.5). The elements of $[S]$ consists of amplitude, $|S_{mn}|$, and phase, ϕ_{mn}, of the backscatter described as $S_{mn} = |S_{mn}| \cdot \exp(i\phi_{mn})$: m,n=H,V. As the phase term is not of absolute value, it is commonly referenced against the HH polarization phase. This results in the amplitudes $|S_{HH}|$, $|S_{HV}|$, $|S_{VV}|$, and the phases, ϕ_{HV} and ϕ_{VV} as the main five parameters of $[S]$ to extract information about the scattering targets ([136], Chapter 5.3).

Based on $[S]$, the second order polarimetric representations of PolSAR data can be derived in form of the polarimetric covariance matrix $\langle[C]\rangle$ or coherency matrix $\langle[T]\rangle$, where angular brackets are the symbol used for time averaging. The resulting second order polarimetric descriptors of the covariance and coherency matrices are able to better characterize distributed scatterers and are employed to extract physical information from the observed scattering process. Due to presence of speckle noise and random vector scattering from surface or volume, PolSAR data are often multi-look processed by averaging *n* neighboring pixels. By using the Pauli based scattering vector, $\underline{k}$, and Lexicographic scattering vector, $\underline{\Omega}$, the multi-look coherency matrix $\langle[T]\rangle$ and covariance matrix $\langle[C]\rangle$ can be written as

$$\langle[T]\rangle = \left\langle \underline{k} \cdot \underline{k}^{*T} \right\rangle,$$
$$\langle[C]\rangle = \left\langle \underline{\Omega} \cdot \underline{\Omega}^{*T} \right\rangle. \tag{3}$$

Both $\langle[C]\rangle$ and $\langle[T]\rangle$ are 3×3 Hermitian positive semi definite matrices, and contain the same information about the polarimetric scattering amplitudes, phase angles and correlations. Anyhow, $\langle[T]\rangle$ is considered to be closer to the physical and geometrical scattering properties, whereas $\langle[C]\rangle$ is directly related to the system measurables ([116], Chapters 3.3 and 6.1).

2.2.2. Polarimetric Target Decompositions

When investigating targets of interest, a multivariate statistical description is necessary to compensate for the effects of mixing surface and volume scatterings. Therefore, finding the *average* (or *dominant*) scatterings is highly advantages for classification. Target decompositions try to find such average scattering as the sum of individual components, which provide a better physical interpretation. Over the years, various polarimetric target decompositions (TDs) have been developed [39] and there is still extensive research in developing new [5], [7] and improved target decompositions [169] for more accurate information extraction and terrain discrimination.

Two theories of target decomposition (TD) can be distinguished: *coherent* and *incoherent* target decomposition. Coherent target decomposition (CTD) theory completely characterizes the polarized scattered waves for which the scattering matrix holds the fully polarimetric information. However, generally, the scattered wave is partially polarized and extracting geophysical parameters from a natural target surface with incoherent scatterers is usually of more interest. Hence, the development of the incoherent target decomposition (ICTD) theory was developed to represent a target-averaged matrix into a sum of single scattering matrices, which provides a better interpretation of the underlying scattering and simpler ways of extracting the geophysical parameters from the measured radar data [15], [58].

2.2.2.1 Coherent Target Decompositions

The CTD theorems aim to express the scattering matrix [S] measured by the radar as the combination of scattering responses of coherent scatterers. So far, [S] combines the entire backscattering information and CTD algorithms treat this backscattering information based on the mixture of the general scattering mechanisms:

$$[S] = \sum_{i=1}^{k} \alpha_i \cdot [S]_i ,$$

(4)

where each scattering mechanism $[S]_i$ is weighted by a complex coefficient α_i. The main goal is to decompose the entire backscattering information into standard target contributions (planes, dihedral, and helices) and extract components with particular algebraic properties such as reciprocity and symmetry.

One way to analyze coherent scatter targets is the Pauli decomposition [39], which expresses [S] as the sum of the Pauli matrices,

$$[S] = \begin{bmatrix} S_{HH} & S_{HV} \\ S_{VH} & S_{VV} \end{bmatrix} = \alpha [S]_a + \beta [S]_b + \gamma [S]_c \text{ where}$$

$$\left\{ [S]_a = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, [S]_b = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, [S]_c = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right\}$$

(5)

with $\alpha = (S_{HH} + S_{VV})/\sqrt{2}$, $\beta = (S_{HH} - S_{VV})/\sqrt{2}$, $\gamma = \sqrt{2} S_{HV}$. Hence, by means of the Pauli decomposition, all polarimetric information in [S] can be represented by combining the intensities $|\alpha|^2$, $|\beta|^2$, and $|\gamma|^2$, which determine the power of different types of scatterers such as single- or odd-bounce scattering, double- or even-bounce scattering, and orthogonal polarization by volume scattering.

The Krogager decomposition [113] exploits the fact that it is possible to uniquely decompose any complex symmetric radar target scattering matrix into the physical interpretation of three components, in this particular case a sphere, an oriented diplane, and a right or left wound helix. In the linear orthogonal basis (H,V) for the scattering matrix, the Krogager decomposition has the formulation as follows:

$$\left[S_{(H,V)} \right] = e^{j\varphi} \left(e^{j\varphi s} k_s [S]_s + k_d [S]_d + k_h [S]_h \right),$$

(6)

where k_s, k_d, and k_h correspond to the weights of the sphere, the diplane, and the helix components, respectively. Its main goal is to resolve different scattering characteristics independent from the incidence angle.

2.2.2.2 Incoherent Target Decompositions

Alternatively, the ICTDs employ the second order polarimetric representations of PolSAR data (such as covariance matrix $\langle[C]\rangle$ or coherency matrix $\langle[T]\rangle$) to characterize distributed scatterers taking into account the polarization state of the backscattered wave. ICTD algorithms often employ algebraic approaches based on the eigenvalue and eigenvector analysis of $\langle[C]\rangle$ or $\langle[T]\rangle$ decomposing $[S]$ into the general scattering mechanisms single-, double-, and volume scattering.

The phenomenological Huynen decomposition [94] represents the first attempt to use decomposition theorems for the analysis of distributed scatterers. In its parametrization for $\langle[T]\rangle$, it consists of nine independent parameters allowing a physical interpretation of the target under consideration. The three elements A_0, B_0+B, and B_0-B represent surface, double-bounce, and volume scattering in the case of distributed targets and otherwise relate to the total scattered power from the regular (smooth, convex) parts of the scatterer, the total symmetric and non-symmetric depolarized power, respectively, for pure targets.

Based on $\langle[T]\rangle$, the Cloude-Pottier decomposition [39] applies eigenanalysis such as

$$\langle[T]\rangle = \lambda_1 e_1 e_1^{*T} + \lambda_2 e_2 e_2^{*T} + \lambda_3 e_3 e_3^{*T}, \tag{7}$$

where $\lambda_1 > \lambda_2 > \lambda_3 \geq 0$ are real eigenvalues and the corresponding orthonormal eigenvectors e_i representing three scattering mechanisms are:

$$e_i = e^{i\phi_i} \left[\cos\alpha_i, \sin\alpha_i \cos\beta_i e^{i\delta_i}, \sin\alpha_i \sin\beta_i e^{i\gamma_i} \right]^T. \tag{8}$$

Furthermore, Cloude and Pottier defined entropy H, a set of four angle averages, $\bar{\alpha}$, $\bar{\beta}$, $\bar{\delta}$, and $\bar{\gamma}$, and anisotropy A for the analysis of the physical information related to the scattering characteristics of a medium as

$$H = -\sum_{i=1}^{n} p_i \log_n p_i \qquad \text{where} \qquad p_i = \frac{\lambda_i}{\sum_{k=1}^{n} \lambda_k} \tag{9}$$

$$\bar{\alpha} = \sum_{i=1}^{n} p_i \alpha_i, \quad \bar{\beta} = \sum_{i=1}^{n} p_i \beta_i, \quad \bar{\delta} = \sum_{i=1}^{n} p_i \delta_i, \quad \bar{\gamma} = \sum_{i=1}^{n} p_i \gamma_i, \tag{10}$$

$$A = \frac{p_2 - p_3}{p_2 + p_3}, \tag{11}$$

with $n = 3$ for monostatic backscatter problems ([116], Chapter 7). For a multi-look coherency matrix, the entropy, $0 \leq H \leq 1$ represents the randomness of a scattering medium between isotropic scattering ($H = 0$) and fully random scattering ($H = 1$), while the average angle $\bar{\alpha}$ can be related to the target average scattering mechanisms from a single-bounce (or surface) scattering ($\bar{\alpha} \approx 0$), dipole (or volume) scattering ($\bar{\alpha} \approx \pi/4$), and double-bounce scattering ($\bar{\alpha} \approx \pi/2$). Due to the basis invariance of the target decomposition, H and $\bar{\alpha}$ are roll invariant, hence

they do not depend on the orientation of the target in the radar line of sight. Moreover, the parameters $\bar{\alpha}$, $\bar{\beta}$, $\bar{\delta}$, and $\bar{\gamma}$ are practically insensitive to parameters such as surface roughness, the moisture of vegetation cover or certain calibration artifacts ([152], Chapter 5.10.3.3). The quality of their estimation is correlated with the local value of entropy. Additionally, information about a target's total backscattered power can be determined by the so-called *Span* defined as

$$Span = \sum_{i=1}^{3} \lambda_i \,. \tag{12}$$

Entropy *H*, estimate of the average angle $\bar{\alpha}$, and the *Span* calculated by the above incoherent target decomposition method have been commonly used as polarimetric features of a scatterer in many target classification schemes [4], [31], [57], [64], [110], [128], [162], [175], [180], [225], and [232].

The Touzi decomposition [185] similar to the Cloude-Pottier's decomposition, is also, based on the incoherent characteristic decomposition of $\langle[T]\rangle$. For a reciprocal target, it permits the representation of $\langle[T]\rangle$ as the incoherent sum of up to three coherency matrices $[T]_i$ representing three different single scatterers, each weighted by its appropriate positive real eigenvalue λ_i. Compared to the Cloude-Pottier decomposition, Touzi's decomposition differentiates in using a roll invariant coherent scattering model for the parametrization of the coherency eigenvectors in terms of their unique target characteristics. For non-interferometric applications, the absolute target phase Φ_s can be ignored, and the coherent scatterer is uniquely characterized with five independent parameters, namely, α_s , $\Phi_{\alpha s}$, ψ, τ_m , and *m* with α_s and $\Phi_{\alpha s}$ being the polar coordinates of the symmetric scattering type; ψ, τ_m, and *m* are Huynen's orientation angle, helicity, and maximum amplitude parameters.

Van Zyl first considered eigenvector / eigenvalue analysis of $\langle[C]\rangle$ to describe symmetrical natural terrain in the monostatic case [207] as

$$\langle[C]\rangle = \sum_{i=1}^{3} \lambda_i \underline{u}_i \cdot \underline{u}_i^{*T} = [C]_s + [C]_d + [C]_v \,, \tag{13}$$

where λ and $\underline{u}$ are eigenvalues and eigenvectors, respectively. This expression represents a three components scattering model where the first two eigenvectors represent equivalent scattering matrices that can be interpreted with single and double bounce scatterings, and the third one corresponds to volume scattering. Furthermore, based on this eigenvalue analysis, Kim and van Zyl introduced an estimation of (forest) biomass from PolSAR data, the so-called Radar Vegetation Index (*RVI*) [105] (also generally characterized by Arii, van Zyl, and Kim [8]) defined as

$$RVI = \frac{4 \min\left(\lambda_1, \lambda_2, \lambda_3\right)}{\lambda_1 + \lambda_2 + \lambda_3} \left(= \frac{8\sigma_{hv}}{\sigma_{hh} + \sigma_{vv} + 2\sigma_{hv}} \right), \tag{14}$$

where the sigmas are the cross- and co-polarization radar scattering cross sections. The representation by the eigenvalues is the general characterization of the *RVI* whereas in the brackets is the alternative but more restrictive definition [105] which is only valid for a medium with reflection symmetry and can be expressed as $RVI = 4\lambda \cdot 2$ [8].

Van Zyl's decomposition of $\langle[C]\rangle$ is also considered to be the starting point for the so-called model-based decompositions such as Freeman [66] and Yamaguchi [216] decompositions. The Freeman decomposition describes the covariance matrix, $\langle[C]\rangle$, as a combination of three simple scattering mechanisms, namely, volume scatter from a cloud of randomly oriented dipoles, even-bounce (double-bounce) scatter from a pair of orthogonal surfaces with different dielectric constants and Bragg (single-bounce) scatter from a moderately rough surface.

Freeman's decomposition follows the same expression for $\langle[C]\rangle$ as van Zyl, see equation (13). The scattering powers for single-bounce, P_s, double-bounce, P_d, and volume scattering, P_v, are estimated as

$$P_s = f_s\left(1+|\beta|^2\right), \ P_d = f_d\left(1+|\alpha|^2\right) \text{ and } P_v = \frac{8f_v}{3}, \tag{15}$$

where f_s, f_d, and f_v correspond to the contribution of single-bounce, double-bounce, and volume scattering of $\langle[C]\rangle$, α and β correspond to surface and double bounce scatter, respectively. This composite scattering model is generally used to discriminate the polarimetric backscatter of different natural surface cover types from naturally occurring scatterers' particularly forest-like vegetation.

Yamaguchi et al. [216] introduced a four-component scattering model based on $\langle[C]\rangle$ (single and double-bounce, volume and helix scattering power contributions) which extends the three-component decomposition method introduced by Freeman. The helix scattering term is added to address the co-pol and the cross-pol correlations, which generally appear in complicated geometric scattering structures and is non-existing for natural distributed scatterer. Hence, this term is mainly relevant for describing man-made targets in urban areas. The scattering powers for P_s and P_d are equivalent to Freemans', i.e., see equation (15), whereas P_c is defined and P_v is modified accordingly,

$$P_c = 2\left|\text{Im}\left\langle S_{HV}^*\left(S_{HH}-S_{VV}\right)\right\rangle\right|, \ P_v = \frac{15}{2}\left(\left\langle|S_{HV}|^2\right\rangle-\frac{f_c}{4}\right), \tag{16}$$

with Im( ) being the imaginary part of a complex matrix. However, there is a drawback, as for all model-based decompositions, in overestimating the volume scattering caused by a large cross-polarization component. To overcome this, a rotation method of the coherency matrix has been proposed by Yamaguchi et al. [217]. Anyhow, the discrimination of vegetation and oriented buildings within the same volume scattering is a still existing problem, which has been addressed to a certain degree by the extended volume scattering model [169]. The covered target decompositions are considered more relevant in this thesis and more details regarding them and other target decompositions can be found in [39] and [116].

2.3.   TEXTURE FEATURES

The traditional PolSAR features such as the scattering coefficients and TD components are based on the physical, geometrical, and scattering properties of targets. Generally, such PolSAR features provide a description purely on a pixel-based level, which can limit discrimination for similar class types due to variations in their terrain surfaces (e.g., different forest

types or man-made areas). More than three decades ago, Shanmugan et al. [173] evaluated Gray-Level Co-occurrence Matrix (GLCM) features [83] for radar images and Ulaby et al. [204] added first, second order statistics over SAR intensity images to evaluate and classify SEASAT-SAR and SIR-A data. By employing texture features, the underlying spatial variability of a particular terrain class can be better exploited. These earlier analysis of texture as a feature in SAR image classification concluded that "*texture may, in fact, be more useful than image tone in interpreting radar images*" [204]. Even though this was related to simple image intensities, the general assumption is still valid, particularly for single-pol and dual-pol SAR imagery. Therefore, exploiting pixel relations in close vicinity using texture features (e.g., GLCM) is highly valuable and contributes a major discrimination capability [173]. Over time, the incorporation of textural information as an image processing method has become the most popular feature in SAR image classification. The following sections detail texture features considered in the experiments of the remaining chapters.

Pixel neighborhood gray levels

70	87	227	65	157	73	135
173	149	245	208	121	193	199
167	57	140	62	90	192	238
41	192	35	237	212	97	33
30	65	38	89	149	145	145
127	129	66	50	140	19	120
245	178	214	64	234	14	3

A

B

$\chi(5, 90°)$

$\chi(3, 45°)$

$\chi(2, 135°)$

$\chi(4, 0°)$

Pixel neighborhood scaled to 8 levels

3	3	7	3	5	3	5
6	5	8	7	4	6	6
6	3	5	3	3	6	6
2	6	2	8	7	4	2
2	3	2	3	5	5	5
4	5	3	2	5	2	4
8	6	7	3	7	1	1

C

GLCM $\chi(1,0)$

1	0	0	0	0	0	0	0
0	0	2	1	1	1	0	1
0	2	2	0	4	1	2	0
0	1	0	0	1	1	0	0
0	1	3	0	2	0	0	1
0	1	1	0	1	1	1	1
1	0	2	2	0	0	0	0
0	0	0	0	0	1	2	0

D

Figure 11 – Example of generating one gray-level co-occurrence matrix over image region in (A) with common directional and angular pixel relationships $\chi(\Delta, \angle)$ shown in (B). (C) shows (A) scaled to eight levels resulting in the GLCM with $\chi(1, 0)$ in (D).

2.3.1. Gray-Level Co-occurrence Matrix

An earlier statistical approach was the gray-level co-occurrence matrix (GLCM) by Haralick [83], where the texture descriptor is gained from second order statistic. The GLCM computed within a region of interest provides information about directional and angular pixel relationships $\chi(\Delta, \angle)$ by defining how often two gray level values g_i and g_j are separated via a certain displacement d=(dx, dy). Figure 11B illustrates such common displacements and the application of multiple displacements enables the opportunity to capture different existing texture properties. Note, that the gray level values are scaled for lower computational complexity. A GLCM entry corresponds to the number of occurrences of a pair of scaled gray levels separated by a displacement d=(dx, dy). Thus, an GLCM entry, M(i, j), is given as M(i, j) = |{ (r, s), (t, v) : I(r, s) = i, I(t, v)=j }| where (r, s), (t, v) ∈ N×N, (t, v)=(r+dx, s+dy), and |·| is the cardinality of a set.

Based on the GLCM, features can be computed such as energy, entropy, contrast, and homogeneity, all of which describe the underlying texture properties. The feature vector size, therefore, depends on the range of distance vectors and the amount of properties calculated from the co-occurrence matrix. The texture description power of GLCM depends on the combination of selected distance vectors where too few will provide a rather poor description and too many will increase the computational costs during feature extraction. Figure 11 shows an example generating one gray-level co-occurrence matrix over gray levels in an image region with common directional and angular pixel relationships $\chi(1, 0)$.

2.3.2. Ordinal Co-occurrence Matrix

The Ordinal Co-occurrence Matrix (OCM) [155] approach is a combination of the traditional GLCM and ordinal descriptors (ordinal methods for texture description). However, one difference of OCM compared to GLCM is the application of ordinal relationship between the pixels rather than just gray level values. Another alternation is that instead of using just one seed pixel to calculate the displacements, multiple seed pixels are able to capture more details within a given image region, as depictured in the left side of Figure 12. This will obviously come with an increase of computational complexity. The constructed features represent the occurrence frequency of certain ordinal relationships such as *greater*, *equal*, *smaller* at different distances *d* and orientations *o* over all the used seed points within a particular image region with an example illustrated in Figure 12 (right hand side). Hence, the theoretical feature dimension yields to $d \times o \times 3$ as related to the number of distances *d* and orientations *o* used to describe the three ordinal relationships. The comparison between two different OCMs can then be done directly by summing the differences of the corresponding distance and orientation relationship matrices. Even though, OCM has not been applied as much as GLCM to SAR images, it is anticipated that OCM will perform on a similar level or even better. It has been shown that OCM outperforms GLCM in texture classification [155] particularly for coarse textures and irregular pattern as found in man-made areas whereas they both achieve

similar results for fine textures and highly repetitive structured pattern as in water, forest, or other vegetation.
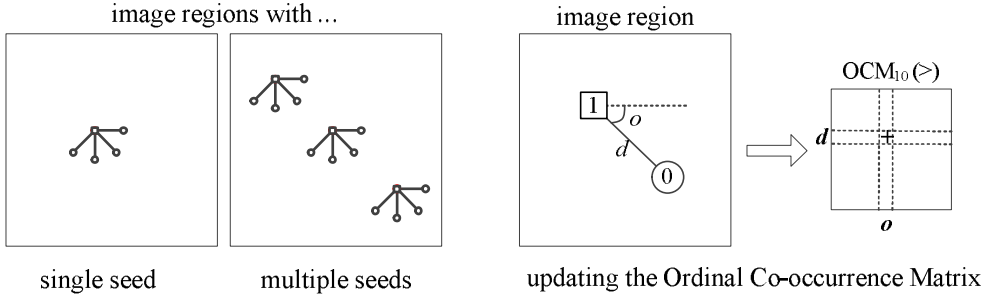
Figure 12 – Ordinal Co-occurrence Matrix with different seed setup to calculate displacements of four orientations from squared seed pixel (left) and example of accumulation and updating the relationship matrix OCM_{10} when the seed pixels is greater than the displacement pixel for distance d and orientation o (right).

2.3.3. Gabor Wavelets

The popular and powerful signal processing texture descriptor called Gabor filter [135] presents a multi-resolution approach and has been previously applied to SAR image classification [101], [220], and [226]. The main idea is to process an image region by a bank of filters at different scales and orientations (i.e., multi-channel as see in Figure 13) where filtering can be applied in either spatial or frequency domain. An image region $I(x,y)$ that is filtered with a Gabor filter g_{mn} results in its Gabor wavelet transform W_{mn}, which captures different frequency and orientation information about an underlying texture [135]. W_{mn} can be formulated as

$$W_{mn}(x,y) = \int I(x_1, y_1) g_{mn}^*(x - x_1, y - y_1) dx_1 dy_1.$$ (17)

For each scale and orientation, the magnitude response $|W_{mn}|$ is calculated as an output from which the first and second order moments are computed as the texture features. Thus, the feature vector is rather small and is formed per scale and orientation.
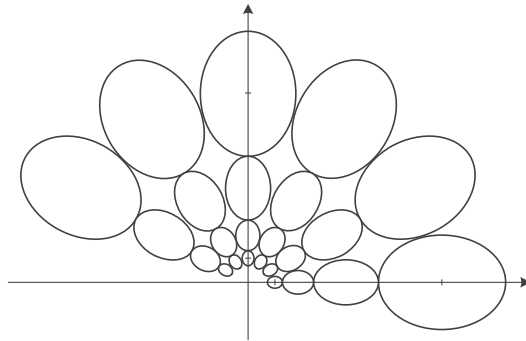
Figure 13 – Contours indicate the half-peak magnitude of the filter responses for six orientations and four scales as used for Gabor Wavelets.

2.3.4. Multiple Local Pattern Histogram

Compared to the previous texture features, the Multilevel Local Pattern Histogram (MLPH) [46] was developed especially to exploit the elementary properties of SAR image texture. MLPH describes the size distributions of bright, dark, and homogenous patterns appearing in a moving window at various intensities capturing local and global structural information. The local information is captured within a pixel neighborhood window, applying a threshold to determine pixels with larger, smaller, or equal intensities based on the center pixel value and its range of $\pm T$. This results in three binary matrices considering white as the foreground. These matrices are processed individually resulting in sub-histograms counting the number of foreground objects regarding their different sizes. All three sub-histogram are then concatenated to obtain the local pattern histogram (LPH). The original LPH dimension is related to the dimension of a single sub-histogram, w^2, which depends on the chosen window size w. Thus, a histogram-combining strategy into K bins is applied to obtain a more compact representation. Dai et al. [46] claim it is best to require that higher order bins represent a wider range than lower-order bins as higher-order bins correspond to rarer larger pattern more difficult to discriminate by humans. Therefore, they proposed a growth rate B for the K bins. The process of generating LPH is illustrated in Figure 14. Global information is captured by calculating the LPH multiple times for M different threshold ranges T of the local window. The final dimension of the MLPH is $M \times 3 \times K$ bins. Their evaluation of a four-class problem over single-pol TerraSAR-X data showed that it can provide high discrimination for urban and forest terrain.
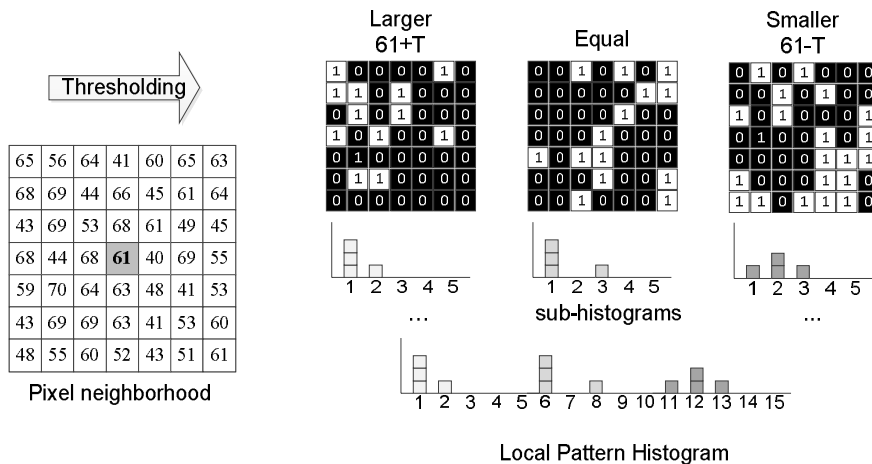
Figure 14 – Example of computing a Local Pattern Histogram over 7×7 pixel neighborhood window, threshold T=3, and K=5 and B=3 for the compact histogram representation.

2.3.5. Local Binary Pattern and Edge Histogram

Another rather popular texture descriptor in computer vision and texture/image classification is Local Binary Pattern (LBP) [150]. LBP is a rotation invariant texture feature extracted over

an n-by-n pixel neighborhood. It works directly on pixels and their neighborhood as shown in Figure 15(a). The neighboring pixels are then compared to the current center pixel as a threshold obtaining a texture pattern, as shown in Figure 15 (b) and binomial factors (e.g., see Figure 15 (c)) are multiplied by the neighboring positions greater than or equal to the center pixel as in Figure 15 (d). Finally, the sum of the binomial factors yields the LBP value being assigned to the center pixel. This procedure is applied to every pixel in the image resulting in a raw 256-bin histogram, which can further be quantized to obtain a more compact representation. To acquire a rotation invariant representation, LBP values can be combined as they represent the same uniform texture pattern but in a rotated fashion. This can be achieved by circularly shifting the original texture pattern until its minimum decimal value is attained, illustrated in the bottom of Figure 15. This results in 36 unique rotation invariant local binary patterns for a 3×3 neighborhood per pixel. One of the advantages of LBP is its simple design that still provides a powerful descriptor for texture classification.

Similarly, to LBP, the Edge Histogram Descriptor (EHD) [134] is mainly applied to content description in image retrieval and classification tasks. As the name suggests, EHD is an edge-based texture feature defined within the family of MPEG-7 descriptors [154]. EHD represents the spatial distribution of five types of edges, four directional edges and one non-directional edge. In its original form, a given image is first divided into sub-images, and local edge histograms for each of these sub-images are computed. Edges are roughly grouped into five categories: vertical, horizontal, 45° diagonal, 135° diagonal, and isotropic (random) applying simple edge detectors over 2-by-2 image pixel blocks within each sub-image. Those image blocks whose edge strengths exceed a certain minimum threshold are used in computing the histogram. Thus, each local histogram has five bins corresponding to the aforementioned five categories. The local histograms from the sub-images are combined to generate the five-bin edge histogram representation of the entire image. The procedure for extracting the EHD is illustrated in Figure 16.
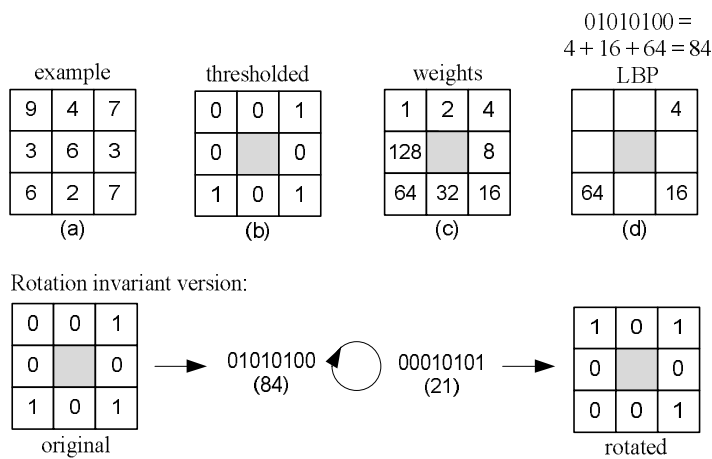
Figure 15 – Computation of rotation invariant Local Binary Pattern (LBP) value over 3×3 neighborhood.
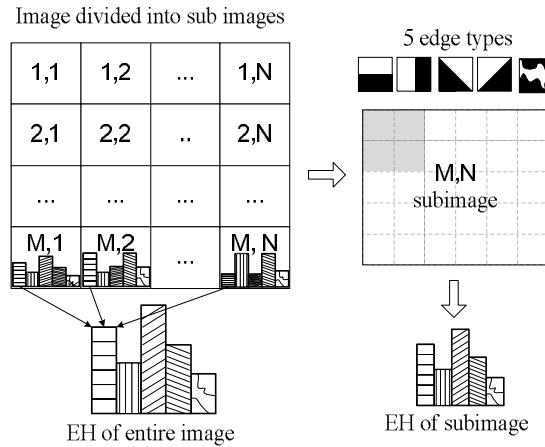
Figure 16 – Accumulation of the Edge Histogram.

2.4. ANALYSIS, INTERPRETATION, AND ASSESSMENT

Over the past years, the most straightforward approach has been to apply the polarimetric SAR (PolSAR) data (scattering coefficients) directly or the generated covariance / coherency matrix as the underlying features in the PolSAR image classification process. In early approaches the basic scattering coefficient information of polarimetric data such as HH, HV, VV over single-band data or various combinations over multi-band data have been used [2], [30], [49], [82], [86], [115], [119], [128], [177], [190], and [192]. Due to the nature of polarimetric data, the covariance matrix provides the complete polarimetric information in amplitude and phase variance as well as correlation for all scattering elements and has been utilized as direct feature itself [2], [29], [81], [119], [178].

Obviously, fully polarimetric (quad-pol) data provide more information regarding the target ground characteristics compared to partially polarimetric data such as dual- and single-pol, which will result in better discrimination in terms of classifying the target under observation. This has been investigated and proven in several studies over AIRSAR data [86], [119], ALOS PALSAR data [128], [190], and EMISAR AgriSAR data [177]. Moreover, quad-pol data enable the application of polarimetric target decompositions see Section 2.2.2 and [39] to describe the average backscattering obtained by the sum of independent components, which provide a better interpretation of the underlying scattering from the measured radar data. Various single TDs have been employed for LULC classification either alone [57], [85], [139], or in combination with the basic scattering coefficients [72], [128], [218]. With more and more TDs available, the investigation of multiple TDs for classification became popular in recent years with and without combination of basic scattering coefficients [31], [45], [56], [128], [162], [175], [180], and [232]. It can be concluded that quad-pol data are preferable when it

comes to SAR image classification; however, it might not always be available or practical in a SAR system to have quad-pol capabilities.

With the introduction of texture features by [173] for radar images and [204] over SAR intensity images, various evaluation and classification approaches over SAR images using textural information were reported with GLCM being the dominant method [51], [64], [101], [156], [162], [170], [186], [218], [220], and [225]. Additionally, investigations also included Fractal features [48], [170], Wavelets and Semivariogram [48] as well as Gabor Wavelets [101], [220] and Histogram of Gradients [220]. Dai et al. [46] proposed the Multilevel Local Pattern Histogram (MLPH) as a novel descriptor of SAR images, which captures local and global structural information while being robust to speckle noise. Furthermore, Local Primitive Pattern (LPP) has been introduced for SAR image classification [11] where local primitives (LPs; local homogeneous connected components) are formed using an adaptive neighborhood rather than the traditional fixed size window approach. The new feature considers the LPs and their relationships with their neighboring primitives. Both features, MLPH and LPP, demonstrated their performances by outperforming GLCM and Gabor Wavelets in their experiments.

Texture has proven as a valuable resource for SAR image classification and many evaluations have shown that texture features and their combination with PolSAR features helped to achieve improved LULC classification. Particularly in the case of dual-pol ([50], [56], [114], [124], [126],) and single-pol ([13], [54], [103], [176], [211], [213]) data, the addition of texture has been extensively applied as their PolSAR features are not as discriminative as quad-pol data; and, therefore, extra features are needed to provide better discrimination for an enhanced interpretation and analysis. Even though texture is the most used additional visual feature, there is ongoing investigation in the direction of polarimetric time-frequency signatures of scatterers for the classification of man-made targets [61] as well as applying bag-of-features and pyramid-representation over single channel TerraSAR-X data [221].

The evaluation and assessment of the achieved classification performances and accuracies are measured by quantitative criteria [40]. One common criterion for such an assessment in remote sensing is the error matrix (or confusion matrix) [41], [179], which can then be used to compute various accuracy statistics. The diagonal elements of the error matrix represent correct classification for individual terrain classes, whereas the off-diagonal elements represent their respective classification errors. Generally, the user and producer accuracy are two widely used measures of class accuracies obtained from the error matrix. The producer's accuracy relates to the probability, where a certain terrain area is classified as that particular terrain class. On the other hand, the user's accuracy denotes the probability that a sample pixel labeled as a certain terrain class does really belong to the ground truth of this terrain class. The user accuracy is commonly used as classification agreement or accuracy, which represent the probability of a reference pixel from the provided ground truth being correctly classified. Besides that, the kappa coefficient [166] has been used as a measurement to determine the agreement between observed and predicted values and whether this agreement is by chance. Hence, it provides a score how much consent there is between user and producer measures in

the error matrix [40]. The kappa statistics have been widely applied yet there is the argument and discussion that kappa can be misleading and the error matrix should instead be summarized with two simpler parameters: quantity disagreement and allocation disagreement [160]. Over the years, there have been various discussions and recommendations regarding results and accuracy assessments using remote sensing data [65], [40].

2.5. SAR IMAGE DATA

There are different PolSAR systems supporting various polarimetric modes. This section provides information about the SAR image data used in the experiments and evaluations throughout the remaining chapters. Selected PolSAR images are taken from one airborne system (NASA/Jet Propulsion Laboratory AIRSAR) [129] and three space-borne systems (Canadian Space Agency RADARSAT-2 [141], TerraSAR-X [112], and COSMO-SkyMed [183]). The AIRSAR system supports (full) polarimetric modes for C-, L-, and P-Band where we primarily focus on the L-Band; RADARSAT-2 works in C-Band also supporting the full polarimetric mode. The TerraSAR system mainly provides dual- and single-pol image data operating in the X-Band yet it can also be configured to provide fully polarimetric data, which is in an experimental stage. As for the COSMO-SkyMed system, its primary configuration provides partially polarimetric data using X-Band in form of dual-and single-pol data. All PolSAR images to be used within this thesis are briefly introduced with time, location, resolution, and their respective available ground truth data.

2.5.1. Flevoland, AIRSAR, L-Band

The four-look fully polarimetric L-Band data of Flevoland, The Netherlands, were collected in mid-August 1989 during MAESTRO-1 Campaign with a size of 1024×750 pixels and a resolution of around 12×6 meters in the azimuth and slant direction, respectively. This particular region has been extensively used as a test side for crop and land classification over the past years with well-established ground truth data [223] of 15 classes as shown in Figure 17. The incidence angles are from 20° to 44° in the near to far range. The majority of crop fields to be classified are within an 18° span of incidence angles. It is expected that this small variation will not affect classification ([116], Chapter 8.8.1.2).

2.5.2. Flevoland, RADARSAT-2, C-Band

This georeferenced RADARSAT-2 fully polarimetric SAR image at fine quad-pol mode (10×5 meters resolution) of Flevoland, The Netherlands, were taken in April 2008. Its format is single look complex and the incidence angle varies around 28-30°. The selected scene is an around 1600×2400 pixel sub-region, which mainly contains four terrain classes: *Woodland/Forest*, *Cropland*, *Water*, and *Urban* area [218] with an emphasis on natural and agricultural terrain. The Pauli color-coded image and the ground truth data are shown in Figure 18.
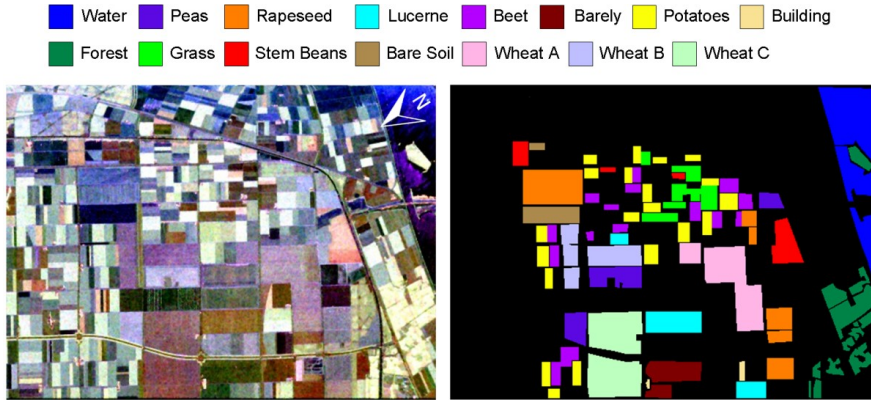
Figure 17 – Flevoland AIRSAR image (L-Band), Pauli color-coded image (left) and used ground truth (right). The class legend for the ground truth is given on the top.
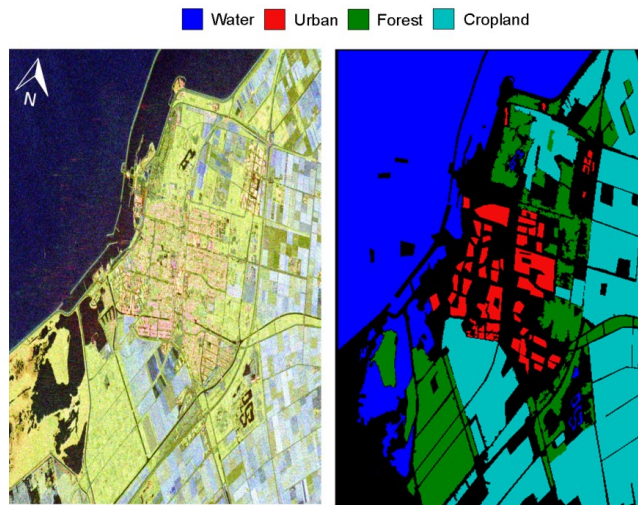
Figure 18 – Flevoland RADARSAT-2 image (C-Band), Pauli color-coded image (left) and used ground truth (right).). The class legend for the ground truth is given on the top.

2.5.3. San Francisco Bay Area

The area around the bay of San Francisco with the Golden Gate Bridge is probably one of the most used scenes in PolSAR image classification over the past decades. It provides a good coverage of both natural (e.g., sea, scrubs, forests) and man-made targets (e.g., buildings, streets, parks, golf course). As no real ground truth data by on-site visits is available, we generated the ground truth data by visually inspecting radar and optical data as well as using USGS National Land Cover Datasets (NLCD) 1992 [210] and 2006 [69]. Therefore, note that the ground truth accuracy cannot be 100 percent guaranteed. For instance, the different man-made classes may also cover trees (planted alongside roads or gardens of houses), thus classification is performed by considering the majority terrain type.
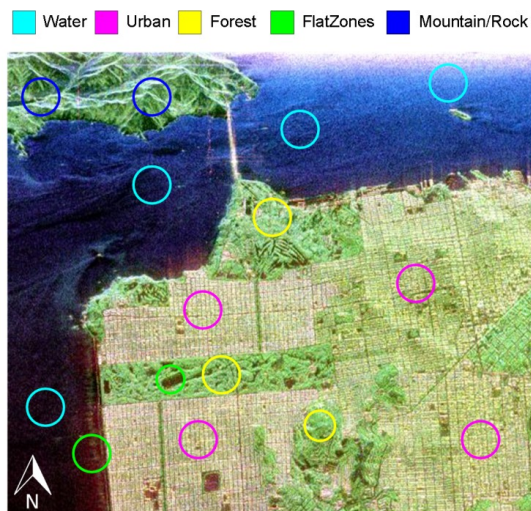
Figure 19 – San Francisco Bay AIRSAR image (L-Band), Pauli color-coded image superimposed with areas representing the five terrain classes. The class legend for the ground truth is given on the top.
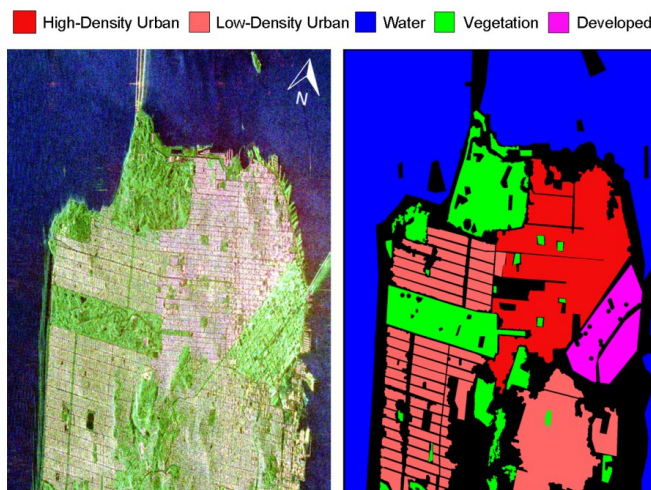
Figure 20 – San Francisco Bay RADARSAT-2 image (C-Band), Pauli color-coded image C-Band (left) and used ground truth (right). The class legend for the ground truth is given on the top.

The NASA/Jet Propulsion Laboratory Airborne SAR (AIRSAR) L-Band data of the San Francisco Bay area from 1988 is mainly used for visual evaluation. This original four-look processed fully polarimetric SAR image has dimensions of 900×1024 pixels with a spatial resolution of about 10×10 meters. The incident angles span from $10°$ to $60°$, as visible in the water regions. We defined five distinct classes for natural area (such as *Water – sea, Mountain – cliffs – rocks, Forest – trees, Flat Zones* such as beach, grass) and *Urban* area (build-

ings, streets, roads) targets with a more complex inner structure. The Pauli coded pseudo color image superimposed with same ground truth is given in Figure 19.

The RADARSAT-2 fully polarimetric SAR image at fine quad-pol mode (single look complex, georeferenced, and 10×5 meters resolution) of the San Francisco Bay was taken in April 2008 and the selected scene is an around 1400×1800 pixel sub-region. The incidence angle varies around 28-30°. We identified three major terrain classes namely *water*, *man-made*, and *vegetation* with the major emphasis on *man-made* terrain type, which may be further divided into *Developed*, *High-* and *Low-density urban* areas depending on their mixture with other natural terrain. The Pauli coded pseudo color image and the ground truth data are given in Figure 20.
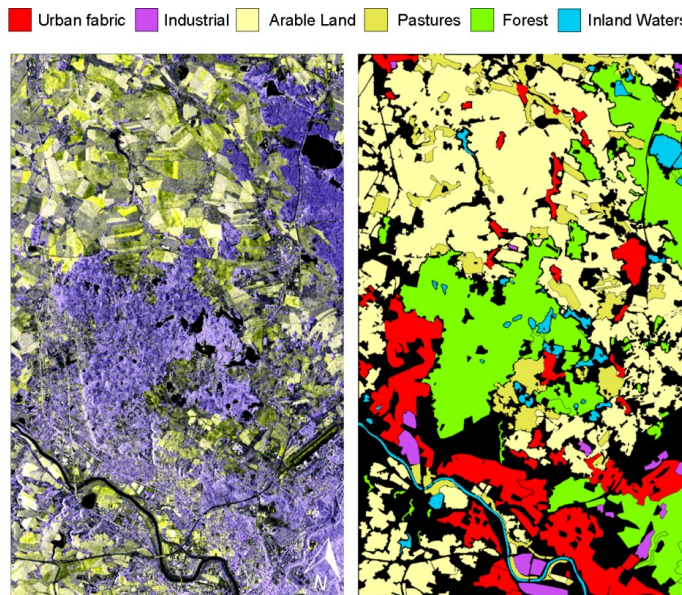
Figure 21 – Dual-pol TerraSAR-X (X-Band) pseudo color image (left) and used ground truth (right). The class legend for the ground truth is given on the top. (© Astrium Services / Info-terra GmbH).

2.5.4. Dresden, TerraSAR-X

The TerraSAR-X data were taken in February 2008 over the area of Dresden in the Southeast of Germany. It was operated in the Strip Map mode dual polarization (VH/VV) and represents radiometrically enhanced multi-look ground range data with reduced speckle, an approximate square resolution of 4×4 meters, and an incidence angle of 41 to 42°. Due to its original size (4419×7154), the data are down scaled by a factor of two to reduce the computation complexity. As a ground truth reference, we have used the Corine Land Cover 2006 initiative [42], which provides a Europe-wide land classification into 44 terrain classes at its finest Level-3 as well as two higher-level categorization legends with 15 and 5 main classes. Due to the different resolution and date of the Corine data, it has been used with the combination of opti-

cal image data as a reference to create the ground truth manually. We applied the 15 class Level-2 legend with the following six classes considered due to their size: *Urban fabric* and *Industrial* as artificial/man-made surfaces, *Arable Land* and *Pastures* as agricultural areas, *Forest,* and *Inland Waters*. A pseudo color image using the VH/VV data (see Section 3.1) and the generated ground truth map are shown in Figure 21.

2.5.5. Po Delta, COSMO-SkyMed-X

The COSMO-SkyMed-X data were taken in September 2007 over the Po Delta region in the Northeast of Italy. It was operated in the Strip Map HImage mode with single polarization (HH) and represents geo-coded (without terrain correction) data with a three-meter resolution. Due to its original size (16716 × 18308), the data are also down scaled by a factor of four to reduce the computational complexity. To create the ground truth, we have applied the same procedure as described for the TerraSAR-X data in the previous Section. From the 15-class Level-2 categorization scheme, the following six major classes are used: *Urban fabric* as artificial/man-made surfaces, *Arable Land* as agricultural areas, *Forest, Maritime Wetlands* as well as *Inland* and *Marine Waters* as water bodies. The HH intensity image and the generated ground truth map are shown in Figure 22.
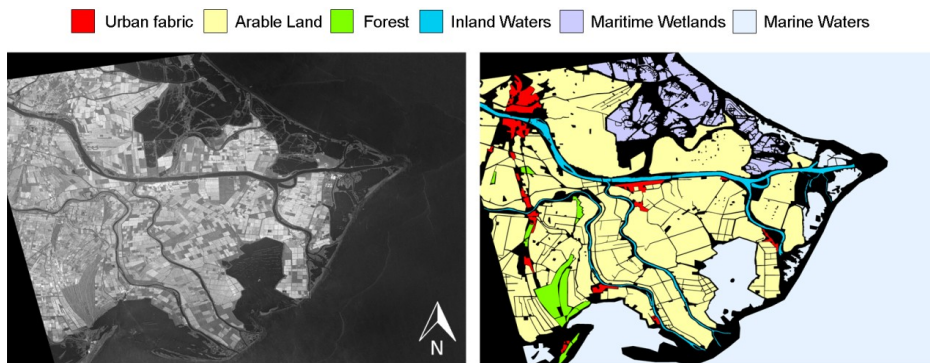
Figure 22 – Single-pol COSMO SkyMed (X-Band) HH intensity image (left) and used ground truth (right). The class legend for the ground truth is given on the top. (© ASI / e-GEOS).

2.6. FEATURE VECTOR FORMATION

This section provides information about the PolSAR and texture features with respect to their feature vector formations used within the experiments and evaluations throughout the remaining chapters. For normalization purposes, all extracted features are linearly scaled into [-1, 1].

The elements of $\langle [T] \rangle$ and $\langle [C] \rangle$ are basic representations of the polarimetric SAR data directly derived from the scattering matrix $[S]$. The Huynen decomposition has been selected, as it was the first approach to decompose $\langle [T] \rangle$ into independent components. Similarly, the components of $H\alpha A$ and the eigenvalue decomposition are chosen as this is the first eigenvector based decomposition and one of the most commonly used target decomposition in PolSAR

image classification. As derived from the eigenvalues, *RVI* is an additional discrimination for vegetation. Regarding other target decomposition components, some of them extend the existing methods by using different coherent scatterings (Touzi decomposition), different physical interpretations of the underlying scattering (Pauli and Krogager decompositions), or they are designed to discriminate particular terrain classes such as vegetation (Van Zyl and Freeman decompositions) or man-made areas (Yamaguchi decomposition). A summary of the employed PolSAR features is listed in Table 1.

Table 1 – Polarimetric SAR feature vector sets used in classification experiments.

Feature set	Feature vector	Dim.	Description
$[T_{11}, T_{22}, T_{33},$ $C_{11}, \|C_{12}\|, \angle C_{12}, \|C_{13}\|, \angle C_{13}, C_{22}, \|C_{23}\|, \angle C_{23}, C_{33}]$	FV_1	12	Elements from $\langle[T]\rangle$ and $\langle[C]\rangle$
$[Span, H, A, \bar{\alpha}, \bar{\beta}, \bar{\delta}, \bar{\gamma}, \lambda_1, \lambda_2, \lambda_3, RVI]$	FV_2	11	Components of $H\alpha A$ and eigenvalue decomposition
$[\|\alpha\|^2, \|\beta\|^2, \|\gamma\|^2, k_s, k_d, k_v, \alpha_s, \Phi_{\alpha s}, \psi, \tau_m]$	FV_3	10	Components from Pauli, Krogager, and Touzi decomposition
$[P_s, P_d, P_v, 2\langle A_0\rangle, \langle B_0\rangle + \langle B\rangle, \langle B_0\rangle - \langle B\rangle]$	FV_4	6	Components from Freeman and Huynen decomposition
$[\lambda_1, \lambda_2, \lambda_3, P_s, P_d, P_v, P_c]$	FV_5	7	Components from Van Zyl and Yamaguchi decomposition

Table 2 – Texture feature vectors used in classification experiments.

Feature abbr.	Feature vector	Dim.	Description
EHD	FV_6	5	Edge Histogram
LBP	FV_7	16	Local Binary Pattern
GW	FV_8	24	Gabor Wavelets
OCM	FV_9	36	Ordinal Co-occurrence Matrix
MLPH	FV_{10}	75	Multilevel Local Pattern Histogram

Individual PolSAR features generally have a single value per image pixel, whereas texture features extract statistics over image regions. To obtain a feature vector per pixel, a surrounding pixel neighborhood is used to define a local image region over the generated intensity images. The feature extraction is performed for every pixel over its ($2w+1$) by ($2w+1$) pixel neighborhood with $w=5$ as a default setting to keep the pixel neighborhood as compact as possible. Section 2.3 describes the five texture features considered throughout this thesis. Particularly, we computed the first and second order moments over three scales and four orientations of Gabor Wavelets and the three ordinal co-occurrence matrices with three distances and four orientations. The MLPH feature is implemented as described by Dai et al. [46] and as

recommended an empirical parameter search was conducted resulting in the following values: $M=5$, $T=2.5$, $K=5$, and $B=3$ over an 11×11 pixel neighborhood for evaluation in Section 3.4. This results in a feature dimension of 75 = (5×3×5) = (M×3×K). For LBP, we quantized the raw 256-bin histogram obtained via the 3×3 neighborhood into a 16-bin histogram whereas MPEG-7 EHD has a fixed five-bin edge histogram. A summary of the extracted texture features is listed in Table 2.

Chapter 3

Color Features in PolSAR Image Classification

There have been various approaches to tackle the challenging problem of PolSAR image classification from using few to many feature combinations, i.e., scattering matrices and target decompositions (TDs) with the addition of texture features. However, among all these alternatives, there is one feature that has been mostly ignored from PolSAR image classification so far: *Color*. For instance, in a survey article [130] covering several techniques for improving classification performance of remote sensing data, no color-based feature has been considered. Yet, for visualization purposes, it is a common practice to generate pseudo color images by mapping polarization matrices or TD components to the different color channels (e.g., the Pauli color-coding in the H,V polarization basis). Even though they do obviously not offer a natural color representation, they may provide useful information for discrimination in terrain classification besides visualization. In particular, several approaches have focused on better color representations of PolSAR images such as assigning same colors to the same scattering information [53], [228], or investigating and comparing different scattering parameters in various color space models for visualization [212]. Furthermore, a contourlet transform based approach has been applied to pseudo coloring using multi-band SAR images [227]. Turner and Woodhouse have recently undertaken a slightly different yet interesting approach where they did not use image pixels but rather icons (in form of ellipses) to represent the image data so that polarimetric properties and geographic context can be visualized together [191].

Similar to texture features, color features are frequently used in the areas of content-based image retrieval [121] and other computer vision areas [71] such as object recognition and image categorization. Yet contrary to texture, color features are usually not considered within the classification of PolSAR images since the original PolSAR data do not indeed provide a (natural) color image representation. Therefore, this chapter draws the focus on investigating the potential of color features over such pseudo color images generated from fully and partially PolSAR data for the purpose of superior performance in supervised land use and land cover (LULC) classification.

The rest of the chapter is organized as follows. It first describes the generation of pseudo color images over the fully and partially polarimetric SAR data in Section 3.1 before Section 3.2 introduces the considered color features, their content information description, and extraction over such pseudo color images. Extensive experiments and evaluations for the application of LULC are reported over fully PolSAR images based on the author's publications [196] and [200] in Section 3.3 and over partially PolSAR images based on the author's publication [195] in Section 3.4. The chapter finishes with some conclusive remarks in Section 3.5.

3.1. DIFFERENT APPROACHES IN PSEUDO COLORING

For best visualization of the original SAR data available in their polarization matrices, they are generally pre-processed to remove the effects of point scatterers with particular strong backscattering. This can be achieved for example by applying a logarithmic approach or simply set any value smaller or larger than a defined lower and upper bound by applying the 1st and 99th percentile [195] respectively. This assures that the point scatterers are eliminated; however, other scatterers backscattering do not change significantly to affect visualization. The pre-processed matrices are then used to generate intensity images over which texture features are usually extracted. Furthermore, these intensity images are also considered to create pseudo color images to extract the desired color features.

For visualization of fully polarimetric data, pseudo color images can be obtained using the H,V polarization basis and assign the polarization matrices HH, VH, and VV directly to the red, green, and blue image components referred to *Sinclair color-coding*. Another approach uses the combinations of |HH+VV|, |HV|, and |HH-VV| to assign them to the red, green, and blue image components, respectively. This mapping produces more human-preferable colors and is commonly referred to as Pauli coding representation [161] or *Pauli color-coding* in the H,V polarization basis, short *Pauli color-coding* (H,V).

Besides using the different polarization matrices, components from polarimetric target decompositions can also be used to generate pseudo color images for visual interpretation or processing. For example, the *Polar color-coding* were the α and H components of the Cloude eigen-decomposition are assigned to the red and green image components with the *Span* used as the blue image component. The *Span* provides the details about the underlying scene terrain whereas α and H are for coloring. Figure 23 illustrates the Sinclair-, Pauli-, and Polar color-coding for the RADARSAT-2 San Francisco Bay image. Note that assigning the TD components to different color channel in case of the Polar color-coding will only affect the visual appearance towards a redder or greener color shade mainly depending on which color channel is assigned to *Span*.

Regarding dual-pol SAR data, the two different polarization matrices can also be used to generate pseudo color imagery directly. There are two main composites of the available polarizations, A and B. They are commonly combined as |A|, |A-B|, |B| or |B|, |B-A|, |A| and the compositions are hereby referred to *RGB1* and *RGB2*, respectively. Figure 24 illustrates com-

positions *RGB1* and *RGB2* for the TerraSAR-X Dresden image with VH as A and VV as B along with the respective intensity image from the *Span*.

Sinclair
(HH, HV, VV)

Pauli
(HH+VV, HV, HH-VV)
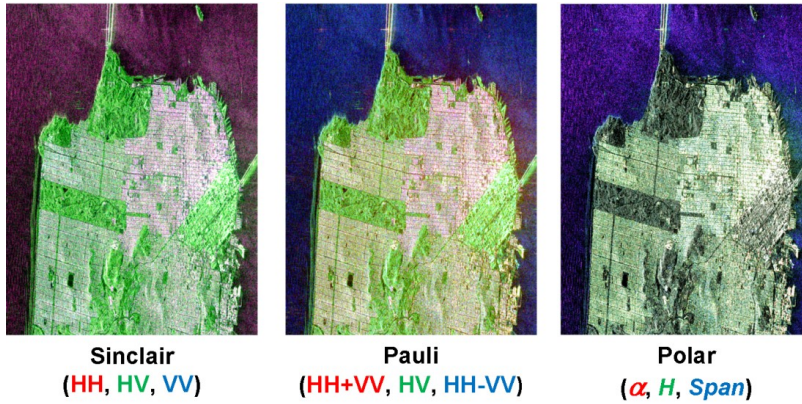
Polar
(α, *H*, *Span*)

Figure 23 – Visualization of pseudo color-coding with Sinclair and Pauli in the H,V polarization basis, and Polar using components of the eigenvalue decomposition.

Span

RGB1
(VH, VH-VV, VV)

RGB2
(VV, VV-VH, VH)
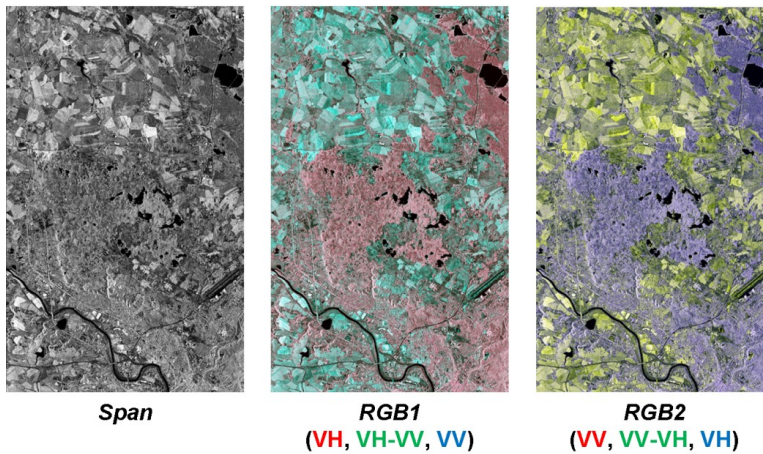
Figure 24 – Visualization of pseudo color-coding for dual-pol TerraSAR-X data (© Astrium Services / Infoterra GmbH).

As for single-pol SAR data, generating pseudo color images similarly as for quad-pol or dual-pol data is not possible since there is only one intensity image from the single polarization mode. Therefore, to form a pseudo color image from single-pol SAR data, the most straightforward way is to either map the intensity values based on a color scale (i.e., via a lookup table) or transform the intensity values into a color space ([73], Chapter 6.3). One of the most commonly applied color scales to colorize gray-scale intensity images is the so-called rainbow or JET map, which reflects the color transition from blue to red. On the other hand, Zhou and Zhang [228] transformed the intensity values of single-pol SAR data into the Hue-Saturation-Intensity (HSI) color space, so generating a pseudo color image for better visualization. In particular, this method tries to maximize the number of just noticeable differ-

ences (JNDs) among colors while maintaining a natural order. It uses the HSI color space as it is a good approximation for the human vision system due the Intensity component being independent of the color information as well as the Hue and Saturation components are in close relation to the perceptive method of humans. Hence, for an 8-bit intensity SAR image, a pixel value is denoted by $g(x,y)$ within the range [0, 255]. The adopted pseudo color coding method is then described as follows:

$$
\begin{aligned}
I &= g(x,y)\ , \\
H &= 2\pi \cdot g(x,y)/255\ , \\
S &= \begin{cases} k \cdot g(x,y) & ,\quad g(x,y) \le 127 \\ k \cdot (255 - g(x,y)), & g(x,y) > 127 \end{cases} ,
\end{aligned}
\tag{18}
$$

with k being a factor for varying Saturation and is empirically set to $k=1.5$ as it produced the visually most pleasing images. For displaying and processing, the generated pseudo color image is transformed into the RGB color space. Figure 25 illustrates the HH intensity image as well as the pseudo color images generated by using JET map and transformation to HSI color space based on [228] for the single-pol COSMO-SkyMed image.
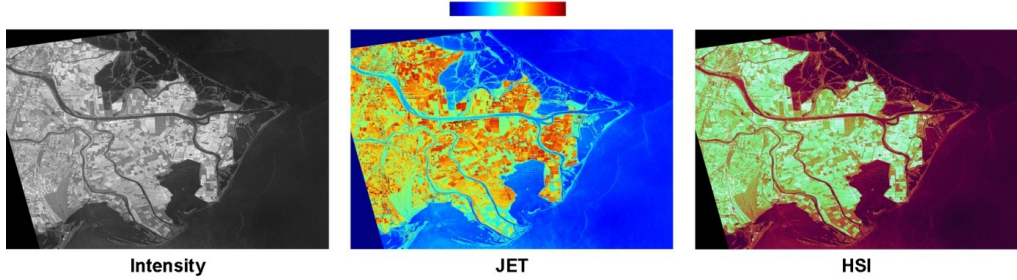
Figure 25 – Visualization of pseudo color-coding for single-pol COSMO-SkyMed data with HSI by Zhou et al. [228] (© ASI / e-GEOS).

3.2. COLOR FEATURE EXTRACTION

As mentioned before color features are commonly used in areas of content-based image retrieval and computer vision. A general set of color features is defined as a part of the MPEG-7 standard [154] formally called Multimedia Content Description Interface. The main idea behind the MPEG-7 is to represent information about the multimedia content (e.g., image, audio, video), but not the content itself (i.e., "bits about bits"). The application of color features follows the same concept as the texture features exploiting statistical properties and relationship within a pixel window over color images. Four color features are considered and detailed next, in particular Hue-Saturation-Value/Intensity (HSV) color histogram, the MPEG-7 Dominant Color Descriptor (DCD), the MPEG-7 Color Structure Descriptor (CSD), and the MPEG-7 Color Layout Descriptor (CLD).

3.2.1. Color Histogram

Color histogram [181] is one of the most basic and widely known features in image processing. Albeit its simplicity, a histogram is powerful in describing major differences and it has been applied over intensity channels of HH, HV, and VV [218] or Span [84] in PolSAR image classification application. Generally, a histogram is a representation of the intensity distribution within an image region usually extracted over each color space component separately. Due to its large amount of information, the different color components are normally quantized to reduce the amount of color levels from millions to thousands or even to just hundreds and perhaps less, so as to make the descriptor representation more compact. We use the HSV color space [134] due to its component partitioning and its similarity to the Human Visual System. The top right hand side of Figure 26 illustrates the content description for the histogram representation for a simple four-color image.
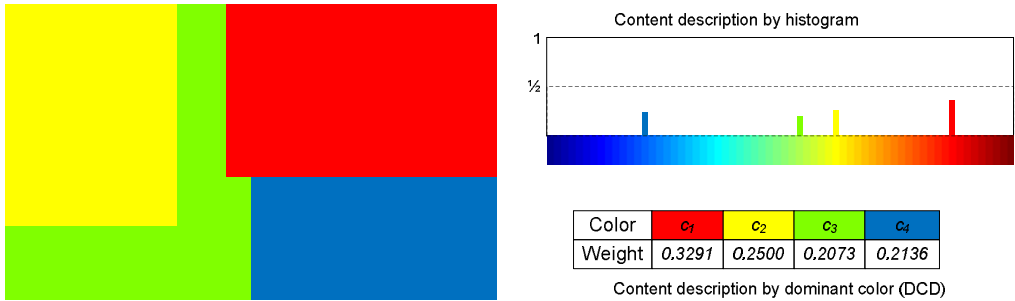
Figure 26 – Accumulation of the Color Histogram and Dominant Color Descriptor.

3.2.2. Dominant Color Descriptor

The MPEG-7 Dominant Color Descriptor (DCD) [134] extracts the most representative colors of a particular color space within an image region where colors are dynamically clustered using the generalized Lloyd algorithm [125] (i.e., by color distortion and area until a certain number of clusters are obtained). Moreover, it is consistent with the Human Visual System as it mainly perceives prominent colors and discards the rest. Due to this fact, it is possible to represent the color content of an image region sufficiently by a few dominant colors. The bottom right hand side of Figure 26 illustrates the content description for the dominant color representation for a simple four-color image.

There are various other approaches to determine the dominate colors in an image region, which provide a better color clustering yet with the burden of larger computational costs as co-authored in [109]. For a general evaluation, the standard MPEG-7 implementation has been chosen due to its simplicity and low computational complexity.

3.2.3. Color Structure Descriptor

Compared to the previous two color descriptors, the MPEG-7 Color Structure Descriptor (CSD) [134] further takes into account the possible relationships among colors with its general idea to describe the local structure in an image region using a sliding 8×8 structure element. CSD is denoted by a histogram where each bin represents the number of structuring elements in the image region containing one or more pixels with a particular color. The operation is illustrated in the Figure 27, where in its current position three colors are present in the structuring element and the CSD histogram is updated accordingly. Commonly the image region is first transformed and quantized into the hue-min-max-difference (HMMD) color space [134] to reduce the number of colors before the structuring element is applied.

Color	Bin Value
c_1	$h(1)$
c_2	$h(2) + 1$
c_3	$h(3)$
c_4	$h(4)$
c_5	$h(5) + 1$
c_6	$h(6)$
c_7	$h(7) + 1$

8x8 structuring element
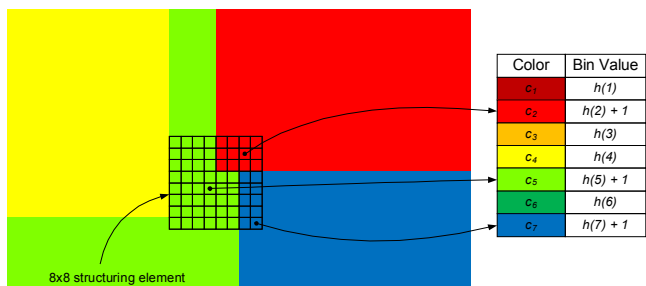
Figure 27 – Accumulation of the Color Structure Histogram.

3.2.4. Color Layout Descriptor

The MPEG-7 Color Layout Descriptor (CLD) [134] is also designed to capture the spatial color distribution in an image region using the YCrCb color space. As illustrated in Figure 28, it has a compact representation consisting of four stages: image partitioning, representative color detection, Discrete Cosine Transform (DCT) application, and nonlinear quantization. In the first stage, an image region is divided into 8×8 blocks, and then a representative color is selected for each block usually by its average color. In the third stage, an 8×8 DCT is applied to the individual color components. A few low-frequency coefficients are selected using zig-zag scanning and quantized to form a CLD, with a default recommendation including six Y coefficients, and three Cr and Cb coefficients each.
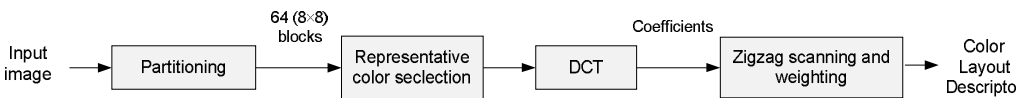
Figure 28 – Extraction process of the Color Layout Descriptor.

3.2.5. Feature Extraction

As for the texture feature extraction, to obtain a description per pixel, the same approach with the surrounding pixel neighborhood is used to define a small image region over the generated pseudo color images in Section 3.1. The feature extraction is then performed for every image

pixel over its ($2w$+1) by ($2w$+1) pixel neighborhood with the same default setting w=5 to keep the pixel neighborhood as compact as possible.

As for the color features themselves, the HSV histogram is quantized into 6×2×2 bins for the Hue, Saturation, and Value component, respectively, the used Dominant Color Descriptor considers the three color components and the weight of the most dominant color to obtain a fixed feature dimension. Regarding the color structure feature (CSD), the final histogram can be quantized into various sizes where we opt for the smallest feature dimension of 32 due to the rather small image region the feature is extracted from, and the color layout descriptor is fixed to a dimension of 12 as described in Section 3.2.4. A summary of the previously described color features is listed in Table 3, as employed in the different experiments.

Table 3 – Color feature vector combinations used in classification experiments.

Feature abbr.	Feature vector	Dim.	Description
DCD	FV_{11}	4	Dominant Color Descriptor
HSV	FV_{12}	24	Hue, Saturation, Value histogram
CSD	FV_{13}	32	Color Structure Descriptor
CLD	FV_{14}	12	Color Layout Descriptor

3.3. COLOR OVER FULLY POLARIMETRIC SAR DATA

This evaluation focuses on investigating the potential of the color features as described in the previous Section 3.2 for their performance in supervised fully PolSAR image classification. The main objective here is to extract these powerful color features from one pseudo color-coded image, in particular the Pauli color-coding in the H,V polarization basis, and to evaluate their discrimination power with several combinations of PolSAR and texture features. Extensive experiments are provided over three fully PolSAR images from an airborne (AIR-SAR) and a space-borne (RADARSAT-2) system taking into account two different band spectra, C and L. Two well-known classifiers, Support Vector Machines [43] and Random Forest [19] are employed, both of which are becoming more popular in this field, to test and evaluate the role of color features over the classification performance.

The rest of this evaluation is organized as follows. The experimental setup introduces the classifier topologies, the different PolSAR image data, the polarimetric SAR-, and image processing features. After that, classification results and evaluations are provided over the different combinations of PolSAR, texture, and color features. At the end, different pseudo color-coding methods for fully PolSAR images are investigated with their effect on classification performances over the texture and color features.

3.3.1. Classifiers

In the past, various machine-learning approaches have been utilized for SAR image classification. During the last two decades, one of the most commonly used approaches is the Maxi-

mum Likelihood Classifier and its variants, e.g., [2], [57], [86], [110], [119], [128], [170], [178], and [204]. However, in recent years the focus is drawn to the applications of Support Vector Machines (SVM) [43] and Random Forest (RF) [19].

SVM is one of the most commonly used classifiers when it comes to classification of SAR data and is widely applied to various remote sensing applications [143]. It has been recently becoming even more popular for numerous SAR image classification applications, e.g., [31], [46], [175], [220], [226], [232]. Since SVM is a binary classifier by nature, there are two main topology approaches to adapt it to a multiclass classification problem: pairwise classification [111], also known as the One-versus-One (OvO) approach, and the One-versus-All (OvA) method [91]. Further, tree classifiers including RF are also frequently applied to SAR image classification problems as in [72], [162], [176], [220], and [232]. Similar to the SVM, RF is also a binary classifier, which makes it necessary to use either an OvO or OvA topology, to perform multiclass classification.

Table 4 – The four classifier combinations employed within the fully polarimetric SAR image color feature evaluation.

Classifier abbreviation	Description
OvA_RF	One-versus-All using RF
OvA_SVM	One-versus-All using SVM
OvO_RF	One-versus-One using RF
OvO_SVM	One-versus-One using SVM

These experiments focus mainly on SVM and RF due to their recent popularity applying the two topologies, OvO and OvA. Since each topology has its advantages and shortcomings, both are considered for individual classification performance analysis. As a result, Table 4 lists the four classifier combinations used within the experiments. In order to determine the best classifier configuration (i.e., kernel and kernel parameters), a sequential search over a range of configurations is performed during the training stage with respect to a training validation set and the best configuration found will be used for classification and performance evaluations.

For the SVM implementation, we employed the libSVM library [27] and in order to find out the best SVM configuration (kernel and kernel parameters), an exhaustive search is conducted for the best kernel type among the linear (*LIN*), polynomial (*POL*), radial basis function (*RBF*) and sigmoid kernel (*SIG*). In addition, the individual parameters are optimized as well, i.e., the respectable penalty parameter, C (2^n; n=0,...,3) and parameter γ (2^{-n}; n=0,...,3), if applicable to the kernel type (e.g., for *POL* and *RBF*). Even though this will merely provide a coarse parameter search, it is definitely better than just keeping the default settings $C=2^0$ and $\gamma=0.01$. Overall, 40 different kernel and parameter configurations are checked for the best possible configuration, in particular 4 *LIN* for C, 16 *POL* for C and γ, 16 *RBF* for C and γ, and 4 *SIG* for C.

As for the RF classifier, we employed Breiman's implementation [19], where the best number of trees within the forest is searched from 50 to 100 in steps of 10. In each tree, a splitting decision is made per node based on a randomly sampled number of input features. This number is commonly determined by the square root of the total input features, $\sqrt{N_{in}}$. A further step is added in-between $\sqrt{N_{in}}$ and all features N_{in}, $\frac{1}{2}(\sqrt{N_{in}} + N_{in})$, which is slightly larger than half the size of N_{in}. Thus, for each forest size, the random split in each tree is optimized between $\sqrt{N_{in}}$ and $\frac{1}{2}(\sqrt{N_{in}} + N_{in})$.

It is worth mentioning here that all configurations are enumerated with a hash function, which ranks them with respect to their complexity, e.g., associating higher hash indices to an individual RF with a higher complexity. This means that while sequentially searching for the best parameters / configurations for any classifier type, it will always favor less complex configurations as long as they realize one of the lowest validation errors during training. Note that it is not the main goal to tweak and achieve the best possible classification accuracies for each individual classifier - rather to illustrate the effects of additional features and input sizes of the different classifiers.

During the experiments, the classification tasks conducted over the selected PolSAR images are considered as *uni-class* where one sample can only belong to one particular class, so that during the training phase each positive sample of an individual class can be used as a negative sample for all others. Yet if there are numerous classes, an unbalanced numbers of positive and negative samples per class may cause a bias problem in OvA. To address this problem of handling unbalanced training data, the concept of positive-to-negative ratio (PNR) as described in Section 4.2.2.4 is applied in this evaluation using PNR equal to 10 for all classifiers.

3.3.2. Polarimetric SAR Images

This evaluation uses the three fully PolSAR images as described in Section 2.5 namely Flevoland, AIRSAR, L-Band, Flevoland, RADARSAT-2, C-Band, and San Francisco, RADARSAT-2, C-Band. With this setup, it can be demonstrated how effective the additional color features are over a variety of polarimetric SAR images in terms of the system (AIRSAR, RADARSAT-2), the operative band (C, L), and the underlying classification problem (e.g., number of classes, main terrain types). Table 5 provides a summary of the used fully polarimetric image data for this setup with their different locations, sensors, and bands.

With the available ground truth, the following procedure is adapted to generate the training and testing sets for each PolSAR image. The training sets contain approximately 100 pixels per class for the lower resolution AIRSAR image and approximately 500 pixels per class for the larger RADARSAT-2 images. The testing sets have either 50000 pixels per class or the entire class ground truth if its size is smaller than 50000. Main reason for this is to keep a training-to-testing ratio between 1:100 and 1:120 for all PolSAR images and to assure that the training and testing sets are of similar sizes for the individual PolSAR images. Furthermore,

the training data were split into 50 percent for training and 50 percent for validation of the classifier parameters; and Table 6 summaries the training and testing data per PolSAR image as well as the abbreviations used for the remainder of this chapter.

All PolSAR images are speckle filtered using the suggested filter by Lee [117] with a 5×5 window before any feature is extracted. Many methods have been considered for speckle reduction [226], [228] with emphasize on detail preservation such as polarimetric signatures, texture, image structures, and prevention of loss in spatial resolution [228]. The speckle filter by Lee [109] might not be the best compared to the state-of-the-art in polarimetric SAR speckle filtering. However, it has been widely used and shown good performance regarding speckle reduction and detail preservation [118], [47]. This will have an influence on classification [26], [109] such as homogenous regions due to the speckle reduction and forest or urban areas due to the filter's capability of preserving texture and line structures. As speckle filtering is outside the thesis scope, the filter by Lee [117] is considered as the main speckle reduction method whenever applied to SAR image data in this thesis.

Table 5 – Overview of image data used within the fully polarimetric SAR image color feature evaluation.

Name	System	Band	Date	Dimensions	Incident angle
Flevoland	AIRSAR	L	Aug 1989	1024×750	20-44°
Flevoland	RADARSAT-2	C	Apr 2008	1639×2393	30°
San Francisco	RADARSAT-2	C	Apr 2008	1426×1876	30°

Table 6 – Fully polarimetric SAR image data with their train and test sets for color feature evaluation.

Name	No. classes	Train size	Test size	Ratio Test/Train	Abbr.
Flevoland	15	1793	208186	116	*Flevo_L*
Flevoland	4	2000	200000	100	*Flevo_C*
San Francisco	5	2500	250000	100	*SFBay_C*

3.3.3. Feature Sets

The PolSAR features from Section 2.6, namely FV_1 - FV_5, are combined into six different PolSAR feature sets (FS) as listed in Table 7. They include basic scattering information in form of covariance and coherency matrices as FS_1, commonly used components of $H\alpha A$ and eigenvalue decomposition as FS_2, and several coherent and incoherent target decompositions as FS_4, as well as combinations of these. Additionally, each PolSAR feature set is also combined with three sets of visual features based on the texture features from Section 2.6, namely FV_6 to FV_9, and the color features from Section 3.2.5, namely FV_{11} to FV_{13}, extracted over the pseudo color image generated by Pauli color-coding (H,V). These features are further joined to form a set of color features ($FV_{11}+FV_{12}+FV_{13}$), C, and a set of textural features

(FV_6+FV_7+FV_8+FV_9), T, see bottom of Table 7, so that each FS_x will be augmented by color (FS_x^C), texture (FS_x^T), and color + texture (FS_x^{CT}). Thus, this will result in 24 feature set combinations with various sizes ranging from 11 dimensions for the smallest ($|FS_2|$) and 187 dimensions for the largest ($|FS_6^{CT}|$). The symbol FS_x^z is introduced to abbreviate different groups of feature set combinations such as FS_x^C for the color combinations of all six different PolSAR feature sets and FS_6^z when referring to all combinations and extensions of the PolSAR feature set FS_6. With this setup of feature set combinations, we can investigate the influence of color and texture as an addition to different selections of PolSAR features.

Regarding the feature dimensions, note that when the number of (representative) training samples is relatively small with respect to the number of features, the well-known problem of the "curse of dimensionality" (i.e., the Hughes phenomenon) occurs. This results in the risk of overfitting of the training data and can lead to poor generalization capabilities of the classifier. Generally, neither RF nor SVM topologies have a serious "curse of dimensionality" problem for features of few hundred dimensions. Particularly SVMs with RBF kernel can theoretically learn in infinite dimensions providing that the training dataset size is infinite too. Under the light of this, the investigation on "curse of dimensionality" over the classification performance is beyond the scope of this evaluation, as there are excellent research works [35] in the literature.

Table 7 – Combinations of PolSAR and image processing feature vectors used in the fully polarimetric SAR image color feature classification experiments.

Feature set	Feature vector	Dim.	Description
FS_1	FV_1	12	Elements from $\langle [T] \rangle$ and $\langle [C] \rangle$
FS_2	FV_2	11	Components of $H\alpha A$ and eigenvalue decomposition, RVI
FS_3	$FV_1 + FV_2$	23	Combination of FS_1 and FS_2
FS_4	$FV_3 + FV_4 + FV_5$	23	Components from target decompositions (Pauli, Krogager, Touzi, Freeman, Huynen, Van Zyl, and Yamaguchi)
FS_5	$FV_2 + FV_3$ $+ FV_4 + FV_5$	34	Combination of FS_2 and FS_4
FS_6	$FV_1 + FV_2 + FV_3$ $+ FV_4 + FV_5$	46	Combination of FS_3 and FS_4
C	$FV_{11} + FV_{12} + FV_{13}$	60	Extracted color features
T	$FV_6 + FV_7$ $+ FV_8 + FV_9$	81	Extracted texture features

3.3.4. Initial Experimental Results

This section will first cover general observations related to the six PolSAR feature sets and the three PolSAR images before going in depth analysis for each of the individual classification experiments performed.

All the presented results for a different classifier, feature set, and image combination, are average values obtained over 12 runs per combination applying the trimmed mean leaving out the minimum and maximum values to reduce the effects of "extreme" outliers. Firstly, the average classification accuracies for the six different PolSAR feature sets (and their respected texture and color combinations) over all classifiers and images are shown in Figure 29 to provide an assessment of utilizing texture and color features in combination with the different PolSAR features. As anticipated, the minimal FS_1 benefits highly from texture and color information with large gains of 6% and 15% because of the small discrimination power with just $\langle [C] \rangle$ and $\langle [T] \rangle$ elements. This can also be seen for all the other feature sets FS_2 to FS_6, where with the application of more discriminative PolSAR features in form of target decompositions, the classification accuracies can be increased on the average by around 10 point, from 77 to 87 percent. Texture features help to improve by an extra one percent on the average and the color features can boost that to an additional 3% to 5% on the average, to over 90% compared to the PolSAR features. It can be observed that the presence of texture features in FS_x^{CT} have a marginal influence on the classification accuracies compared the color features alone, FS_x^C. Anyhow, we can also see that overall differences among the feature combinations for FS_2^z, FS_4^z, FS_5^z, and FS_6^z are not significant, which probably demonstrates why the Cloude-Pottier decomposition (FS_2) has been used so extensively. The drop for FS_3 compared to FS_2 comes from the addition of FS_1 and its rather low classification performance. The features of FS_2 are not able to compensate this negative effect.
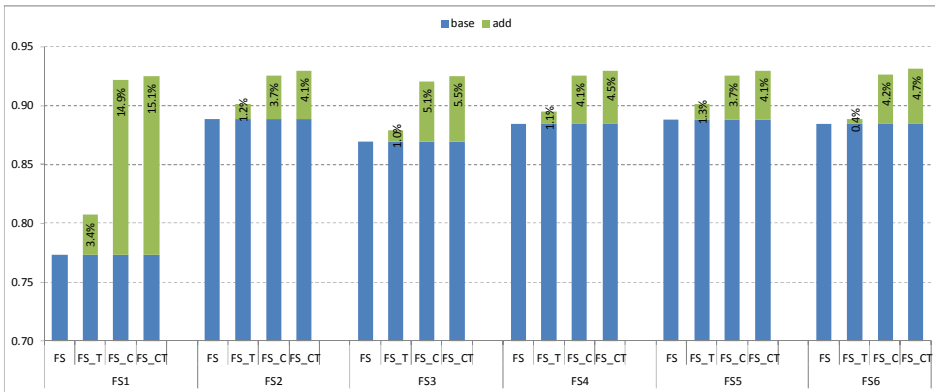
Figure 29 – Average classification accuracies for the six different feature sets over all classifiers and datasets for the fully polarimetric SAR image color feature experiments. (Blue describes the base line accuracies of the PolSAR FS and green is the improvement with the additional texture and/or color features).

Secondly, the average classification accuracy for the three different PolSAR images over all classifiers and feature sets are presented in Figure 30. It can be observed that for all three images used in these experiments, color features provide an increase in the classification accuracy from around 4% up to 9%.

Considering *Flevo_L*, the classification accuracy drops (almost -3%) with the addition of the textural information to FS_x and the differences between FS_x^C and FS_x^{CT} are marginal. The reason for this is that the texture information has a rather minimal effect on the classification accuracies probably due to the lower resolution of the data and the majority of classes are agriculture fields with no individual prominent texture features.

For the *Flevo_C* image, the texture features provide some contribution to the classification accuracy and we will see later in the more detailed analysis for this image that primarily the *Urban* class accounts for this gain, whereas with the color features a gain of 2.5% - 5.5% compared to FS_x^T / FS_x can be achieved. Note that the classification results with the combination of color and texture is pretty much the same as the one with color alone.
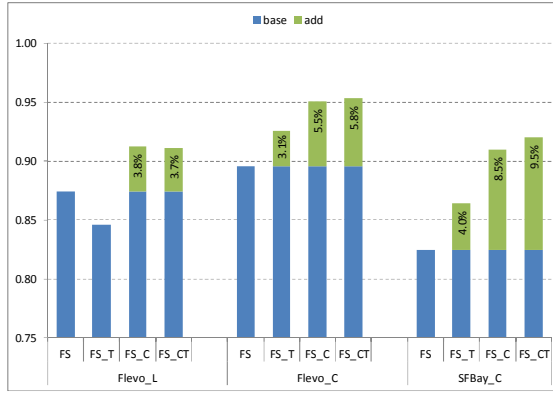
Figure 30 – Average classification accuracies for the three different datasets over all classifiers and feature sets for the fully polarimetric SAR image color feature experiments. (Blue describes the base line accuracies of the PolSAR FS and green is the improvement with the additional texture and/or color features).

As for the third image, *SFBay_C*, the addition of texture features provides a contribution of about four percent to the overall classification accuracy. On the other hand, the addition of the color features increases the accuracy by 4.5% / 8.5% compared to FS_x^T / FS_x. Thus, whenever the color features are utilized, the contribution of the additional texture features becomes in general less significant, only a one percent improvement for all tested images.

Based on these observations, we will now have a deeper analysis for each of the individual PolSAR images regarding the different feature sets and how they affect individual classes. The classification accuracies for the images and their classes are visualized via box plots. We have chosen them as an effective way for representing graphically a large amount of numerical data. They depict five values, namely the least observation (in this case, it is the minimum accuracy), lower quartile (25%), median, upper quartile (75%), and the highest observation

(maximum accuracy). As observed in the previous results (Figure 29), all feature sets besides FS_1^z show similar behavior and tendencies with the additional color and texture features. Therefore, all box plots for an individual image are generated over all classifier runs for a particular feature set (e.g., FS_2^T to FS_6^T) applying some basic outlier removal for the extreme cases where a particular classifier run performs significantly different from the others. This mainly removed outliers from the bottom end of the accuracy range.

The main observation for the *Flevo_L* image in Figure 31 is the loss of accuracy for the FS_x^T feature sets for individual classifiers. The performance of SVM seems to be varying with respect to the multiclass topology, i.e., the OvA_SVM shows heavy drops for each FS_x^T; and the performance of OvO_SVM drops to or below the median and being among the lowest accuracies for the majority of FS_x^T. It can be observed that three out of four classifiers performances are severely affected by texture especially for FS_1^T, FS_3^T, and FS_6^T, which include FV_1 (i.e., $\langle [C] \rangle$ and $\langle [T] \rangle$ elements) and cannot provide good discrimination in this classification task. This is probably because SVM work on the entire feature set and, obviously, the texture has quite a negative effect particularly on the large number of classes (15) in this image. Note that OvA_RF is the most invariant to FS_x^T as RF employs random feature selection within each ensemble decision tree, hence, not suffering from the lower discrimination of FV_1 and the texture features. However, we can observe that RF is quite affected by the OvO multiclass topology due to its pairwise classification. In this case, OvA_RF performs significantly better than its OvO counterpart as well as the two SVM classifiers.
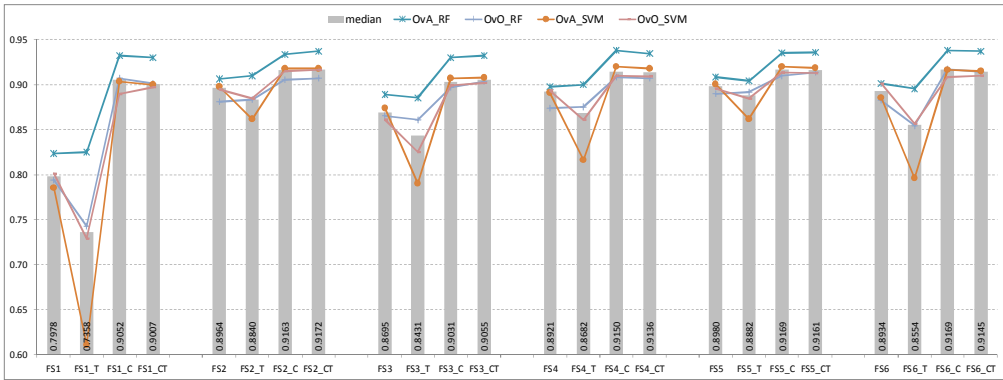
Figure 31 – Average classification accuracies for the four classifier topologies per feature set over *Flevo_L* image in the color feature experiments (gray bar represents the median value).

The effect of the texture and color features can be closely investigated in Figure 32, which displays the results for the single and combined texture and color features per class for *Flevo_L*. We can observe that overall the texture features perform rather poorly, where *Water, Forest, Potatoes,* and *Building* are exceptions in the sense that texture features obtain results three to four times better than the other classes. Furthermore, we see that the single GW and OCM feature achieved best results and are the main contributor to the combined texture feature for these four classes. For the other classes, texture feature discrimination does virtually

not exist, so that they cannot provide any significant additional discrimination when combining with the extracted PolSAR features in this *Flevo_L* classification task. Regarding the color features, results look more positive with single and combined color features able to obtain better results and in this particular case outperform their texture counterparts, which is also reflected in the results when combined with the PolSAR features.
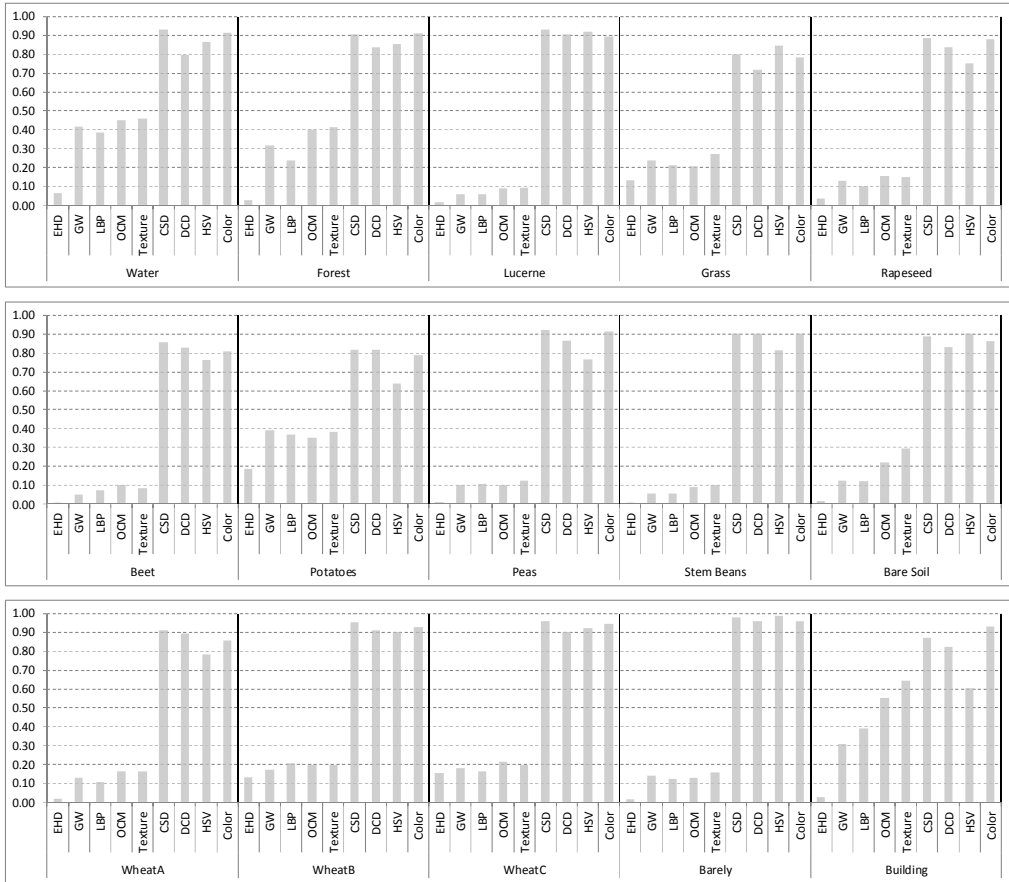
Figure 32 – Classification accuracies for the *Flevo_L* classes over the single and combined texture and color features.

Overall, the introduction of the texture features has either an insignificant or rather negative effect, which is also visible in the box plots for FS_x^T in Figure 33. The lower quartile (25%) and upper quartile (75%) rectangles are far larger indicating a higher variation in accuracies compared to the more compact representations for the other feature sets (also see Figure 33). As discussed earlier, this is probably due to the larger number of classes, majority of which being agriculture fields, thus no significant texture difference exists among them. Furthermore, such a low-resolution SAR image may not allow for major textural discriminations and moreover, the applied speckle filtering may further degrade the texture information. Therefore, such a degraded and somewhat similar texture information among most of the

classes rather introduce "confusion" and makes it difficult to learn individual class patterns when the majority of features are quite similar and noisy for many classes as seen in Figure 32. Recall that the texture features have an overall dimension of 81 compared to the PolSAR features having dimensions between 10 and 40 for different FS_x^T; hence, texture will have a rather dominating influence. RF can better cope with this effect than SVM as a few features are randomly selected within each tree in the forest, which in turn, will negate the influence of the large texture feature dimension. Yet, the differences for FS_x^C and FS_x^{CT} are not as significant as for FS_x^T besides some variations with respect to whisker ranges. This means that the color features carry the main discrimination information and can cancel out negative effects from the texture features.
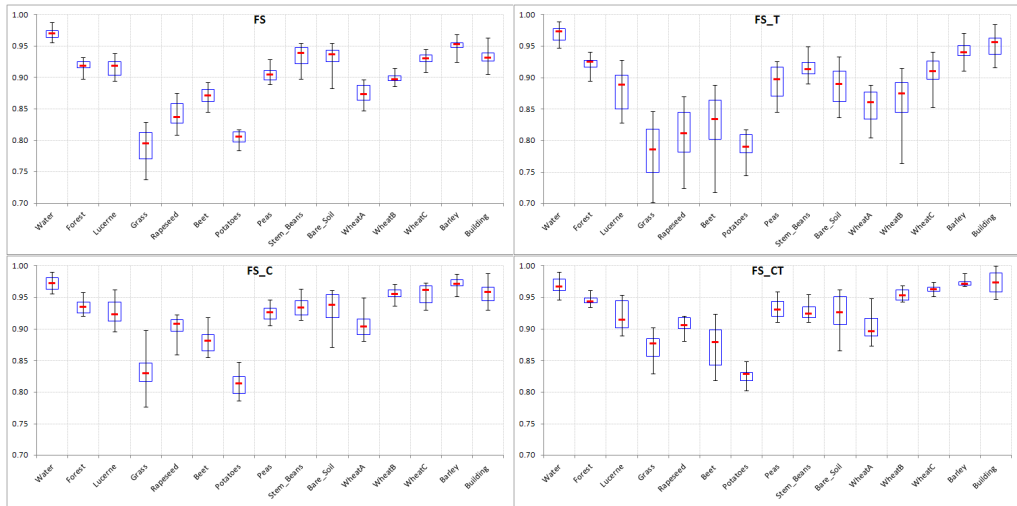
Figure 33 – Classification accuracies for the *Flevo_L* classes with FS_x (left), FS_x^T (right) in top row and FS_x^C (left), FS_x^{CT} (right) in bottom row for the color feature experiments. The box plots show 25 and 75 percentile as the blue rectangle, the red line indicates the median, and the whiskers represent the min. and max. values, respectively.
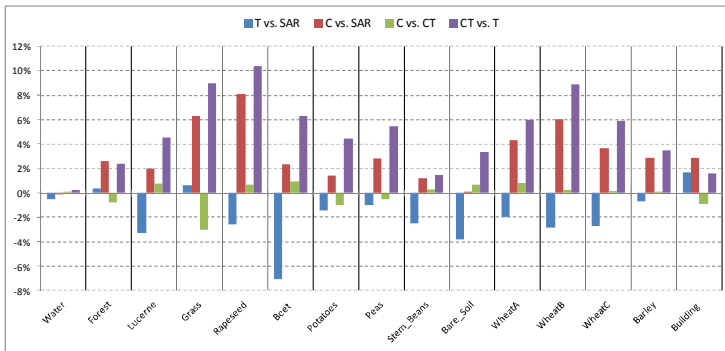
Figure 34 – Differences in classification accuracies comparing the influence of texture and color features per class in the *Flevo_L* image.

A better visualization of the feature differences is illustrated in Figure 34, which shows classification accuracy gains/losses per terrain class. Only for the *Building* class, texture makes a noticeable yet marginal positive impact (1%). For the other 14 classes, 12 of them suffer in accuracy with the addition of texture features due to the absence of strong textural pattern to provide significant discrimination. For the addition of color features, two classes gain around one percent accuracy and the majority of eight classes gain between 2% and 4%. The Highest impact is observed for classes *Grass*, *Rapeseed,* and *Wheat B*, which gain more than 6% in accuracy compared to just PolSAR features (FS_x).
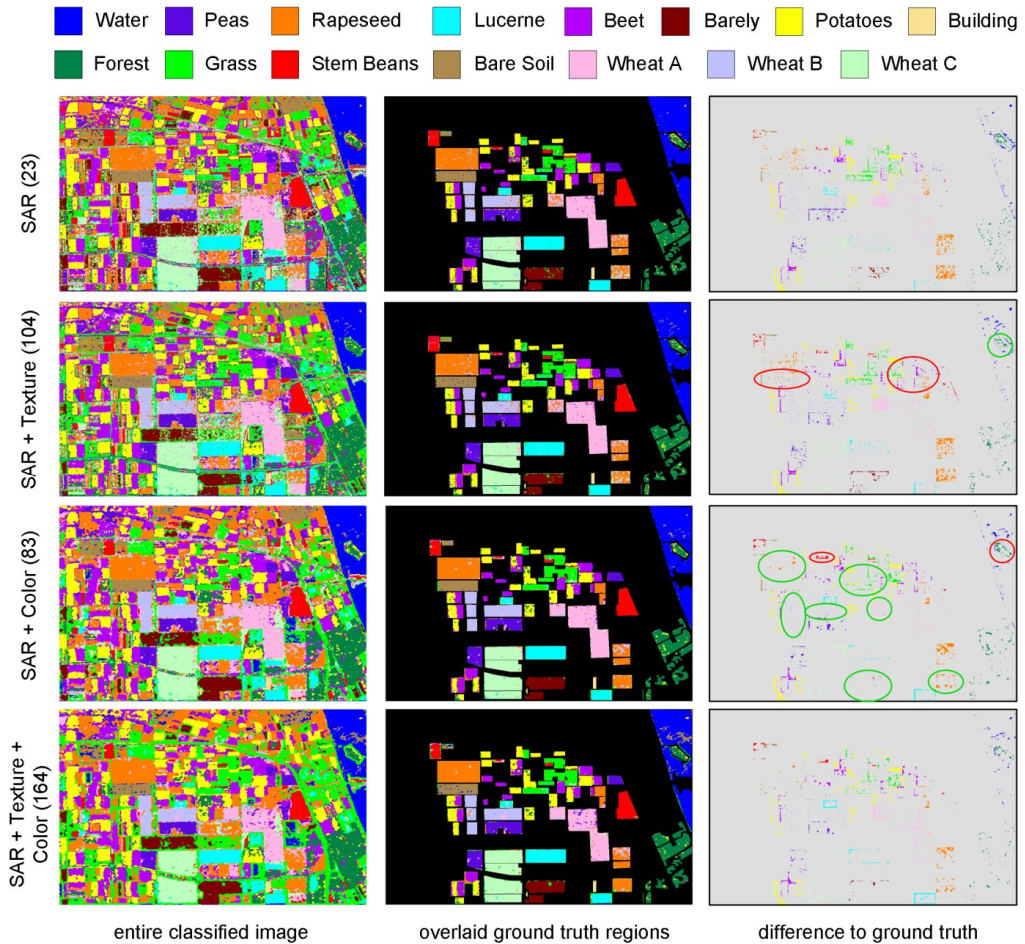
Figure 35 – Classification results for the *Flevo_L* image for FS_3 over OvO_SVM. Left column is the classified PolSAR image, the middle column is the classified image overlaid with the ground truth, and the right column shows the difference to the ground truth with the color indicating what the ground truth should have been. Circles mark degradation (red) and improvement (green). First row shows results for FS_3, the second row FS_3^T, the third row FS_3^C, and the last row FS_3^{CT}.

Figure 35 presents visual classification results for the *Flevo*_L image showing the differences of just PolSAR features (FS₃) versus its augmented feature sets with texture, color, and texture + color, respectively, for the OvO_SVM classifier. For the FS_3^T case, the larger drops for *Beet* and *Bare_Soil* class are visible, whereas for the classes with drops of two percent or lower, differences are not as perceptible. It is noticeable that the *forest* area on the small island is slightly better classified as for FS₃, particular around the borders. As for FS_3^C, visual differences to FS₃ and FS_3^T are apparent in particular for the classes, which have shown larger numerical improvements in Figure 34.
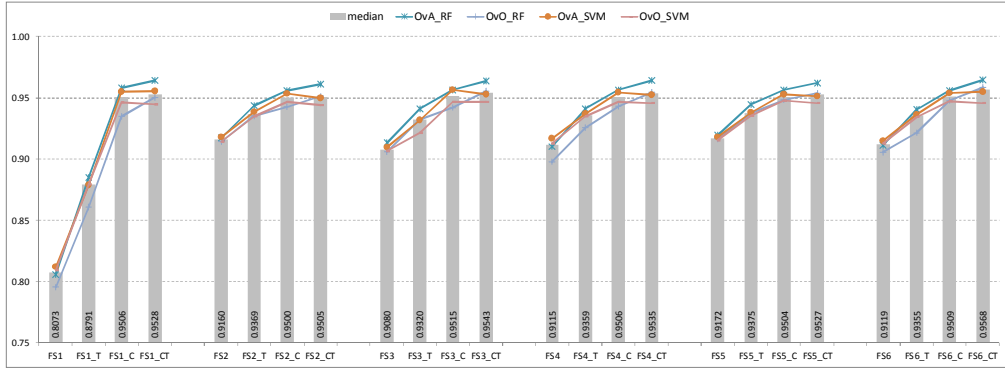
Figure 36 – Average classification accuracies for the four classifiers per feature set over *Flevo_C* image in the color feature experiments (gray bar represents the median value).

Compared to the *Flevo_L* image, it can be noticed for the average classification accuracies of the *Flevo_C* image in Figure 36 that texture features have a more positive impact, as no severe drops exist for FS_x^T and they rather improve the classification accuracies compared to FS_x. Yet, the color features in FS_x^C are able to provide a better discrimination to improve classification results further. With the combination of texture and color features, it can be noted that both RF classifiers can actually increase their classification accuracies from FS_x^C to FS_x^{CT} whereas SVM either drop (FS_2^{CT} to FS_5^{CT}) or stay around level (FS_1^{CT}, FS_6^{CT}) with their respective FS_x^C results.

As observed in Figure 36, texture features provide additional discrimination power to improve classification accuracies with particularly *Water* and *Urban* obtaining good results with texture features alone as well as for *Forest* even though not quite as good as for the previous two terrain classes as illustrated in Figure 37. On the contrary, the *Cropland* class does not seem to carry any dominant texture patterns; however, the color features seem to provide a better discrimination for this terrain class.

From the box plot in Figure 38, one can notice that the different box plot shapes and values are close together and the minimum and maximum values are not too far off the main quartile body, thus there is not too much variance within the different classifiers. When investigating the effect of particular feature sets for individual classes, it is revealed that the increase of FS_x^T accuracy in Figure 36 appears primarily from the *Urban* class with an improvement around

9.5% and this is actually the only one that significantly gains from the texture features due to the unique patterns of roads and buildings. This can also be seen when visually inspecting the image where the majority of classes do not really have a strong texture pattern besides the *Urban* class. Even though reasonable results with texture features can also be obtained for *Forest* (60% as seen in Figure 37); they do not seem to add as much additional discrimination to the FS_x. The same observation can be made for *Cropland* over the color features with FS_x^C similar to FS_x^T and marginal improvements for FS_x, which already achieves high classification accuracy. *Forest* and *Urban* are the classes that highly benefit from the additional color features with 5% and 10% accuracy gains compared to PolSAR features only, where FS_x^C and FS_x^T perform equally for the *Urban* class. Note further that additional color and texture features have no effect on the classification accuracy of *Water*. The PolSAR features already provide the highest possible discrimination against the other classes due to its dominant surface scattering.
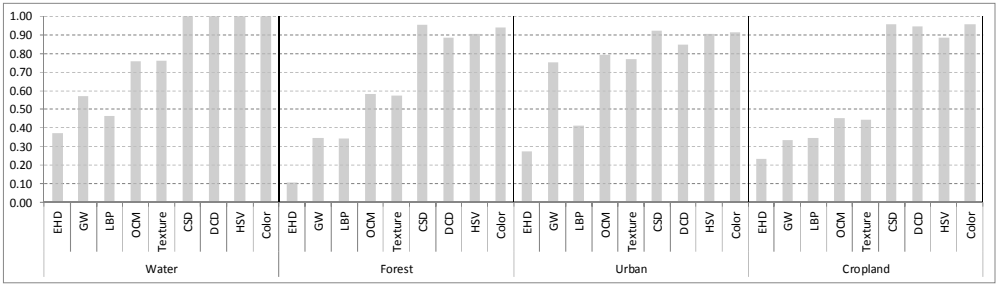
Figure 37 – Classification accuracies for the *Flevo_C* classes over the single and combined texture and color features.
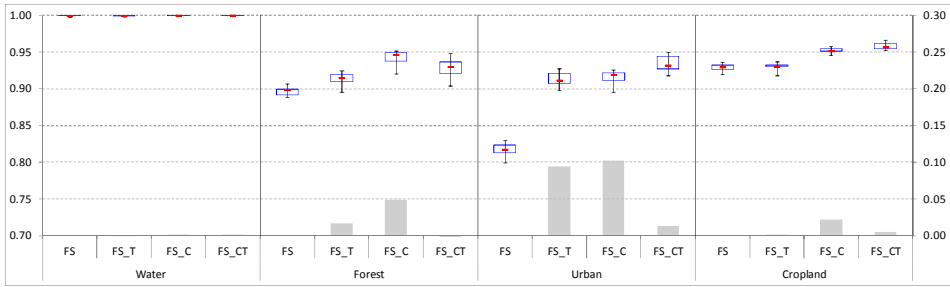
Figure 38 – Classification accuracies for the *Flevo_C* classes and differences in classification accuracy comparing the influence of texture and color features per class in the color feature experiments. The box plots show 25 and 75 percentile as the blue rectangle, the red line indicates the median, and whiskers representing the min. and max. values, respectively, with accuracies on the primary axis on the left. The gray bars indicate the feature accuracy differences with the secondary axis on the right.
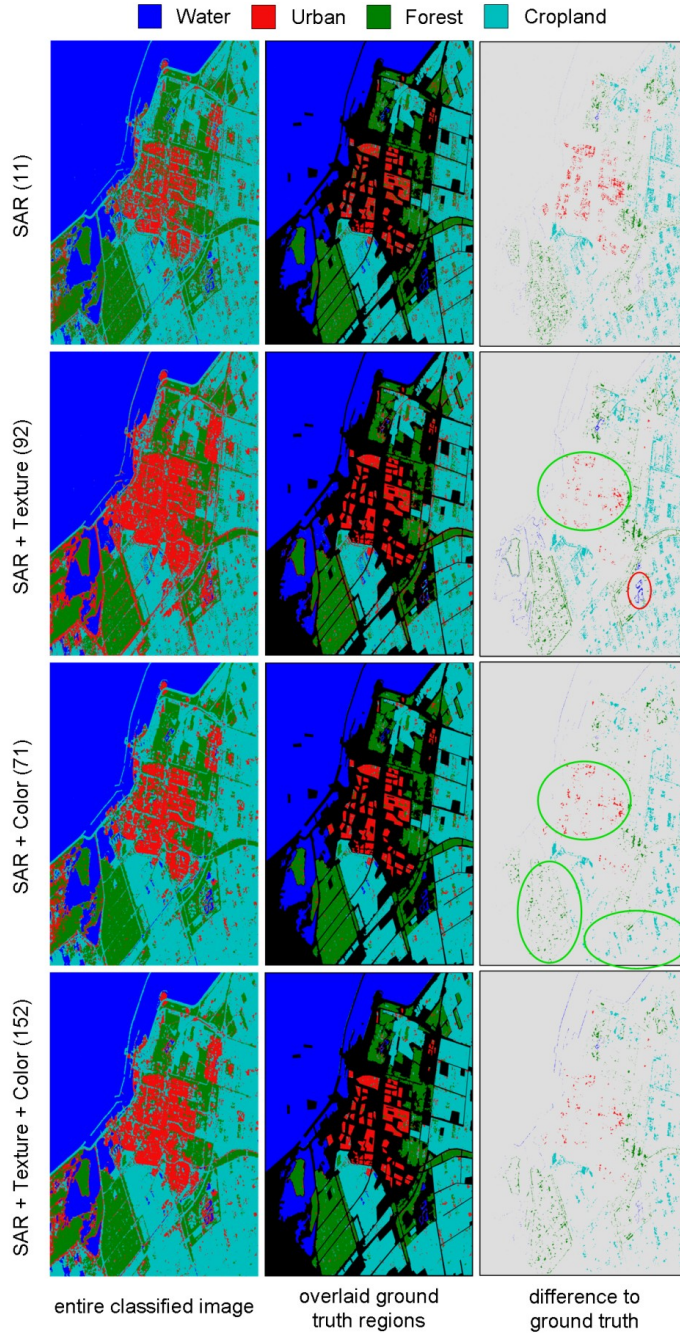
Figure 39 – Classification results for the *Flevo_C* image for FS_2 over OvA_RF. The left column is the classified PolSAR image, the middle column is the classified image overlaid with the ground truth, and the right column shows the difference to the ground truth with the color indicating what it should have been. Circles mark degradation (red) and improvement (green). First row shows results for FS_2, second row FS_2^T, third row FS_2^C, and last row FS_2^{CT}.
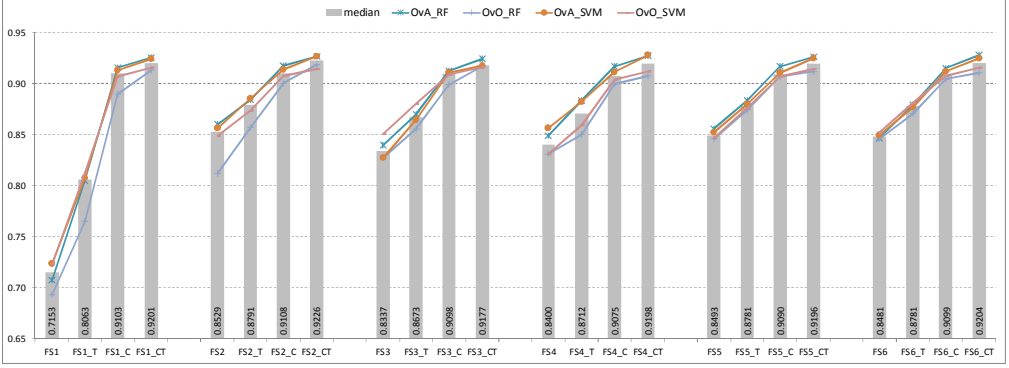
Figure 40 – Average classification accuracies for the four classifiers per feature set over *SFBay_C* image in the color feature experiments (gray bar represents the median value).

Figure 39 presents visual classification results for the *Flevo_C* image showing the differences of PolSAR features alone (FS$_2$) versus its augmented feature sets for the OvA_RF classifier with texture, color, and texture + color, respectively. For the FS$_2^T$ case, the numerical improvements in the *Urban* areas are also noticeable visually. However, in the classified image it tends to overestimate the *Urban* area as well as misclassifies edges between *Forest* and *Water* or other flat areas most probably due to feature extraction over the pixel neighborhood. It also misclassifies some water regions as visible in the bottom right part. On the contrary, using FS$_2^C$ provides correct classification for those water regions and does not suffer from the *Forest*/*Water* borders. The improvements of the numerical classification performance for FS$_2^C$ are visible, too. Moreover, for this feature set *Forest* and *Cropland* areas in the bottom left and right of the image are classified closer to the ground truth than for FS$_2$ and FS$_2^T$.

Over the results for the *SFBay_C* data, it can be noticed that the behavior of the classifiers, as illustrated in Figure 40, is different compared to the two previous images. The texture features provide good discrimination improving the FS$_x$ accuracies by 2-3% compared to FS$_{2\text{-}6}$ and 9% compared to FS$_1$, whereas the color features are able to provide an additional improvement of 3-4% and 10% compared to FS$_2^T$ to FS$_6^T$ and FS$_1^T$, respectively. Moreover, for this particular image, the texture features are able to provide, though small, additional discrimination when combined with the color features. These tendencies are visible for all classifiers as observed distinctively for the two Flevoland images.

The OvA versions of RF and SVM perform at the top level; however, the OvO topology classifiers perform on a just slightly lower level within a maximum margin 4% difference for FS$_x^T$ and 1% to 3% for the others, where OvO_RF accounts for the majority of the larger 4% gaps. However, there is a drop (-3.5%) of OvO_RF using FS$_2$. The main reason is that RF performs a random feature selection during training and this particular case selecting just a few feature elements from the already small feature dimension of size 11. Furthermore, in the OvO approach, this is done for each of the 10 binary classification pairs to discriminate the terrain classes.
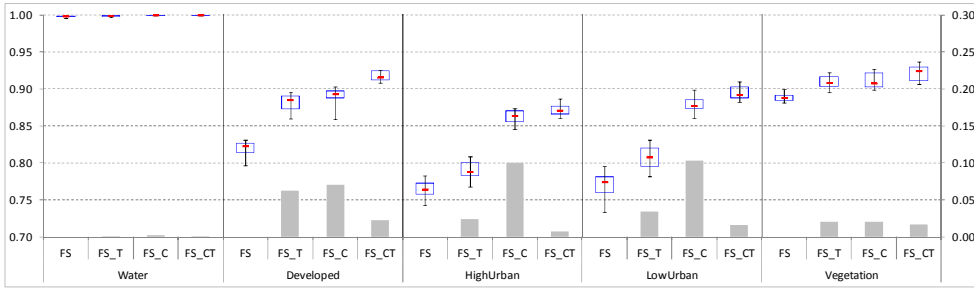
Figure 41 – Classification accuracies for the *SFBay_C* classes and differences in classification accuracy comparing the influence of texture and color features per class in the color feature experiments. The box plots show 25 and 75 percentile as the blue rectangle, the red line indicates the median, and whiskers representing the min. and max. values, respectively, with accuracies on the primary axis on the left. The gray bars indicate the feature accuracy differences with the secondary axis on the right.
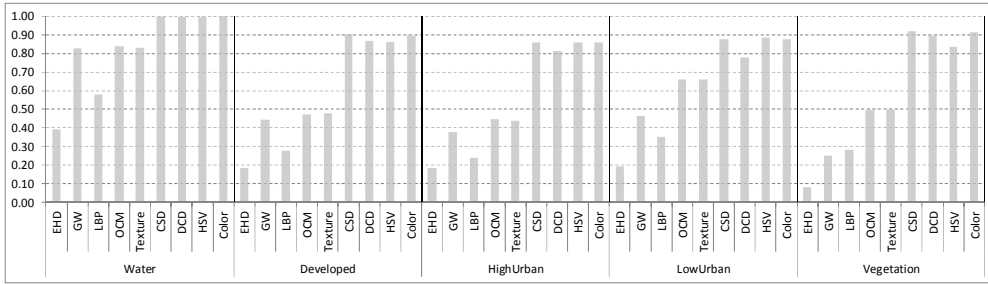
Figure 42 – Classification accuracies for the *SFBay_C* classes over single and combined texture and color features.

The classification accuracy of the *Water* class stays constant (Figure 41) due to its unique single-bounce backscattering discrimination, whereas the other classes are scatter mixtures of double-bounce and volume scattering. For other classes, the texture features have a low contribution on *High Urban* and *Vegetation* classes with around 2% and larger influence with 3% and 6% for *Low Urban* and *Developed*, respectively. However, even though the overall classification accuracy gain of the man-made terrain classes is minimal (Figure 41), the effect of the texture features is still visible with regard to the box plot shapes showing more variance within FS_x^T than FS_x^C due to the larger quartile body. Anyhow, the color features have varying contributions from lower 2% for *Vegetation*, to significant higher 7% for *Developed*, and 10% for the two urban classes, *High Urban* and *Low Urban*. Evidently even though the man-made terrain classes have textural patterns (roads and buildings), as visible in the results of the single texture features in Figure 42, differences among these classes are not discriminated enough by the texture features as they are by the color features.
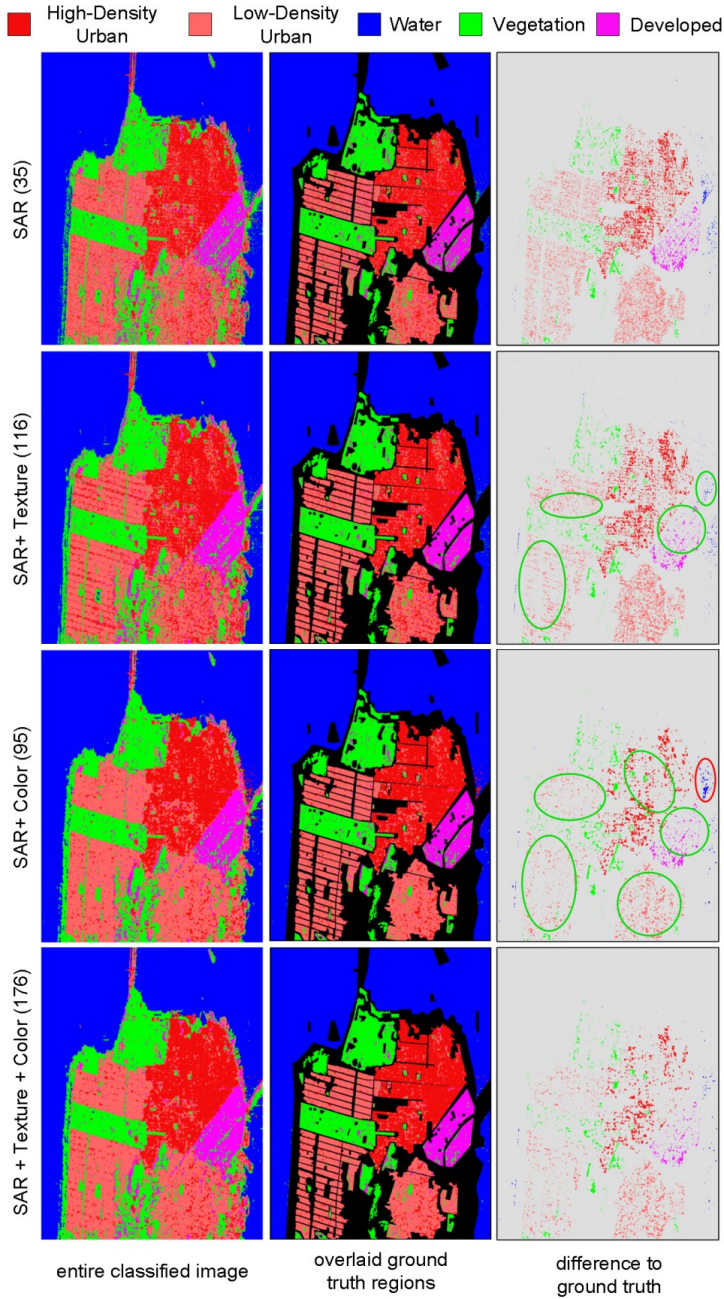
Figure 43 – Classification results for the SFBay_C image for FS_5 over OvA_SVM. The left column is the classified PolSAR image, the middle column is the classified image overlaid with the ground truth, and the right column shows the difference to the ground truth with the color indicating what it should have been. Circles mark degradation (red) and improvement (green). The first row shows results for FS_5, the second row FS_5^T, the third row FS_5^C, and the last row FS_5^{CT}.

This is further visible in Figure 43, which presents visual classification results for the *SFBay_C* image for the OvA_SVM classifier showing the differences of PolSAR features (FS$_5$) versus its augmented feature sets with texture, color, and texture + color, respectively. In the FS$_5^T$ case, it is capable of providing better results of the *Water* region around the bridge on the right hand side of the image. In addition, the improvements in the classification of the *Developed* and of the *Low Urban* areas are visually noticeable compared to FS$_5$ in the graph of Figure 41. However, only parts of the *Low Urban* areas on the left side of the image show better classification results whereas the area in the bottom right corner is only slightly affected. On the other hand, the classifier with FS$_5^C$ achieves better performance for all larger *Low Urban* areas as well as the *Developed* and *High Urban* regions on the right hand side.

Overall, when evaluating individual classes in all PolSAR image datasets, *Water* was the least affected by the addition of texture or color features due to the aforementioned reasons. *Forest* or main tree-like *Vegetation* benefited from color more with around 2% to 5% whereas with texture it is around 1% to 3%. In this regard, man-made terrain classes such as *Urban* areas profit the most due to their underlying surface characteristics, so that for the *Flevo_C* and *SFBay_C* images, they could gain up to 10% reaching or exceeding 90% level in classification accuracy.

Generally, it is evident that with the additional color features higher classification accuracies can be obtained. When a certain texture pattern is present within the classes, the texture features can enhance the classification accuracies; however, vice versa is also possible that when none of the classes have significant textural pattern, the texture features may rather blur or degrade the discrimination, which in turn will cause degradations in the classification performance. On the average, over all experiments, the color features outperformed the texture features in terms of classification accuracy improvements in all PolSAR feature combinations.

3.3.5. Performance Evaluation with Different Pseudo Color Images

After this first color feature evaluation using the Pauli color-coding (H,V), this section evaluates now additional pseudo color images for fully PolSAR data as described in Section 3.1. The experimental setup is similar to the initial extensive evaluation with regard to data, classifiers, and features. Comparative evaluations are performed over the two RADARSAT-2 images San Francisco (Section 2.5.3) and Flevoland (Section 2.5.2) with their focus on manmade/urban and vegetation/crop classification, respectively. Approximately 500 pixels per class for training and 50000 pixels for testing have been selected. What is more, the training data were kept as 50 percent for training and 50 percent for validating the classifier parameters.

For these two particular PolSAR images, the previous results have shown that they are not critically depending on the used classifier topology. Hence, only the pairwise multiclass SVM (OvO_SVM) is considered due to the increasing popularity of SVM in SAR image classification applying the same optimization procedure as before (see previous Section) to determine the best SVM configuration in case of kernel and its parameters.

Regarding the texture features, we use the same four features as in the previous Section with Local Binary Pattern (LBP), the MPEG-7 Edge Histogram Descriptor (EHD), Gabor Wavelets (GW), the Ordinal Co-occurrence Matrix (OCM) as a GLCM extension as described in Section 2.3 and 0. Similarly, for the extraction of the color features, we again use the HSV (Hue, Saturation, Value) color histogram, the MPEG-7 Dominant Color Descriptor (DCD), the MPEG-7 Color Structure Descriptor (CSD), and additionally the MPEG-7 Color Layout Descriptor (CLD) as describe in Section 3.2. Moreover, the individual texture and color features are merged in to a combined texture, T, and combined color, C, feature set, respectively, summarized in Table 8. The combined feature sets are referred to as FS^T, FS^C, and FS^{CT}.

Table 8 – Texture and color feature vector combinations applied for evaluation of different pseudo color image images.

Feature combination	Feature vector	Dim.	Description
T	$FV_6 + FV_7 + FV_8 + FV_9$	81	Combined extracted texture features
C	$FV_{11} + FV_{12} + FV_{13} + FV_{14}$	72	Combined extracted color features

Table 9 – Extracted feature vectors and combinations with their respective dimensions as considered for evaluation of different pseudo color image images.

$H\alpha A$	LBP	EHD	GW	OCM	DCD	HSV	CSD	CLD	FS^T	FS^C	FS^{CT}
11	16	5	24	36	4	24	32	12	81	72	153
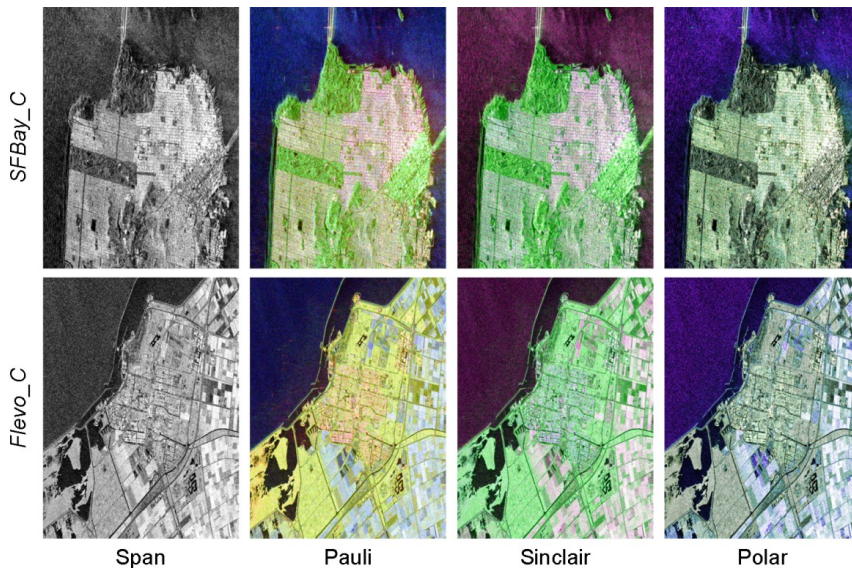
Figure 44 – Samples of different pseudo color and intensity images considered for evaluation.

We further extract features based on the $H\alpha A$ decomposition in form of entropy H, anisotropy A, the average angles averages $\bar{\alpha}, \bar{\beta}, \bar{\delta}, \bar{\gamma}$, the three eigenvalues, RVI, and $Span$ for comparison and augmentation as it is one of the most considered target decomposition for fully polarimetric data. These are also the same features as previously used in FV_2. The individual dimensions of all used features are provided in Table 9. Note that speckle filtering is applied as previously before any feature is extracted.

Our focus is on the evaluation of the texture and color features when extracted over different pseudo color representations considering three different pseudo color-coding approaches as described in Section 3.1 namely the *Sinclair* color-coding, the *Pauli* color-coding, and the *Polar* color-coding. An example of these pseudo color-coding is illustrated in Figure 44 for two PolSAR images. Additionally, the single intensity image representation of the *Span* is used defined as the total scattered power in a polarimetric radar system. Note that the texture features are extracted over the intensity representation of the *Span* and pseudo color images as obtained by $I = 0.2989 \cdot R + 0.5870 \cdot G + 0.1140 \cdot B$. I is equivalent to the luminance component Y of the YIQ / YCrCb color space defined in the NSTC / PAL color TV systems, respectively.
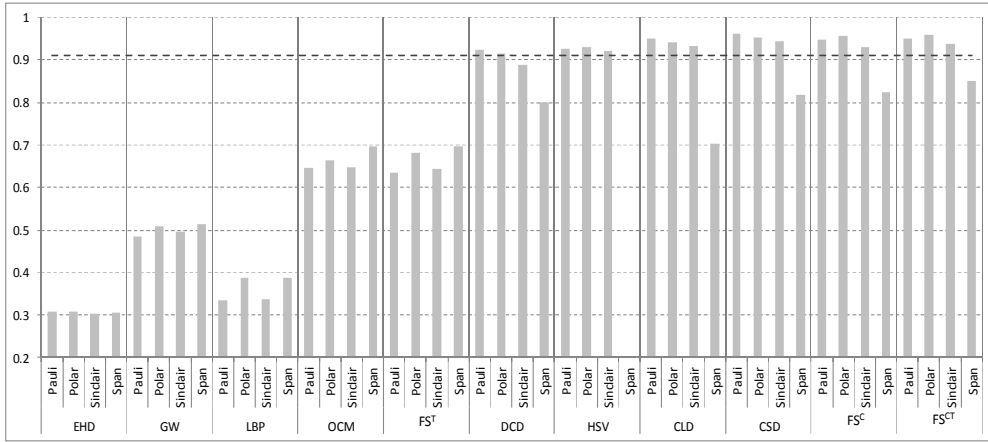
Figure 45 – Average classification results over different pseudo color images on *Flevo_C*. Dashed line is the classification accuracy achieved with $H\alpha A$ based features.

All the presented results are average classification accuracies obtained over 16 runs computing the trimmed mean leaving out the minimum and maximum classification accuracies to reduce the effects of outliers. The general observations of the extracted visual features over the particular *Pauli* pseudo color image have been shown in the previous evaluation. Therefore, the focus lies on the different pseudo color images and their effect on SAR image terrain classification. The classification results include the results over the *Span* image mainly for comparison reasons to evaluate how effective the color features work over intensity images.

We shall start with the evaluation over the *Flevo_C* image. As previously observed, texture feature performances are quite diverse in terms of classification accuracies. With respect to the different pseudo color images, classification results shown in Figure 45 are pretty much level for EHD and GW, yet there is a minor gap among accuracies with the GW feature when

extracted over the *Polar* and *Span* images compared to the other two. This accuracy gap significant increases for extracted features LBP and OCM. The reason is that both of them work directly with underlying intensity values whereas EHD and GW apply edge filtering with thresholds and multi-scale filtering, respectively, before extracting details about possible texture pattern. This will reduce difference among intensity values and affect discrimination capabilities.
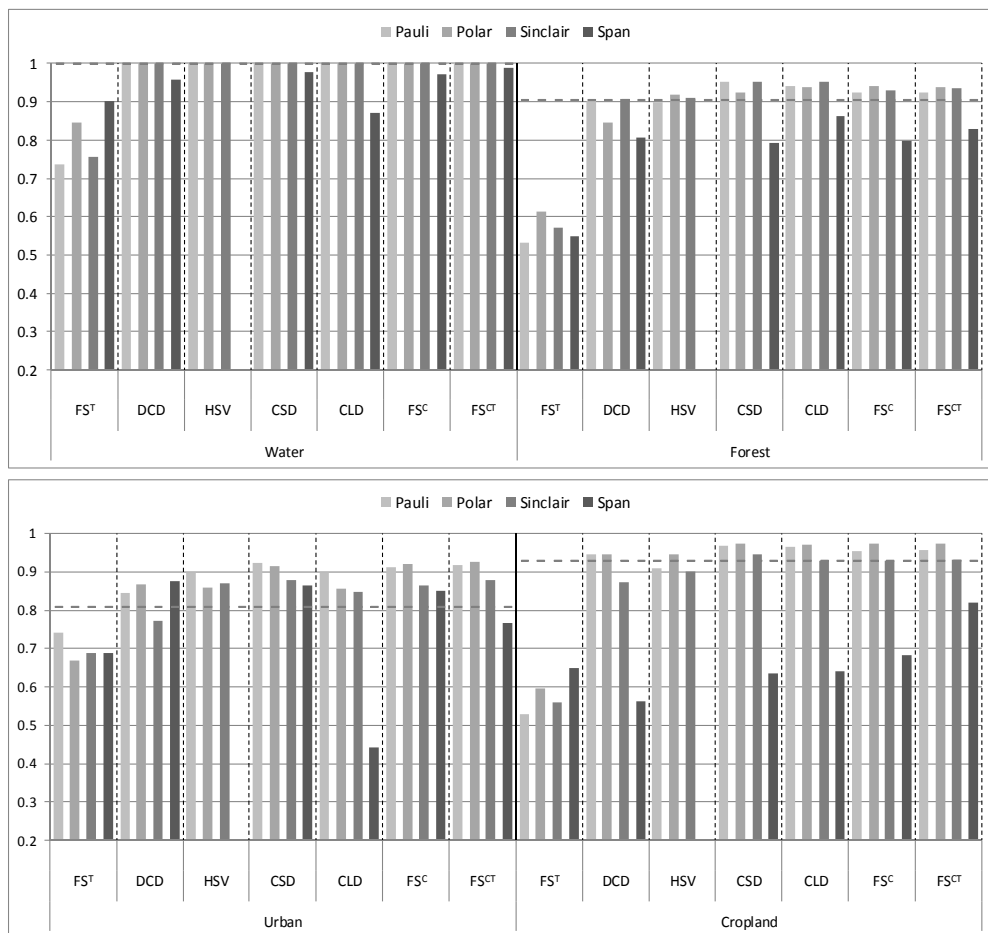
Figure 46 – Individual terrain classification results over different pseudo color images on *Flevo_C*. Dashed line is the classification accuracy achieved with *HαA* based features.

Classification results in Figure 46 for the color features over all pseudo color images are around 90 percent or higher. Results achieved over *Pauli* and *Polar* images are, however, slightly better than the one with the *Sinclair* image due to the different pseudo color generation. It is anticipated that accuracies obtained with *Span* image are lower caused by its gray level intensity nature, which will lack discrimination in color space conversions within the applied color features. However, it is an interesting observation that results achieved with the

combined color features, FS^C, over *Span* image are significantly better than its counterpart FS^T, which is related to the particular terrain classes in this classification setup.

When evaluating the individual class results as presented Figure 46, we can see that classification results using the texture features over *Polar* and *Span* images are among the highest particularly for natural surfaces such as *Water*, *Forest*, and *Cropland*. This is related to the image generation as the *Polar* image contains the *Span* image as its blue component, which in this case will carry the main distinguishable texture patterns compared to α and H. Moreover, accuracies with FS^T over the *Pauli* image result in the lowest performances for these natural terrains with accuracies over the *Sinclair* images being just slightly better. On the contrary, highest accuracies are achieved for *Urban* over the *Pauli* image when using the texture features. This is related to the combination of polarization matrices in the pseudo color images.

In case of color features over the individual classes (see Figure 46), the different pseudo color images do not affect the accuracy for *Water*. This is expected, as there are clear visual differences to the other classes in all images, which result from the different backscattering characteristics for *Water* compared to the other surface classes. For the *Forest* areas, the different pseudo color images do not significantly affect the classification accuracies using the features HSV, CSD, and CLD. We can observe that the best results can be achieved using CSD and CLD as both of them include color relationships within their feature description so as encoding possible underlying texture patterns. The noticeable drop in accuracies over the *Polar* DCD feature results from the fact that only the single largest color cluster is used to describe the pixel neighborhoods. This does create potential mismatches between *Forest* and *Urban* as they both might be represented with similar colors (see Figure 44) due to the DCD's underlying color clustering and merging. The same problem arises with *Urban* and *Cropland* for DCD when extracted over the *Sinclair* image. Similarly, to the texture feature observation, the best classification results for *Urban* are obtained with color features over the *Pauli* image as it provides the clearest color difference among the classes. Within the results for the *Urban* class, the accuracy for the CLD feature over the *Span* image is significant lower than for the other color features. This is caused by CLD's color averaging for each processed block over an image region. The average intensity value is not representative enough for this particular class. As in the next step of CLD, DCT is performed over the color components individually and cutting its high frequency elements, this further results in loss of pattern information for discriminating *Urban* areas. For the *Cropland* class, accuracies over *Pauli* and *Polar* images are best as they provide the best color discrimination to the other classes; and CSD and CLD results over all pseudo color images are higher due to their integration of color relationships.

Overall, in this particular classification setup, the attained performances among the three generated pseudo color images do not have any significant differences. There are minor gaps for the *Sinclair* image results with its rather simple assignment of polarization matrices. Yet performances over *Pauli* and *Polar* images seem on average more stable for the different terrain classes as their pseudo color image generations use combinations of the available polarization matrices or even employing TD components.
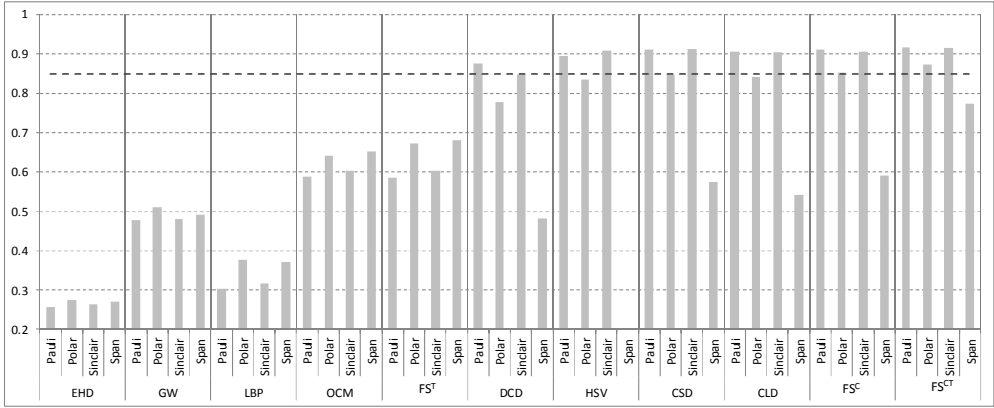
Figure 47 – Average classification results over different pseudo color images on *SFBay_C*. Dashed line is the classification accuracy achieved with *HαA* based features.
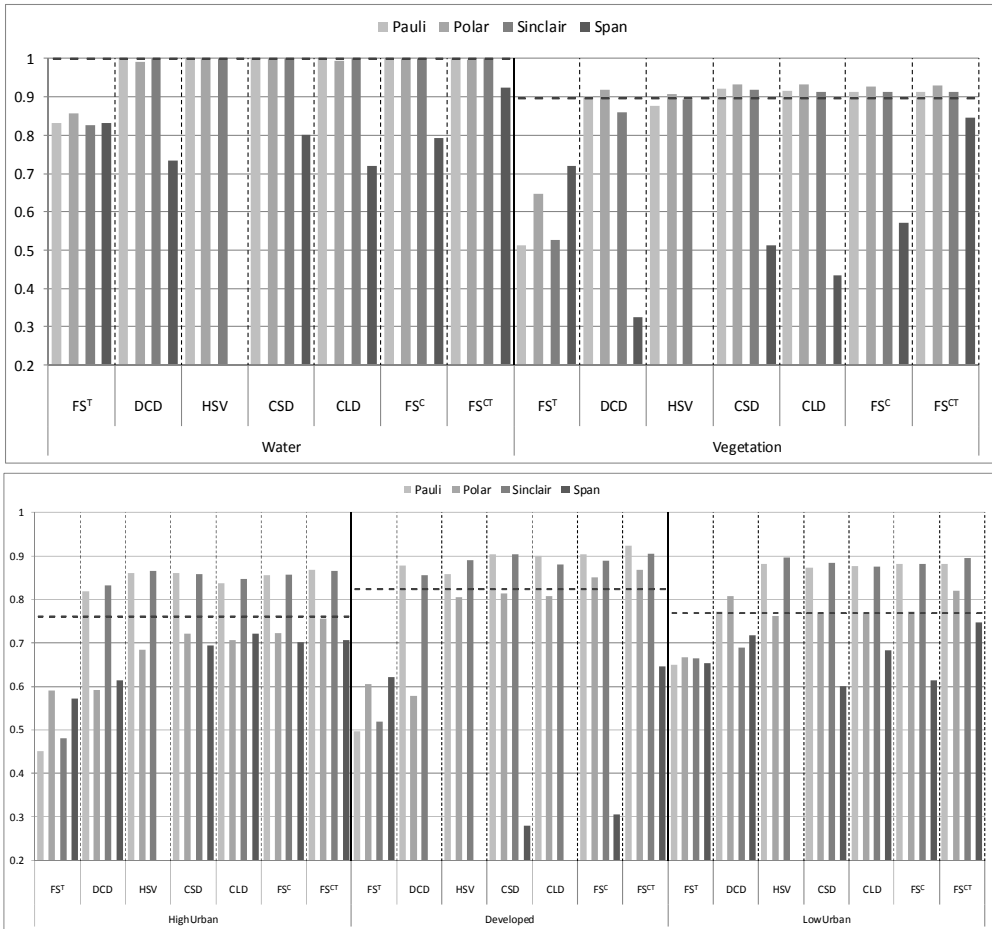
Figure 48 – Individual terrain classification results over different pseudo color images on *SFBay_C*. Dashed line is the classification accuracy achieved with *HαA* based features.

Evaluating the classification results for the *SFBay_C* data with its more man-made / urban terrain classes (see Figure 47) shows that the general observations related to the texture feature are similar to the *Flevo_C* data. We can see a similar accuracy gap for the *Polar* and *Span* images, which has slightly increased for the individual texture features compared to *Flevo_C*. Considering the color features, the main difference to *Flevo_C* is the 5-6% better results achieved over the *Pauli* and *Sinclair* images. Moreover, color features extracted over the *Span* images result on average in worse a classification performance in this classification setup.

When evaluating the individual class results as presented Figure 48, we can see that classification results for *Water* using the texture features show no significant differences among the pseudo color images. Similarly, texture feature results for *Vegetation* match observations made for the natural surfaces in the *Flevo_C* data. Likewise, accuracies obtained with the color features are almost equivalent beside a minor 1-2% increase over the *Polar* image related to *Vegetation*'s darker color compared to the other classes reducing potential mismatches.

As for the main terrain surfaces of man-made areas in this PolSAR image, *High Urban* and *Developed* accuracies for texture features shared the same behavior as seen for *Vegetation* favoring *Polar* and *Span* image results whereas *Low Urban* results do not show such differences for the pseudo color images. This means that for the other two man-made classes, the *Pauli* and *Sinclair* images do seem to lose underlying information related to texture pattern due to that fact that both of them are generated using polarization matrices. On the other hand, the *Pauli* and *Sinclair* images generate different visual color representations for the various man-made regions making it easier for the color features to provide distinctive descriptions. Hence, their obtained results using the color features are significantly better than for the *Polar* image. However, for the *Polar* image, the DCD feature lacks discrimination power compared to the other color features as CSD, CLD include color relationships, and HSV carries more information within its histogram representation. Compared to other classes, the color features extracted over the *Span* image fail to provide any comparative results for *Developed* as its intensity values are similar to *Low Urban*. This shows that color is a main discriminator for *Developed*. Related to the FS^T and FS^C *Span* results over *Flevo_C*, *High Urban* is the only class where we can observe the similar results again. This is probably regarded to the classification setup and data, where the terrain classes have strong texture pattern characteristics.

Overall, in this particular classification setup, the attained performances among the three generated pseudo color images do show significant differences in differentiating various urban areas. There are major gaps on these urban areas for the *Pauli* and *Sinclair* image results using color features due to their unique visual color representation. Differences to results over the *Polar* image are as high as 10%.

3.4. COLOR OVER PARTIALLY POLARIMETRIC SAR DATA

After the promising classification results for fully polarimetric SAR images, the focus is directed to investigating the potential of color features when applied in supervised classification over partially PolSAR images. The main difference now is that the full polarimetric infor-

mation is not available, as obviously in dual-pol data there are only two of the four possible polarization combinations whereas in single-pol SAR data just one polarization matrix is available. Therefore, the generation of the necessary pseudo color image is different from the fully polarimetric data as described in Section 3.1. The main objective here is to investigate color features from pseudo color-coded images generated from two partially PolSAR images one from the TerraSAR-X and the other on from the COSMO-SkyMed system.

The rest of this evaluation is organized as follows. The experimental setup presents the PolSAR image data, and the polarimetric SAR and image processing features before the classification results over combinations of PolSAR, texture, and color features are evaluated.

3.4.1. Image Data

This particular evaluation investigates the dual-pol TerraSAR-X data and the single-pol COSMO-SkyMed data with their ground truth as described in Sections 2.5.4 and 2.5.5, respectively. The COSMO-SkyMed image is been speckle filtered whereas the TerraSAR-X is kept as is due to its radiometrically enhanced multi-look ground range processing already reduces speckle. Table 10 provides a brief summary of the two PolSAR images.

With the generated ground truth, the following procedure is adapted to create the training and testing sets for the PolSAR images. The train dataset of the TerraSAR-X image contains approximately 1000 pixels per class; and the test dataset has around 100000 pixels per class. As for the COSMO-SkyMed image, the selected ground truth consists of around 2000 pixels per class for training and 100000 pixels per class for testing. The training dataset size is doubled due to the larger image size and its single-pol nature. What is more, the training datasets were split into 50 percent training and 50 percent validation for optimizing the classifier parameters; and Table 11 provides a summary of the train and test dataset used per PolSAR image.

Table 10 – Overview of image data used within color feature experiments over partially polarimetric SAR images.

Name	System	Mode	Date	Dimensions	Incident angle
Dresden	TerraSAR-X	Dual	Feb 2008	2209×3577	41°
Po Delta	COSMO-SkyMed	Single	Sep 2007	4642×3156	30°

Table 11 – Partially polarimetric SAR image data with their train and test sets for color feature evaluation.

Name	No. classes	Train size	Test size	Ratio Test/Train	Abbr.
Dresden	6	6000	600000	100	*TSX*
Po Delta	6	12000	600000	50	*CSK*

3.4.2. Features and Classifier

The used color features are the same features as in the previous Section 3.3.5 with HSV (Hue, Saturation, Value) color histogram, the MPEG-7 Dominant Color Descriptor (DCD), the MPEG-7 Color Structure Descriptor (CSD), and the MPEG-7 Color Layout Descriptor (CLD) as describe in Section 3.2. As for the extraction of the texture features, the Local Binary Pattern (LBP) and the MPEG-7 Edge Histogram Descriptor (EHD) are not considered due to their rather poor performances over the fully PolSAR images; and it is not expected to change for partially PolSAR images. As a replacement, the proposed SAR texture feature Multilevel Local Pattern Histogram (MLPH) is added, as it had been used over single-pol SAR images.

Table 12 – Texture and color feature vectors and combinations used in the partially polarimetric SAR image color feature classification experiments.

Feature set	Feature vector	Dim.	Description
GW	FV_1	24	Gabor Wavelets
OCM	FV_2	36	Ordinal Co-occurrence Matrix
MLPH	FV_3	75	Multilevel Local Pattern Histogram
DCD	FV_4	4	Dominant Color Descriptor
HSV	FV_5	24	Hue, Saturation, Value histogram
CSD	FV_6	32	Color Structure Descriptor
CLD	FV_7	12	Color Layout Descriptor
FS^T	$FV_1 + FV_2 + FV_3$	135	Combined texture features
FS^C	$FV_4 + FV_5 +$ $FV_6 + FV_7$	72	Combined color features
FS^{CT}	$FS^T + FS^C$	208	Combined texture and color features
$SAR + FS^X$	$FS^T / FS^C / FS^{CT}$	-	Combine visual feature sets with respective PolSAR features
FS_D^{BL}	VH FS^T + VV FS^T	272	Combined texture features extracted over VH and VV, baseline feature set for dual-pol image

Besides extracting the individual visual feature vectors, the individual texture and color features are combined to generate one texture feature set, FS^T, one color feature set, FS^C, and one feature set containing all visual features, FS^{CT}. Furthermore, these three feature sets are combined with the respective PolSAR feature for the dual-pol and single-pol data using the abbreviation $SAR + FS^X$. Table 12 summarizes the extracted features, their combinations, and dimensions.

For the dual-pol SAR image, color features are extracted over the *RGB2* pseudo color representation (detailed in Section 3.1) as it is expected that similar results can be achieved for the different visualization *RGB1* and its reversed combination of the VH and VV data, which contents the same underlying information. The texture features are extracted over the intensity

representation of the *Span* and *RGB2* as obtained by I = 0.2989·R + 0.5870·G + 0.1140·B. I is equivalent to the luminance component Y of the YIQ / YCrCb color space used in the NSTC / PAL color TV systems, respectively. Due to the dual polarization (VH/VV), generally texture features are extracted separately for each polarization matrix (e.g., FS^T over VH and FS^T over VV). They are then concatenated and combined with the dual PolSAR (VH/VV) features to form the final feature vector (FS_D^{BL}) mainly for comparison. Over the single-pol SAR image, the texture features are extracted over the intensity representation of the HH polarization matrix, and the color features are extracted over the *HSI* pseudo color representation of the HH intensity image as detailed in Section 3.1. Similar to the dual-pol data, the combination of the single HH PolSAR feature and the FS^T from its intensity image representation is used as the main feature set for comparison.

Our focus is to investigate the contribution of individual visual features, especially color, to the classification outcome and thus to perform comparative evaluations against well-known PolSAR and texture features employing the pairwise multiclass SVM (OvO_SVM) applying the same optimization procedure as before to determine the best SVM configuration in case of kernel and its parameters.

3.4.3. Experimental Results

We will discuss observations related to the two different polarimetric modes: dual-pol and single-pol, firstly presenting some general classification performances before going in depth analysis for each individual PolSAR image. All the presented results are average classification accuracies obtained over 16 runs computing the trimmed mean leaving out the minimum and maximum classification accuracies to reduce the effects of outliers. As introduced in Section 3.4.2 and Table 12, the respective feature set abbreviations are used to refer to the different feature sets in the text.

The dual-pol TerraSAR-X image results providing average classification accuracies for individual visual features and their combinations are presented in Figure 49. As for the performance evaluations of the texture features, GW and MLPH achieve around the same classification accuracies whereas OCM reaches around 5% better results. The combined texture features, FS^T, extracted over the image *RGB2* has the highest accuracies even though the classification performance differences to the corresponding feature over *Span* are minor. OCM and FS^T have quite similar classification performances, as OCM seem to provide the main contribution in this texture combination. When joining FS^T with the dual PolSAR (VH/VV) feature, significant improvements can be achieved with 5% - 6% showing the additional contribution of the two PolSAR features. The two percent gap between the performance over the *RGB2* and *Span* image probably results from the two different image compositions where *RGB2* and the dual PolSAR (VH/VV) feature seem to provide a little more diversity between each other. *RGB2* incorporates both VH and VV polarization matrices information in the different color components and *Span* represents the total power over VH and VV as an intensity representations. In Figure 49 it is observable that a slightly lower classification performance of -3% is

achieved by extracting the texture features over the *RGB2* color image compared to the baseline feature set, FS_D^{BL}, which, however, has twice the feature dimension. This is due to the extraction of FS^T over both VH and VV polarization matrices separately.

As for the color features, the main observation is the performance of the individual color features achieving higher accuracies compared to the best individual texture feature, OCM. Similarly, to the texture features, the combination of all color features, FS^C, results in an equal performance as the single best color features, CSD and HSV. Additionally, the color features DCD and CLD are not able to provide the same level of discrimination due to their more compact and lower feature dimensions. Furthermore, color features CSD, HSV, and FS^C yield an accuracy gain of up to 2% compared to the FS_D^{BL} performance. With the addition of $FS^{T,}$ a further improvement of 1.5% can be achieved exceeding the baseline results up to 3.5% with lower overall feature dimensionalities, particularly as using the combination of FS_D^{BL} with FS^C will result in similar performances.
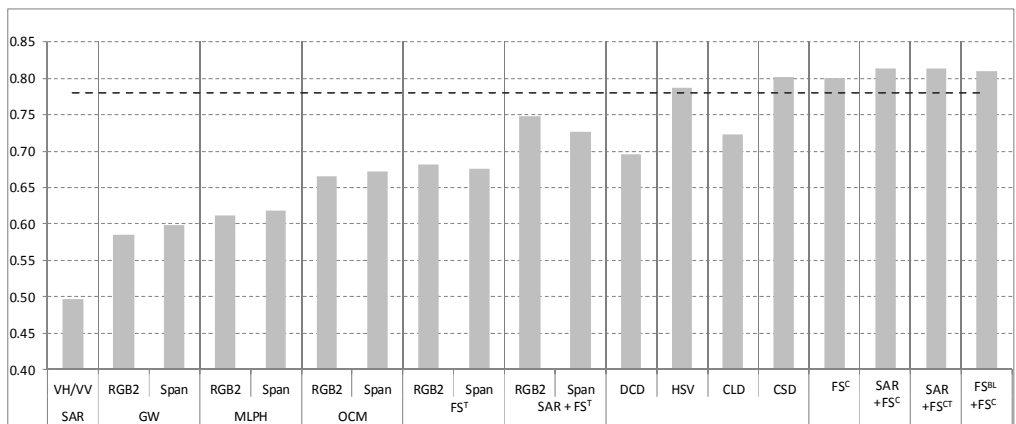
Figure 49 – Average classification accuracies using the individual features and feature combinations over the dual-pol TerraSAR-X image. The dashed line is the result for the feature combination (FS_D^{BL}) of dual PolSAR (VH/VV) plus FS^T over VH and FS^T over VV.

Figures 50 to 52 illustrate the classification results for the individual terrain classes. It is evident that for the *Urban* class in Figure 50, the dual PolSAR feature alone provides a poor discrimination since it cannot describe the underlying pattern of urban areas with buildings, roads, and urban vegetation due to its pixel-based nature. With the addition of texture features, the average classification accuracy is raised to 65 percent due to the fact that the texture features can better describe the typical pattern of urban areas, particularly OCM with is multiple seeds. It can be observed that all feature sets, which include texture, have a similar level of classification accuracy. Note further that the overall dimensions of these feature sets (135-208) are smaller than the baseline feature set dimension (272). Overall, performance of color features reaches similar levels compared to the texture features, yet the additional contribution seems minor for the classification of the urban terrain type.

Since the terrain of the *Industrial* class has a similar pattern as the *Urban* class, comparable observations and conclusions for the dual PolSAR (VH/VV) and texture features can be made. We can observe that the achieved classification accuracies using the color features, CSD, HSV, and FS^C, are just slightly below the performance of the texture features, FS^T. In case of the DCD feature, classification accuracies are quite low because the DCD suffers from the larger variety of colors to cluster and when picking the largest cluster it might not be the best representation for this class characteristic. Hence, there is generally a lot of misclassification with the *Urban* class. Again, texture features provide major discrimination already so the color features can only provide minor improvements to the performance level achieved with the texture features.
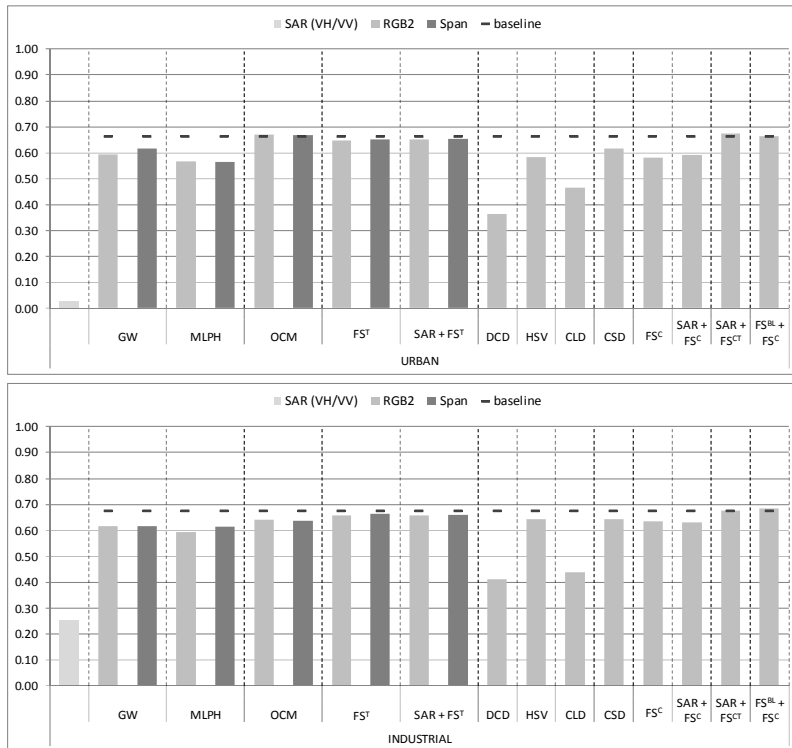
Figure 50 – Average classification accuracies using the individual features and combinations for the terrain types, *Urban* and *Industrial,* over the dual-pol TerraSAR-X image. The dashed line is the result for the feature combination (FS_D^{BL}) of dual PolSAR (VH/VV) plus FS^T over VH and FS^T over VV.

The water bodies, *InWater*, are classified with a (nearly) 100 percent accuracy for most of the feature sets including the two PolSAR features, as surface scattering, textural pattern or color significantly differs from the other classes.

For the *Forest* class, an observation from Figure 51 is that classification accuracies using the GW or MLPH feature drop significantly compared to the other classes due to misclassifi-

cations with the classes *Urban*, *Pastures*, and *Arable Land*. The main reason is that all these classes are represented by similar intensity values particular for the *Span* image. As GW relies on first and second order moments and MLPH employs thresholds as an initial step, they both lack the discrimination capabilities compared to OCM. Furthermore, employing the OCM feature over the *Span* image, classification accuracies are considerably lower than the corresponding ones over the *RGB2* image due to similar *Span* intensity values with *Arable Land*, which affects the co-occurrences within the different distances and directions. Note also that this is the only class, where the results obtained with the texture feature extracted over *RGB2* and *Span* images are significantly outperformed by the baseline feature set results, which is also the main reason for the 3% drop observed in the average classification accuracies. This is because the baseline feature set includes texture features extracted over both VH and VV polarizations, which seems to provide better discrimination for such a terrain class. When using the single and combined color features, better classification accuracies can be achieved due to clear color discrimination present for the *Forest* regions (see Sections 2.5.4 and 3.1).
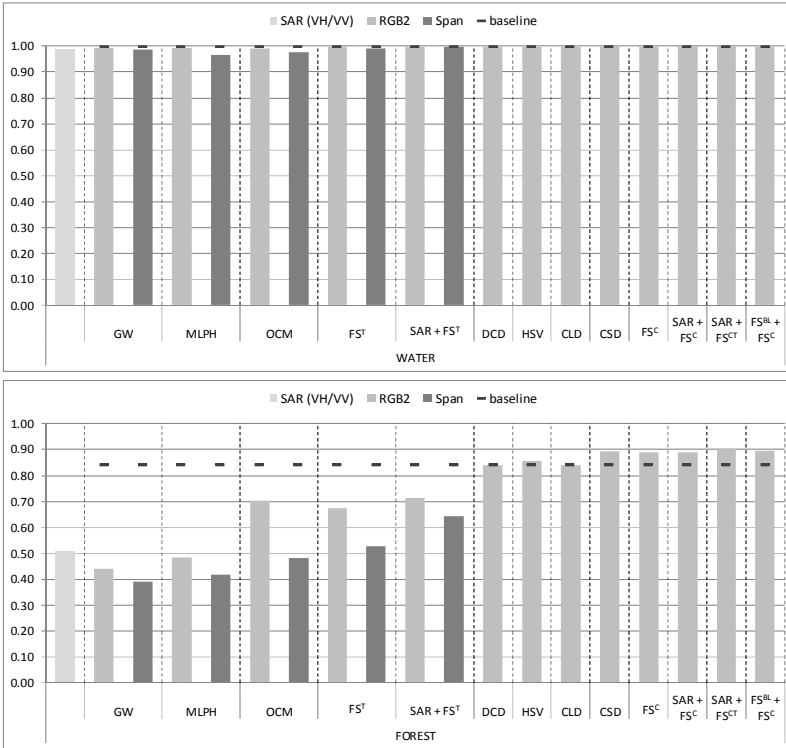
Figure 51 – Average classification accuracies using the individual features and combinations for the terrain types, *Water* and *Forest,* over the dual-pol TerraSAR-X image. The dashed line is the result for the feature combination (FS_D^{BL}) of dual SAR (VH/VV) plus FS^T over VH and FS^T over VV.

For the *Pastures* class, the first observation from Figure 52 is that the classification accuracies with texture features over the *RGB2* images drop substantially compared to performances

over the *Span* image. Contrary to the texture feature results for the *Forest* class, the total power represented by the *Span* seems to provide better discrimination for the texture features over *Pastures* regions. However, when combining PolSAR features with FS^T extracted over *RGB2*, it seems that the dual PolSAR feature can provide additional diversity, so that this combination is able to achieve similar results that the PolSAR + FS^T extracted over *Span*. Besides the single texture features and FS^T, all color feature sets perform on a rather similar classification level. This implies that the applied color features, single and their combinations, provide a good discrimination to the other five classes, whereas the texture features have lower discrimination abilities for this terrain class when extracted over the intensity representation of the *RGB2* image. Finally, it is observed that by using the color features similar or better classification accuracies up to 7% can be achieved compared to the baseline feature set results.
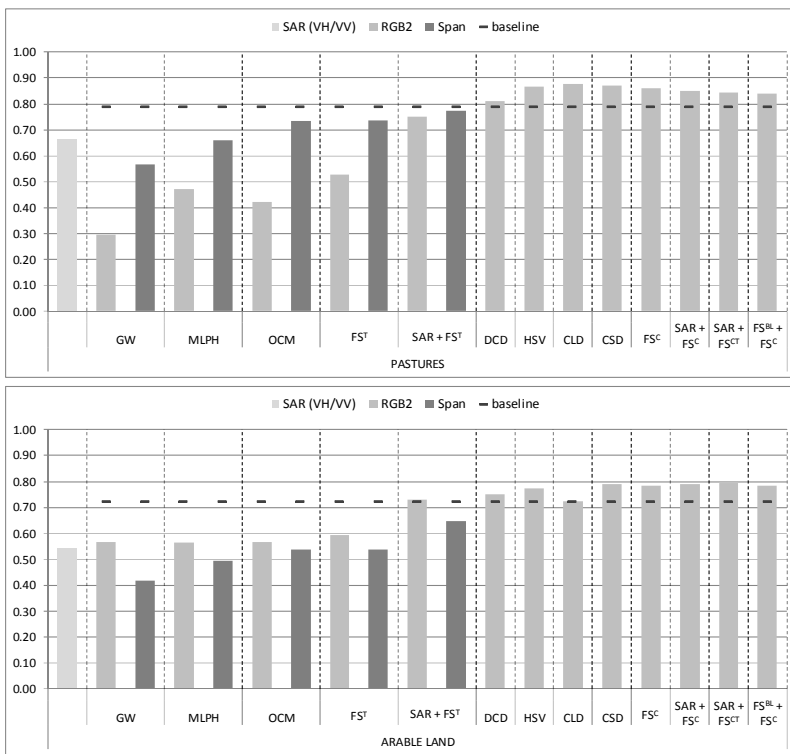
Figure 52 – Average classification accuracies using the individual features and combinations for the terrain types, *Pastures* and *Arable Land,* over the dual-pol TerraSAR-X image. The dashed line is the result for the feature combination (FS^{BL}_D) of dual PolSAR (VH/VV) plus FS^T over VH and FS^T over VV.

For the regions of *Arable Land*, as shown in Figure 52 using the single and combined color features extracted over the *RGB2* image, better classification accuracies can be achieved due to distinguishable color differences, particularly compared to the texture features, which achieve slightly higher performance than the dual PolSAR features results. However, in their combination the two different components PolSAR + FS^T provide good diversity to achieve

10% accuracy gains than results using FS^T. As for *Pastures*, single color features can provide higher discrimination than the baseline features resulting in up to 7% higher accuracies.
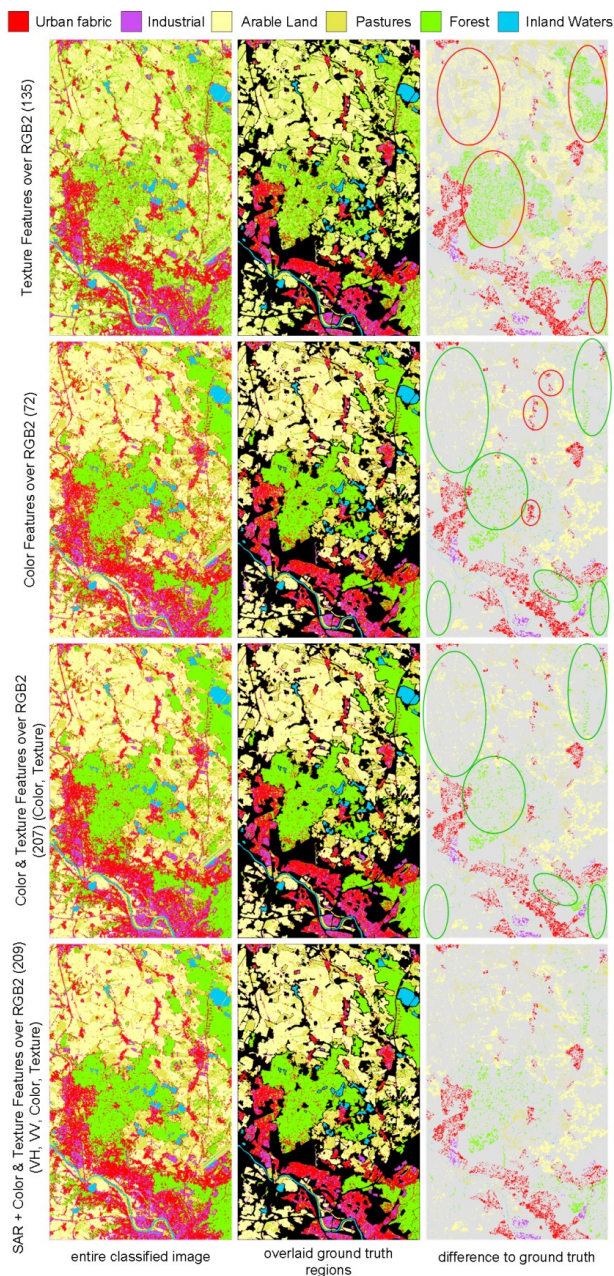
Figure 53 – Classification results for the dual-pol TerraSAR-X image. Left column is the classified SAR image, the middle column is the classified image overlaid with the ground truth, and the right the difference to six-class ground truth to the middle column, respectively, and with the color indicating what the ground truth should have been. Circles mark degradations (red) and improvements (green).

Figure 53 presents the visual differences of the classification results using the feature sets: texture features (FS^T), color features (FS^C) and color + texture feature sets (FS^{CT}) over the *RGB2* image. Note that in all three cases, the water terrain types are classified correctly; however, *Urban* and *Industrial* areas are misclassified, as shown in the bottom part of the images, and an even larger *Forest* area is misclassified as *Urban*. An important observation for the classification results using the FS^T features is that significant amount of misclassifications are evident among the classes, *Forest*, *Pastures*, and *Arable Land*, that can be easily identifiable in the third column, when overlaid with the ground truth. Regarding these three classes, different color feature combinations provide improved classification results as marked with the green circles. Nonetheless, they seem to overestimate *Urban* regions slightly as marked with the red circles. The classification results over *RGB2* image using the color and texture feature sets are almost identical to the results using FS^C with the exception that the texture features provide better classification accuracy for the *Urban* areas. Moreover, with the addition of the dual SAR features, further yet small visual improvements can be observed for *Urban*.

For the dual-pol TerraSAR-X data, the application of texture features carries a high amount of discrimination especially for man-made terrain classes. The applied MLPH features extracted over the respective intensity representation of the *Span* and the *RGB2* pseudo color image can provide comparable classification results to GW and OCM yet was not able to achieve (significantly) higher discrimination in this classification setup. Nevertheless, the combination of the three single features is quite effective. That being said, for the terrain classes *Forest*, *Pastures*, and *Arable Land,* the color features are able to provide similar or better discrimination particularly DCD and CLD with their feature dimensions of 4 and 12, respectively, when compared to the larger dimensions of the texture features. The main confusion between *Urban* and *Industrial*, observable in visual classification results, is expected has both have similar characteristics, which are not 100 percent distinguishable by the applied texture and color features.

About evaluations for the single-pol COSMO-SkyMed image, the top of Figure 54 provides average classification accuracies for the individual visual features and combinations over the *HSI* pseudo color image. Observations are that the results with OCM features are better than for MLPH and GW as seen for the dual-pol TerraSAR-X image. Once again this demonstrates particularly OCM has certain superiority for the purpose of SAR image classification. The individual color features extracted over the *HSI* pseudo color image achieve comparable results to the combined texture features as well as the single best texture feature, yielding up to 7% accuracy gains. Classification accuracies can be improved up to 2% by using the HSV, CSD, and the combined color features, FS^C, compared to the baseline feature sets. The addition of either the single PolSAR feature or the combined texture feature, FS^T, to FS^C provides a mere one percent accuracy gain, where in the case of the texture augmentation with the expense of higher feature dimensions from 136 to 209.

Regarding the results for the individual classes as illustrated in Figures 54 and 55, first observation for *Urban* is that the OCM texture feature is able to provide the best discrimination resulting in the highest texture classification accuracies and is the main contributor for the FS^T

results. Using the majority of feature sets yields in a similar level of classification accuracies particularly using the combined feature sets, FS^C and FS^{CT}, does not add any further discrimination power compared to the single color features, which accuracies already match the baseline performance. For this class, color features provide only minor contributions in comparison to the texture features.
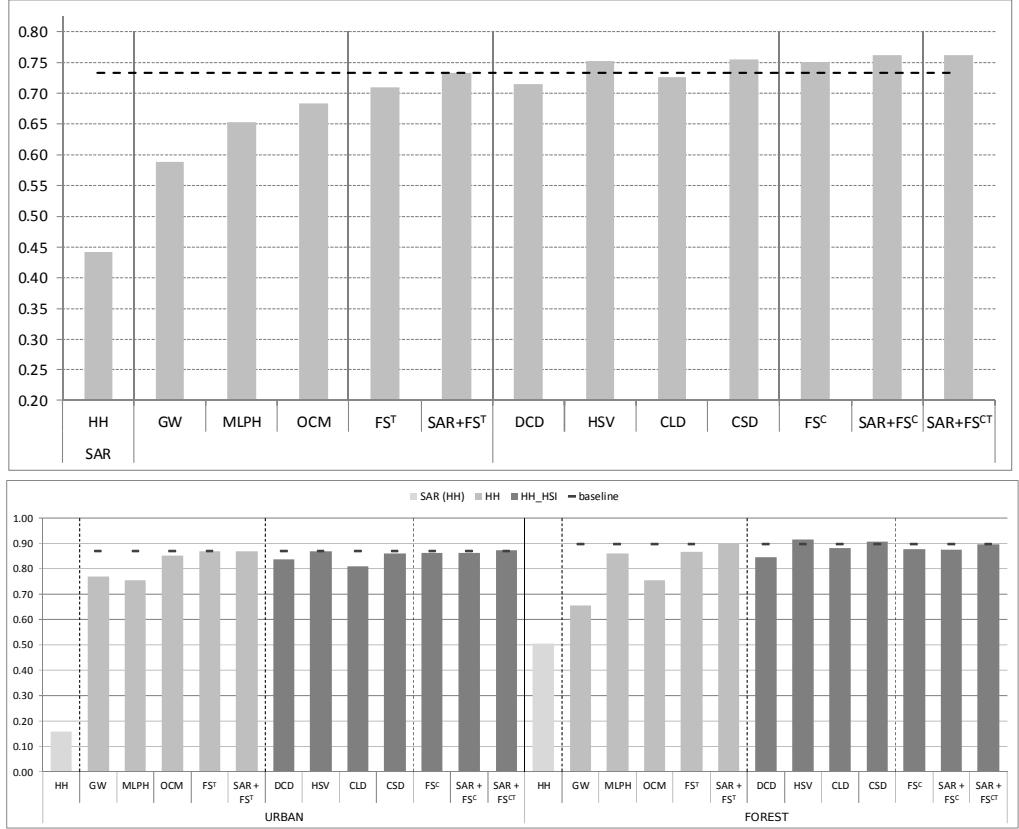
Figure 54 – (top) Average classification results for the individual features and combinations for the single-pol COSMO-SkyMed image. (bottom) Classification results for the individual features and combinations for *Urban* and *Forest* for the single-pol COSMO-SkyMed image.

Considering *Forest*, the classification performance using single color features are better than using the GW and OCM texture features, where the performance gap can be around 10% in terms of accuracy, and equal or slightly better than results obtain with the extracted MLPH feature, which is the main contributor to the FS^T results. As for *Urban*, color features can only match the texture feature performance but not excel it, which is related to the high average classification accuracy achieved by the features, as *Urban* and *Forest* are quite distinguishable from the other terrain classes (see Section 3.1).

As for the agricultural class, *Arable Land*, the classification performance for the texture features highly benefits from the single feature combination and augmentation with the HH PolSAR feature as illustrated in Figure 55 compared to the other terrain classes where these

combinations reach similar performances as the single best texture feature for this image. This means that all three individual texture features can provide different levels of diversity, which is required to achieve better results when combining single features. The classification performance using CSD and combined color features is around 3.5% higher than using texture features. When using the combined color features, FS^C, with HH and texture, classification accuracy improvements of 6% are achieved. This shows that additional discrimination information for this terrain class can be obtained by extracting color features over a pseudo colored intensity image.
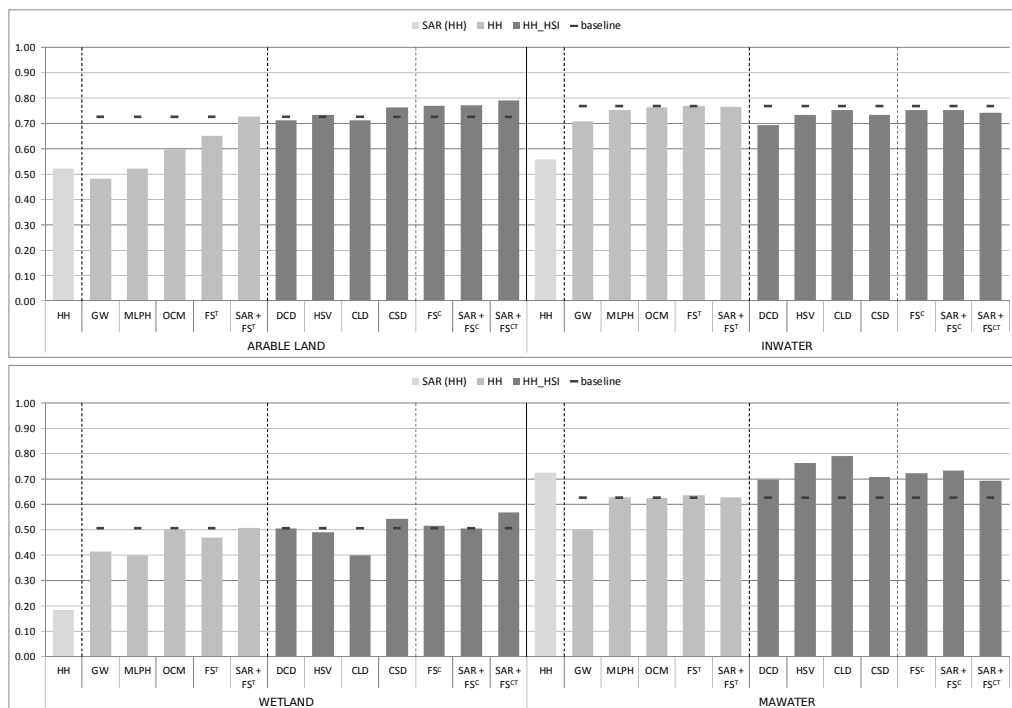
Figure 55 – Classification results for the individual features and combinations for *Arable Land, InWater, Wetland, and MaWater* for the single-pol COSMO-SkyMed image.

For the inland water class, *InWater*, a similar performance level (around 75 percent) is observed in Figure 55 by the majority of features. The performance level by using neither single nor combined color features can exceed the corresponding result using $SAR+FS^T$. Therefore, for this class, color features do not have any significant contribution on the classification accuracies.

As for the *Wetland* class in the bottom of Figure 55, varying classification accuracies are obtained using different feature sets, which results from the confusion with *MaWater* as both are maritime water classes, yet with a visual difference. Using of either the following features or feature sets, HSV, OCM, FS^C, $SAR+FS^C$, FS^T, and $SAR+FS^T$ will result in a similar classification accuracy of about 50 percent for this terrain class despite entirely different feature dimensions. However, when all features are used together the classification accuracy is im-

proved about 7% compared to the classification over individual visual features. This means that each individual feature still provides a different level of discrimination not evident from their respective single results.

For the marine water class, *MaWater*, a classification accuracy of 72 percent is obtained using the single HH PolSAR feature. The combination with texture features does result in a 10% drop in classification accuracy due to larger feature dimension of FS^T, so that the single HH features appears as an outlier. As for the single color features, they are able to achieve similar or better classification accuracies yet their combination, FS^C, will perform almost 10% worse than their two best single features. This is related particularly to CLD and its lack of discrimination between *MaWater* and *Wetland*, where its accuracy is 10% lower than for the other single color features so that it tends to classify either of the two classes as *MaWater*. Even though the color features achieve higher classification accuracies than the texture features, they are not able to exceed the performance of the single HH PolSAR feature. However, both texture and color features are able to provide some sort of discrimination between the two maritime water classes *MaWater* and *Wetland* whereas the single HH PolSAR features fails in classifying *Wetland*. This can be anticipated, as both classes will share a similar backscattering characteristic.

Figure 56 presents the classification results showing the visual differences of the texture features (FS^T), Color (FS^C) over *HSI* as well as *HSI* color + texture feature sets (FS^{CT}). Note that using all three feature sets, the regions of *InWater* class are mainly misclassified due to the presence of the thinner river arms, while the classification results for the main *Urban* regions are intact. For the results using the texture feature set, there is a major confusion among the different water classes *InWater*, *MaWater*, and *Wetlands* as well as *Forest* and *Arable Land*. Furthermore, several *Arable Land* areas are misclassified as *Urban* particularly along their boundaries; and large water regions are misclassified as *Forest*. Note that when using the SAR and color feature sets together over the *HSI* pseudo color image, the classification of the water regions is improved noticeably with fewer noisy occurrences over *MaWater* on the right, *Wetland*, and fewer misclassifications of *MaWater* as *InWater* or *Forest*. Whenever texture features are used along with SAR and color feature sets, more misclassifications are observed for the *Wetland* terrains and vice versa for the *Urban* areas, which is an expected outcome due to the unique pattern of this terrain class.

Overall, for this single-pol COSMO-SkyMed PolSAR image, classification results for the *Urban, Forest, and Arable Land* regions are high using texture and color features as their terrain characteristics are different to the water bodies. Among the water bodies, the color features are able to provide additional discrimination for an improved classification. However, similar to the dual-pol TerraSAR-X image, for *Arable Land* the color features seem to provide better discrimination than the applied texture features as they will not have strong texture pattern. There is anticipated confusion among the three water classes where *InWater* represented by river and river arms most probably benefits from window based feature extraction due to the river banks.
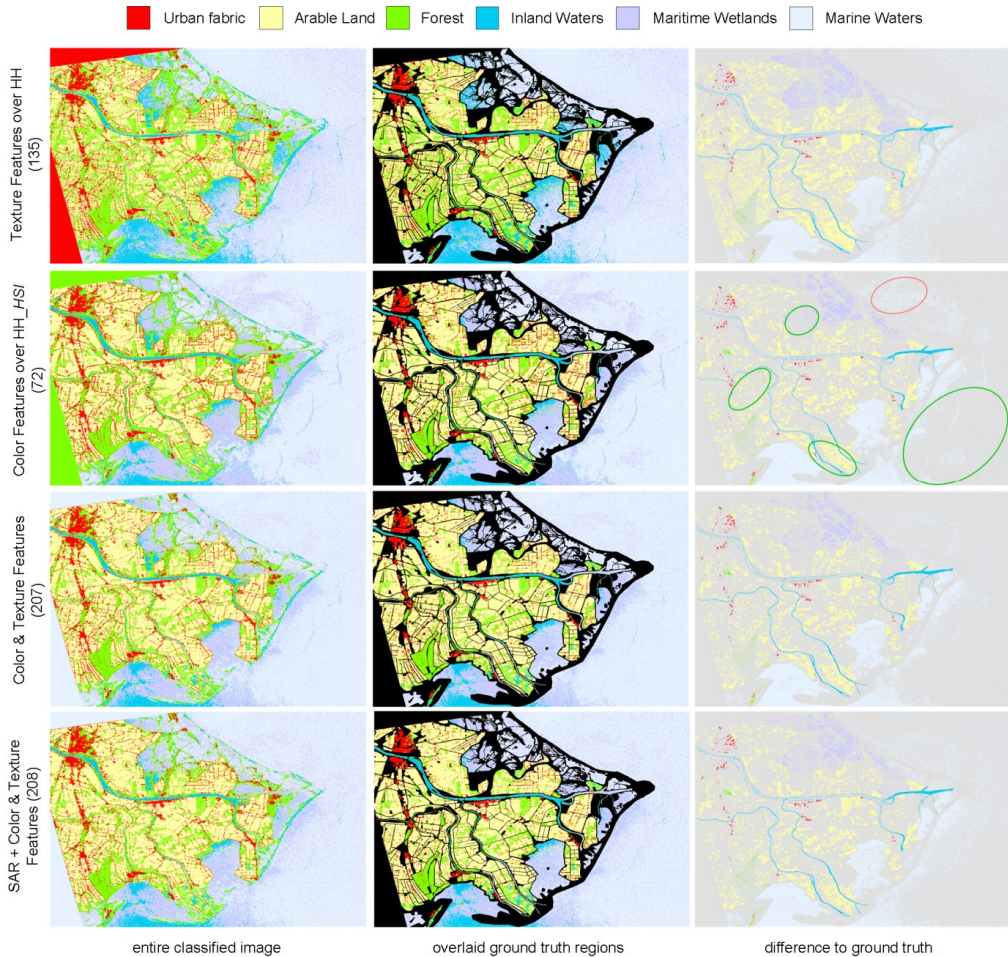
Figure 56 – Classification results for the single-pol COSMO-SkyMed image. Left column is the classified SAR image, the middle column is the classified image overlaid with the ground truth, and the right the difference to six-class ground truth to the middle column, respectively, and with the color indicating what the ground truth should have been. Circles mark degradations (red) and improvements (green).

3.5.   SUMMARY

This chapter introduced a new group of features to improve PolSAR LULC classification. In particular, the application of color features is addressed with their respective description and extraction over the generated pseudo color images for fully and partially PolSAR data. For fully polarimetric SAR data, the generation of pseudo color images is straightforward due to the full polarimetric information available in the scattering matrix [S]. The first evaluation was done over the pseudo color image in Pauli polarization basis (H,V) commonly used for visualization of PolSAR data.

The experimental results demonstrated that with the addition of the color features, all clas-sifiers outperformed the results with common PolSAR features alone as well as achieved higher classification accuracies compared to the traditional feature combination of PolSAR and texture features. We can conclude that the tested color features extracted over the pseudo color image in the Pauli H,V basis provide further improvements in terms of class discrimina-tion when compared to their texture counterparts. Texture features may still provide valuable discrimination and, as for the color features, the gains and losses achieved will vary with the PolSAR image classification problem. In the experiments, we have observed that the applica-tion of the color features to fully polarimetric SAR image classification can provide notewor-thy improvements in particular, man-made, foliage, and tree-type dominated vegetation ter-rain classes highly benefited from the additional color features.

The experiment and evaluations were extended to investigate the effects of pseudo color images generated by different combinations of polarization matrices and/or target decomposi-tion components. Experiments validated that individual visual color features alone are able to provide high discrimination particularly over pseudo color images generated using the polari-zation matrices. It could also be observed that the color features do work over intensity imag-es yet obviously not as effectively as over color images. However, they were still able to pro-vide comparable results in relation to results with the applied texture features, which might be related to particular terrain classes in certain classification setups.

Compared to fully PolSAR data, the generation of pseudo color images to extract color features is different for partially PolSAR image data as not the full polarimetric information is available to apply similar approaches. For dual-pol images, combinations of the two polariza-tion matrices assigned to the RGB color components provide a similar straightforward option to obtain such pseudo color images. However, this is obviously not possible considering sin-gle-pol images or one intensity channel. Thus, in such cases, an intensity to color conversion is usually applied.

In the experiment and evaluation of dual-pol and single-pol PolSAR images, color features yield noteworthy improvements, in particular the classification accuracies over the agricultur-al, forestry, and vegetation terrains are improved. Especially the extracted HSV and CSD col-or features were able to provide the highest discrimination over various terrain types. Over the dual-pol image, similar or better results may be achieved by extracting texture and/or color features from a single pseudo color image reducing the overall feature dimension rather than processing the two polarization matrices individually. As for the single-pol image, it is not expected that color features based on a color transformed intensity image will significantly outperform traditional texture features yet it could be observed that such color features can provide additional discrimination for classification of such image data.

Generally, for all color feature experiments regardless of the PolSAR data, we observed that additional texture features contribute to the classification performance and, as for the col-or features, the performance gains and losses they caused vary according to the terrain types and the SAR images over which they are extracted.

Chapter 4

Data-Driven and Adaptive Classification of PolSAR Images

A wide majority of remote sensing applications is in the areas of monitoring, detection, and classification of local or global regions and targets. Such analysis and interpretation tasks are generally performed with the help of automatic tools to support human experts and such tools have been introduced from various disciplines such as signal and image processing as well as computational- and machine learning domains. With the growing amount of available very high-resolution PolSAR image data nowadays, this obviously becomes a large-scale learning challenge, where such huge amount of data should be processed, analyzed, and in turn managed efficiently and effectively. By enabling this, the objective is to minimize the manual labor while achieving a highly accurate classification performance for human experts to analyze and interpret the data with utmost efficiency. The application of such methods is especially valuable if they are capable of processing and learning from large amount of data. Additionally, it is highly beneficial if they are able to adapt to the task at hand, which might change due to the introduction and availability of new data provided by the human expert.

The rest of the chapter is organized as follows. The next section briefly introduces the area of machine learning and its principles. To address the large-scale learning problem over the PolSAR data, Section 4.2 proposes the application of an adaptive and data-driven classification framework, namely *Collective Network of Binary Classifier*, as co-authored and published in [108]. The section presents CNBCs topology and design to adapt to the dynamic changes of the classification task at hand such as the integration of new ground truth data or features. Furthermore, this framework provides the capabilities to adapt to the data-driven classification of single PolSAR images as demonstrated in Section 4.3 as well as over PolSAR image collections as presented in Section 4.4 based on the author's publications [108], [199] and [197], [201], respectively.

4.1. THE PRINCIPLES OF MACHINE LEARNING

Machine learning concerns the design and study of systems that can learn from data. A more general definition of machine learning is given by Mitchell [182] where "a computer program is said to *learn* from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E." In the field of remote sensing SAR data, the task could be terrain or LULC classification within a SAR image, the experience would be the SAR data with the terrain class labels, and the performance is measured by the classification accuracy.

When designing such learning machines, the main objectives are to extract and generalize the *predictions* learned from underlying properties of the training data. Generally, the probability distribution of the provided training data is unknown so that learning machines try to model the distribution from the training data. If successful, such a generalized model enables accurate predictions over the unseen (test) data.

Machine learning distinguishes two main principles: *unsupervised* and *supervised* learning. In this context, supervision is provided as the prior information in form of class membership values (labels) to the given training data. This has its advantages and drawbacks for both learning principles, which are detailed next.

4.1.1. Unsupervised Learning

Learning is called unsupervised (UL) if no supervision takes place so that the data are given without any corresponding class labels. Hence, unsupervised approaches try to discover and model hidden structures in the data based on the underlying data properties or characteristics. Common approaches are blind source separation, self-organized maps and adaptive resonance theory, and data clustering. The latter is probably the most commonly known approach in unsupervised learning [60], [76].

There are various clustering methodologies exploiting different properties of the given data [60], [76]. Approaches are, for instance, density-based, hierarchical, centroid-based, and distribution-based clustering with the last two being the most widely used methods. As their names suggest, a centroid-based approach tries to find representative centers that best describe subsets of the data by minimizing the distance from the potential cluster centers to the data. The found cluster centroids represent the "learned" model. The most popular clustering algorithm is k-means [132] with numerous variations such as k-medoids [102] (centroid is member of the data) and fuzzy c-means [16] (allowing fuzzy cluster assignment). On the other hand, a distribution-based approach tries to define clusters where the corresponding data share the same underlying distribution. The most prominent method is (Gaussian) mixture models [138] using the expectation maximization method [52] to optimize iteratively a fixed number of (Gaussian) distributions to fit the data.

As mentioned before, unsupervised learning has the drawback that no prior information as class labels is available. On one hand, this is an advantage, as it does not require collecting

such ground truth information, which can be cumbersome. On the other hand, unsupervised methods do not generally perform as well as supervised approaches due to the lack of such ground truth information.

4.1.2. Supervised Learning

Within supervised learning (SL), the main objective is to infer a function from labeled training data. The training data generally consist of the data itself and the corresponding ground truth information indicating class memberships. The obtained inferred function is then used to predict class memberships from newly unseen data. Hence, the goal of a supervised learning algorithm is to be able to best generalize the underlying function from the training data.

While trying to infer the function or model that best fits the data, the bias-variance tradeoff ([17], Chapter 9) is a central problem in supervised learning and generalization. Intuitively, we try to find a model which best fits the data and at the same time generalizes well to unseen data. Models can have high bias, meaning they make inaccurate assumptions of the functions that can be learned (e.g., linear classifiers), as they have not enough flexibility, hence, the model is too "simple" to represent the relevant characteristics of the data. This is considered *underfitting*. Alternatively, models can have high variance, meaning they can learn various forms of complex functions, therefore, being too sensitive to the training samples and fitting irrelevant characteristics including noise in the training data. This is usually the case for complex algorithms with too many parameters, as they might tend to memorize the training examples without generalizing well resulting in *overfitting*. To achieve good performance on newly unseen data, an algorithm seeks to find a good tradeoff between bias and variance.

The bias-variance tradeoff is related to the number of available training data, the input feature dimensions, and noise within the features and training data. With regard to the size of the training data, if the underlying function is simple then a small amount of data will be sufficient to learn it. If, on the other hand, the function is complex then more data are necessary to be able to infer the true function. Additionally, the input feature dimension can also affect the learning algorithm. Especially when the input feature dimension is huge, yet only a few elements are actually relevant to model the true function. Thus, the presence of the irrelevant features makes it difficult for the learning algorithm to find the relevant features to learn the underlying function. Practically, feature selection [20], [79], [127], or dimension reduction [209] algorithms are used prior to applying a supervised learning algorithm. In the event of noisy training data, the learning algorithm should not try to find a function perfectly matching all training data as it would result in overfitting.

Overfitting is a general problem of single classifier systems. In the case of supervised learning, combining multiple classifiers to a committee or ensemble has shown to improve classification performance over single classifier systems [58], [158]. Typically, ensemble learning tries to improve generalization by combining multiple learners, therefore, reducing the chance of overfitting.

4.1.3. Active Learning

In machine learning, having enough data for a proper learning is not the main issue. Having the corresponding ground truth information to apply supervised learning is the challenge as generally the majority of the available data are unlabeled. Labeling huge amounts of data is tedious and time consuming. Therefore, reducing such labeling efforts and utilizing unlabeled data to improve the learner's classification performance is highly beneficial. One option in supervised learning is the so-called *active learning* (AL) [44].

AL starts with a supervised learning process by training a classifier over the initial training data. After obtaining the initial classification results, the idea is to provide the most confusing samples (e.g., samples close to or on a decision boundary among classes) to a human expert to obtain their class labels. The advantage is that these newly labeled samples can now be integrated into the learning process helping the classifier to find a better-generalized model of the data. Hence, it is beneficial to select examples among the most confusing samples, which will provide the best additional information into the learning process [189].

With active learning, data are actively labeled during the ongoing learning process, which comes with an obvious drawback. The human expert has to be available during this learning process to provide the ground truth information for the new samples, which might not always be practical or feasible option at all.

4.1.4. Semi-Supervised Learning

Active learning provides an iterative and interactive option to improve the learner's classification performance by incorporating previously unlabeled data to reduce significant training errors. To benefit from the unlabeled data without the human interaction, the interest in *semi-supervised learning* (SSL) has increased because it can combine supervised and unsupervised learning approaches. The general notion behind SSL is to start from a set of labeled data and then to utilize the large amount of unlabeled data to improve the initial learner [171]. Therefore, the crucial part in this process is, contrary to AL, the automatic selection of reliable training data among the unlabeled data so as to provide new training data to the ongoing learning process. The selection can be performed by several approaches such as unsupervised clustering methods [3], [80], self-training [231] where one initial learner iteratively selects the most confident samples to add them to the training set, or co-training [231] where two learners either work on different feature spaces or are completely different altogether and add new training samples to one another. Figure 57 illustrates the relation of typical supervised learning (SL) to SSL, which might encapsulate an UL approach preceding a SL process or a self-training method over a SL process.

To aid the selection of reliable new training data for SSL, several assumptions ([28], Chapter 1.2) are generally exploited where the first two are the most commonly used. Firstly, there is the *local smoothness / consistency assumption*, where nearby points are more likely to have the same label. This is the same assumption any SL algorithm exploits to learn from the training data and generalize a model or function applicable to any unseen data provided in the fu-

ture. Typically, this is performed in the feature space; however, it can also be applied *spatially* over the neighborhood of each image pixel. Secondly, the *global cluster assumption* exploiting the fact that points sharing the same structure, hence, would fall into the same cluster are likely to have the same label so that those unlabeled samples which are highly similar to a labeled sample should share its label. This includes approaches based on the *low-density separation* assumption where in many clustering methods the cluster centers are considered of high-density zones so that the decision boundaries should lie within regions of lower density. In this case, rather than to find the high-density sample regions directly, the focus lies in finding such low-density regions to best draw decision boundaries among clusters. As another assumption, the *fitting constraint* implies that a "good" classifier should not deviate too much from its initial label assignments during a learning process.
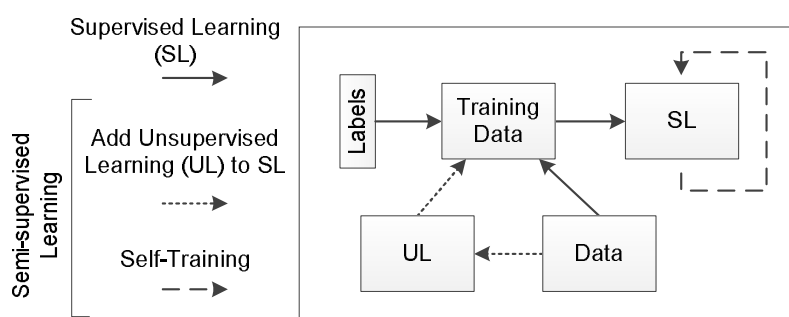
Figure 57 – Semi-supervised learning approaches with a typical supervised learning process.

4.1.5. Incremental Learning

A special form of supervised learning is *incremental learning* where the training data are not entirely available when the initial learning process starts. The training data might be acquired over time in sub sequential batches. Therefore, incremental learning approaches utilize the information in the newly presented data without forgetting their previously learned models and ideally do not re-use the previous training data. If all of these circumstances are fulfilled, such learning method is capable of incremental learning. Active Learning and Semi-Supervised Learning can also be seen as special incremental learning approaches since new training data become available sequentially.

A practical approach for learning from new data may involve discarding the existing classifier, and retraining a new one from scratch using all available data, previous and new. This, however, results in losing all previous acquired information so-called *catastrophic forgetting* [68]. Even though past knowledge could be restored due to the addition of the previous original training data, a serious drawback may still occur as the data might be lost, corrupted, or unavailable. Moreover, considering all available training data (new and previous) within every incremental training step will obviously result in an increase in computational complexity during retraining due to the growing amount of training data.

To avoid retraining a new classifier from scratch, an incremental learning process should be able to update the existing classifier and capable of adapting to the necessary structural changes. This can be the case if the newly available data change the existing classification problem by introducing previously unknown classes. However, an incremental algorithm should result in the same model after receiving n observations in multiple steps, as the common (batch-based) fundamental classifier does for the n observations; otherwise it might be an online algorithm, which is not incremental [168].

4.2. APPLICATION OF COLLECTIVE NETWORK OF BINARY CLASSIFIERS

For the classification of PolSAR images, several conventional approaches were proposed such as Maximum Likelihood Classifier and its variants, e.g., [2], [57], [86], [110], [119], [128], [170], [178], [204], [219], various types of Artificial Neural Networks [23], [29], [30], [49], [56], [70], [81], [82], [85], [95], [131], [175], [192], Support Vector Machines [31], [46], [175], [220], [226], [232], variations of Decision Trees and Random Forest [72], [162], [176], [220], [232], and unconventional approaches such as AdaBoost [72], [139], Multi-Modal Markov Aspect Model [218], Object Oriented Image Classification [45], and Conditional Random Fields [180]. While the focus is on supervised classification techniques, there are also recent unsupervised learning works that incorporate spatial proximity [63] or a spatial context model [223] beside the initial statistical modeling and segmentation to improve clustering and classification. The inclusion of such a spatial model may result in visually smoother and cleaner classification compared to simpler clustering techniques or pixel-based classification approaches on PolSAR images mainly due to the presence of speckle noise.

The application of supervised techniques with non-linear operators such as Artificial Neural Networks (ANNs) is quite popular, yet designing an optimal ANN for the problem at hand is a crucial and challenging task. For instance, an ANN with no or too few hidden nodes may not differentiate among complex patterns, instead leading to only a linear estimation of such – possibly non-linear – problem. In contrast, if the ANN has too many nodes/layers, it might be affected severely by the noise in the data due to over-parametrization, which eventually leads to a poor generalization or training. On such complex networks, proper training may be infeasible and/or highly time-consuming. The optimal number of hidden nodes/layers may depend on the input/output dimensions, training and test data sizes, more importantly the characteristics of the problem. In those works where a single (fixed) classifier is used, the overall performance directly depends on the choice of that classifier and its parameters. Furthermore, the feature set and the number of classes are usually kept as limited as possible not to cause the aforementioned feasibility problems on the training process due to the increased complexity and the well-known "curse of dimensionality" phenomenon. For this purpose, it is common to select only a certain subset of features while discarding the others or to apply feature dimension reduction techniques [209].

As most approaches apply a supervised training scheme, some of those support the methodology of incremental training. This means classifiers such as RBFs [151] and SVMs [97] are able to adapt their earlier decision-making capabilities using new training samples without suffering from catastrophic forgetting. However, besides the incremental training capabilities another important point is to make the actual structure of the classifier more scalable so as to adapt to possible dynamic changes such as new features, classes or (modified) ground truth that can be introduced to improve the classification accuracy at any time. It is obvious that static and fixed-structured classifiers cannot scale to such changes without discarding the existing classifier and retraining a new one from scratch using the ground truth accumulated so far.

An earlier ensemble of classifier type of approach, Learn++ [159], incorporates an ensemble of weak learners, which can perform incremental learning of new classes, however, albeit at a steep cost, i.e., learning new classes requires an increasingly large number of underlying learners for each new class to be learned. A modified version, the Learn++.NC [144], was designed to address this drawback more efficiently with a voting mechanism. The Resource Allocating Network with Long-Term Memory (RAN-LTM) [151] can avoid this problem by using a single RBF network, which can be incrementally trained by so-called "memory items" stored in a long-term memory. These items are selected from training samples while generating the network and are later combined with newly given training samples to suppress forgetting of the previous knowledge. However, RAN-LTM has a fixed output structure and thus, is not able to accommodate a varying number of classes. For the incremental learning problem when new classes are dynamically introduced, some hierarchical techniques [78], [97] have been proposed. They separate a single class from the previous classes within each hierarchical step, which builds up on its previous step. One major drawback of this approach is parallelization since the addition of N new classes will result in N steps of adding one class at a time. Furthermore, this does not support the possible removal of an existing class and hence requires retraining of the entire classifier structure. None of the ensemble of classier methods proposed so far can support feature scalability and thus a new feature extraction will eventually make the current classifier ensemble obsolete and require a complete re-design and re-training.

In order to address these limitations and drawbacks, we adopt a global framework structure, which is designed to seek for optimal classifier architecture for each distinct class type and feature set while utilizing a large set of features within. Specifically, we aim for the following objectives

I.   *Class Scalability*: Support for varying number of classes. Any class can dynamically be inserted into the framework without requiring a full-scale set-up and re-evolution.

II. *Feature Scalability*: Support for varying number of features. Any feature can be dynamically integrated into the framework without requiring a complete re-design.

III. *Evolutionary Search*: Seeking for the optimum network architecture among a collection of configurations (the so-called Architecture Space, AS).

IV. *Evolutionary Update in the AS*: Keeping only the *best* individual configuration in the AS among indefinite number of evolution runs.

V. *High efficiency* for the evolution (or training) process: Using as compact and simple classifiers as possible in the AS.

VI. *Online (incremental) Evolution*: Continuous online/incremental training (or evolution) sessions can be performed to improve the classification accuracy.

VII. *Parallel processing*: Classifiers can be evolved using several processors working in parallel in local or cluster environment.

Our goal is to achieve these objectives for a database (i.e., collection) of similar PolSAR images. For this, the Collective Network of Binary Classifier (CNBC) [106] is adapted as a PolSAR image classification framework, which can adjust its structure to any change in the database by means of incremental evolution sessions. During these sessions, each classifier in the framework structure evolves to improve their classification accuracy. In short the CNBC, if properly evolved, shall learn the significance (or the discrimination power) of each feature and its individual components. The framework is developed over a dedicated application with a user-interface (UI) where the user can define new classes, or update the existing ones, while specifying the ground truth data over a given PolSAR image.

4.2.1. Collective Network of Binary Classifiers – The Topology

In order to accomplish the aforementioned objectives, the most important arguments for a data-driven and adaptive classification framework are its easy scalability and extendibility in the input and output dimensions of the underlying classification problem. Therefore, CNBCs adapts the concept of "divide and conquer" ([18], Chapter 7) splitting a massive learning problem into many smaller tasks, which are easier and faster to solve. CNBC applies this concept in two levels. Firstly, the overall topology follows the One-versus-All (OvA) approach where a multiclass problems is divided into multiple binary classification problems. In this particular case, binary classification is considered per class, therefore, separating the individual class decisions from one another. This allows adding and removing classes providing the desired class scalability as stated by objective I. Secondly, for each individual class within the OvA approach, a decision is made by adapting the concept of stacked generalization [215], which provides the flexibility of the second objective: feature scalability. Additionally, by accomplishing these two objectives, the *parallel processing* objective can be fulfilled in a straightforward manner. Figure 58 presents the overall topology of the CNBC framework illustrating the "divide and conquer" concept in the two levels, per class and per feature.

4.2.1.1 Network of Binary Classifier (NBC)

An individual class decision is provided by a *network of binary classifiers* (NBC). An NBC corresponds to a unique class and shall contain varying number of *binary classifiers* (BCs) in the input layer. Each BC performs binary classification discriminating the class of this particular NBC with a unique feature (or sub-feature) based on their representative feature vector

(*FV*). Therefore, whenever a new feature is integrated, its corresponding BC will be created, evolved, and inserted into each NBC, yet keeping each of the other BCs "as is". On the other hand, whenever an existing feature is removed, its corresponding BC is simply removed from each NBC in the system. In this way, scalability with respect to any number of features is achieved and the overall system can avoid re-evolving from scratch. Similar to the stacked generalization, each NBC has a *fuser* BC in the output layer, which collects and fuses the binary outputs of all BCs in the input layer and generates a single binary output, indicating the relevancy of each feature to the NBC's corresponding class. Due to the OvA approach, whenever the user defines a new class, a new NBC can simply be created (and evolved) without requiring any need for change or update on the other already existing NBCs. This way the overall system dynamically adapts to user demands for varying number of classes.
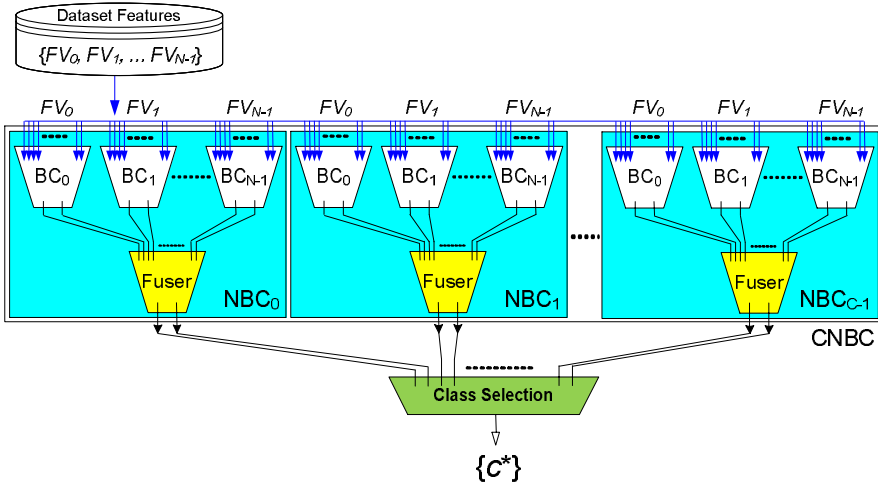
Figure 58 – Topology of the CNBC framework with *C* classes and *N* features (*FV*s).

4.2.1.2 Evolutionary Binary Classifier (BC)

Binary classification over the employed (sub-) features in each NBC is done over a set of *evolutionary Binary Classifiers* (BCs), where each BC shall learn in time the significance of individual feature components for the discrimination of its class. The main motivation of the evolutionary component is to avoid a static approach with the classifier configuration being fixed for all features. Therefore, each BC is optimally chosen within a pre-defined set of classifier configurations, called *architecture space*. The optimality therein can be set with a user-defined criterion. Two common ANN types, the Multi-Layer Perceptron (MLP) and the Radial Basis Function (RBF) network are mainly considered as BCs. Besides the exhaustive search with numerous runs of backpropagation (BP), the recently proposed multi-dimensional Particle Swarm Optimization (MD-PSO) [107] is employed as the primary evolutionary technique. Note that any other classifier types such as Support Vector Machine and Random Forest can also be used within this framework as the BC type. However, to utilize the full framework potential, they should be able to support the principle of incremental learning.

As shown in Figure 58, due to the applied "divide and conquer" concept, it prevents the need of using complex classifiers within the BCs as the performance of both training and evolution processes degrades significantly as the complexity rises due to the "curse of dimensionality". A major benefit of this approach with respect to efficient training and evolution process is that the configurations in the architecture space can be kept as compact as possible avoiding unfeasibly large storage and training time requirements. This is a significant advantage especially for the training methods performing local search, such as BP, for ANNs since the amount of deceiving local minima is significantly lower in the error space for simple and compact architectures. Furthermore, when BP is applied exhaustively, the probability of finding the optimum solution is significantly increased.

4.2.2. Initial Training and Evolution

In general, CNBC applies the supervised learning principle with the required training dataset[1] given by individual feature vectors (*FV*s) and their corresponding ground truth as target class vectors (*CV*s). The training and evolution of the entire CNBC is always performed in an *evolution session* for each NBC individually with a two-phase operation. In the first phase for each NBC, there are a varying number of evolutionary Binary Classifiers (BCs) in its input layer, depending on the number of features employed using their *FV*s and *CV*s. During the second phase, the individual BC outputs are merged by the fuser BC to obtain the final NBC class decision. Each classifier (BC) in the CNBC body is subject to *evolution*, which is defined by seeking the best (optimal) configurations of the underlying classifier to improve the overall classification accuracy of the CNBC.

4.2.2.1 Evolutionary Update in the BCs

Recall that each BC performs binary classification over a single (sub-) feature. Rather than limiting the BC to one specific configuration of a particular binary classifier, we employ an *architecture space* (AS), which can be defined over any type of binary classifier with any configuration properties such as number of hidden layers and number of neurons per hidden layer for ANNs. Therefore, rather than simply training one single configuration in every BC, the optimal configuration is sought among a collection of configurations in the defined AS by a search called *evolutionary update*. For any applied evolution technique, e.g., MD-PSO, in order to improve the probability of convergence to the global optimum, several *evolutionary runs* are performed within an evolutionary update where N_R is the number of runs and N_C is the number of configurations in the AS. For each evolutionary run, the objective is to find the best configuration within the AS with respect to a pre-defined criterion (e.g., training/validation mean squared error (MSE) or classification error, CE). Note that along with the best classifier, all other configurations in the AS are also subject to evolutionary update and, hence, they are continuously (re-) trained with each evolutionary run. Thus, during this ongo-

---

[1] As a common term, "train(ing) dataset" is used to refer to the dataset over which the CNBC is evolved.

ing process between two consecutive evolutionary runs, any classifier configuration can replace the current best one in the AS if it surpasses the pre-defined criterion.

Figure 59 demonstrates an evolutionary update over a sample ANN AS containing five MLP configurations. The tables in Figure 59 show the training MSE, as the criterion to select the optimal configuration at each run. The best evolutionary runs for each configuration are highlighted and the best configuration in each run is marked with '*'. Note that at the end of the three runs, the overall best network with MSE = 0.10 has the configuration 15×2×2 and, hence, is used as the classifier for any classification task until any other configuration surpasses it in a next evolutionary run during an evolutionary update. In this way, each BC configuration in the AS may *evolve* to a better state, which is the main purpose of this evolutionary update mechanism.
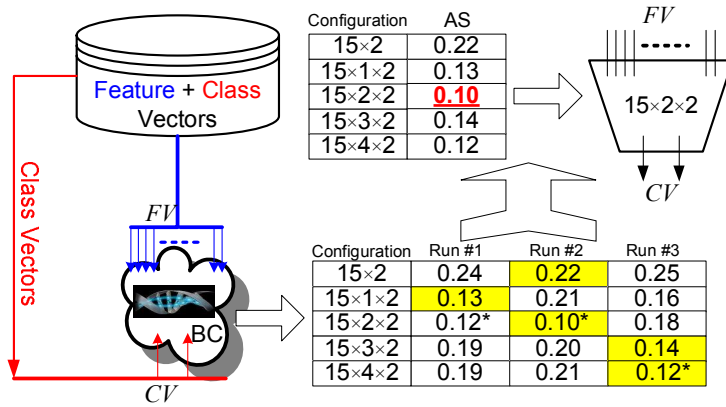
Figure 59 – Evolutionary update in a sample architecture space for MLP configuration arrays $R_{min}=\{15,1,2\}$ and $R_{max}=\{15,4,2\}$ where $N_R = 3$ and $N_C = 5$. The best runs for each configuration are highlighted and the best configuration in each run is marked with a '*'.

4.2.2.2 Phase 1: Evolving the BCs

The training and evolution process of each BC in an NBC is performed within the current architecture space (AS) to find the best (optimal) BC configuration with respect to a given criterion. In this first phase, the BCs of each NBC are evolved given an input set of *FV*s and a target *CV* as illustrated in Figure 60. Recall that each *CV* is associated with a unique NBC and the fuser is not needed in this phase. Once an evolution session is over, the AS of each BC is then recorded and can be used for potential (incremental) evolution sessions in the future. Due to applying the evolutionary update rule as described in the previous Section 4.2.2.1, we achieve the desired objectives III and IV.

4.2.2.3 Phase 2: Evolving the Fuser BC

Recall that each evolution process may contain several runs (e.g., 5 to 10) and according to the aforementioned evolutionary update rule, the best configuration achieved will be used as the classifier. Hence, once the evolution process is completed for all BCs in the input layer (Phase 1), the best BC configurations are used to propagate all *FV*s of the items in the training

dataset to compose the *FV* for the fuser BC from their output *CV*s, to evolve the fuser BC in the second phase. Apart from the difference in the generation of the *FV*s, the evolutionary method (and update) of the fuser BC is the same as any other BC has in the input layer. In this phase, the fuser BC *learns* the *significance* of each individual BC (and its feature) for the discrimination of that particular class. This can be viewed as the adaptation of the entire feature space to discriminate a specific class in a large dataset. In other words, a crucial way of applying an efficient *feature selection* scheme as some *FV*s may be quite discriminative for some classes whereas others may not and the fuser, if properly evolved and trained, can *weigh* each BC (with its *FV*), accordingly. In this way, the usage of each feature (and its BC) shall optimally be *fused* according to their discrimination power for each class.
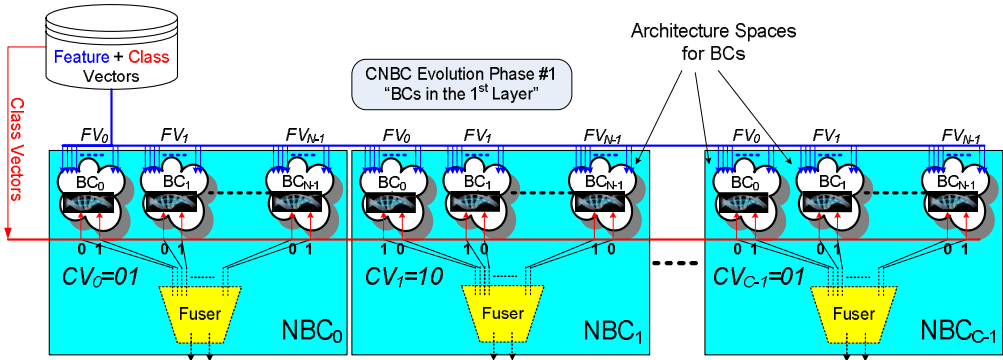
Figure 60 – Illustration of the phase 1: evolution session over BCs' architecture spaces in each NBC as the evolution of the BCs with the feature (*FV*) and class (*CV*) vectors.
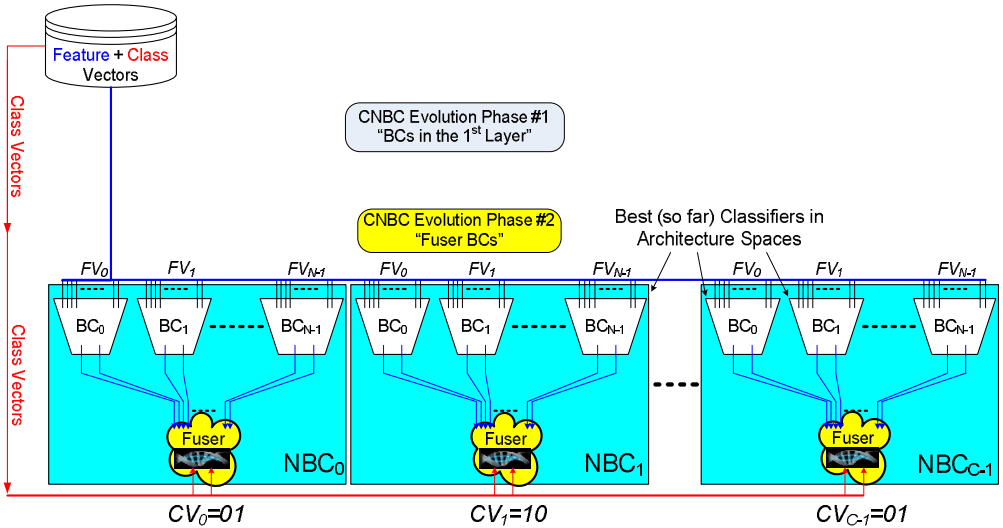
Figure 61 – Illustration of the phase 2: evolution session over BCs' architecture spaces in each NBC as the evolution of the fuser BCs with the actual BC outputs and the class vectors (*CV*s) are shown.

4.2.2.4 Handling Unbalanced Training Data

Due to the applied OvA approach, a common assumption to generate the individual training sets for each class (NBC) is that a positive sample of one class can be used as a negative sample for all others. Yet if the user defines many classes, the *unbalanced* numbers of positive and negative samples per class, may cause a bias problem. For every positive sample, there will be numerous negative samples, which may bias the classifier. To prevent this, a negative sample selection is performed in such a way that for each positive sample, the number of negative samples (per positive sample) will be limited according to a pre-determined positive-to-negative ratio (PNR). Selection of the negative samples is performed with respect to the closest proximity to the positive sample so that the classifier can be evolved by discriminating those negative samples (from the positive sample) that have the highest potential for the false positive. An example on a toy classification problem is presented in Figure 62. Therefore, if properly trained, the classifier can draw the "best possible" boundary between the positive and (PNR number of) negative samples, which shall in turn improve the classification accuracy.
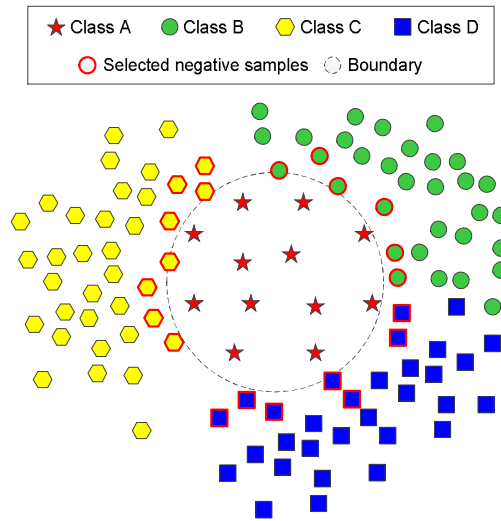
Figure 62 – Toy example of negative sample selection for Class A against three other classes using positive-to-negative ratio (PNR).

4.2.3. Incremental Evolution

To accomplish the objective VI – Online Evolution, the CNBC framework is designed for continuous *incremental evolution sessions* where each session may further improve the classification performance of each NBC using the advantage of the evolutionary updates for the underlying BCs. Bear in mind that CNBC and its NBCs with their BCs are *evolved* and only the particular configurations within the architecture space undergo (*re-) training*.

In incremental evolution sessions, the main difference between the initial and the subsequent evolution sessions is the initialization of the process. The former uses *random* initialization for the underlying BC configurations (e.g., weights of ANNs) whereas the latter starts from the last AS parameters of each classifier in each BC as a starting point for the evolutionary update. Note, all configurations in the AS of a BC will be trained with the new ground truth data; and the training dataset used for the incremental evolution sessions may be different from the previous ones. Thus, before the evolutionary update takes place it compares the performance of the last recorded and the current (after the last evolutionary run) BC over the current training dataset. As a result, any BC configuration, which surpasses the previous best configuration, may emerge and thus be used afterwards for the classification task.

The evolution of an existing CNBC is three-fold and can be performed by

- incremental evolution sessions on top of the existing NBCs when new ground truth data are available for them (the existing classes),

- the initial evolution session when a new class is introduced and its NBC has just been created,

- updating / adapting evolution sessions when new features are introduced into an existing CNBC.

4.2.3.1 New Training Samples

As mentioned in Section 4.1.5, training data do not necessarily be completely available when initiating the training process. For the case an existing CNBC is given a new batch of training data including additional samples for a known class NBC, the following procedure is applied to verify if the existing NBCs require an update by retraining their underlying BCs. Note that in this case, the topology of the existing CNBC stays untouched.

NBCVerify ($\underline{S}$)

➢ For $\forall NBC_c \in CNBC$ do:

- ○ Separate sample set $\underline{S}$ into positive S_c^+ and negative S_c^- samples

- ○ Classify S_c^+ with NBC_c

- ○ Classify S_c^- with NBC_c

- ○ If NBC_c fails to classify majority of either subset correctly
    → mark NBC_c for updating

➢ End loop.

The main idea is to check if the current NBCs are able to classify the new data correctly. If so then no update is necessary. Otherwise, incremental evolution is performed to improve those NBCs using the previous and new training data.

4.2.3.2 New Classes

Let us consider a training batch that contains samples related to an unknown class. In this case, the CNBC requires a new NBC in the existing topology to learn and hence discriminate the new class. Additionally, this might further require incremental evolution sessions over

some of the existing NBCs if the new data cannot be classified correctly by them (validated by `NBCVerify`) as described in the previous Section. The NBCs for the new classes are obviously due for evolution without such verification. Figure 63 illustrates an example where an existing CNBC with three classes (blue) is updated to four classes by the addition of one new NBC (red). This is straightforward due to the adopted "divide and conquer" paradigm.
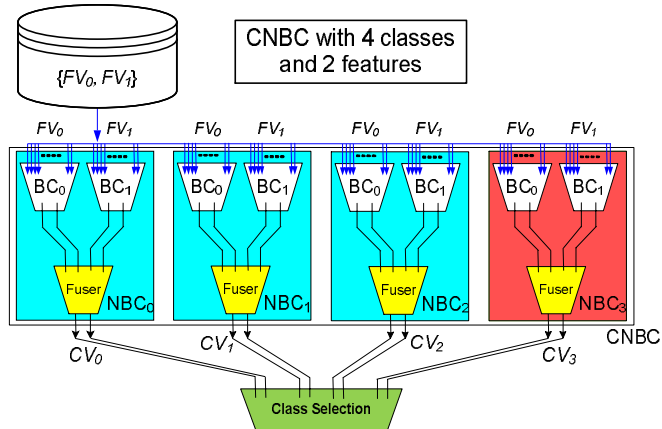
Figure 63 – Incremental evolution of the CNBC topology where training samples include new class ground truth.

4.2.3.3 New Features

In case of integrating new features to an existing CNBC body, it is more a topology update rather than an incremental learning process. For each new feature, one new BC is inserted into the input layer of every NBC as shown in Figure 64. The sample CNBC (four classes, two features) from now becomes a CNBC with four classes and four features. Then each new BC is subject to an initial evolution using the evolutionary update rule. Due to these changes in the input layer of an NBC, the fuser BC in each NBC has to be updated to accommodate the changes as detailed in Section 4.2.2.3.
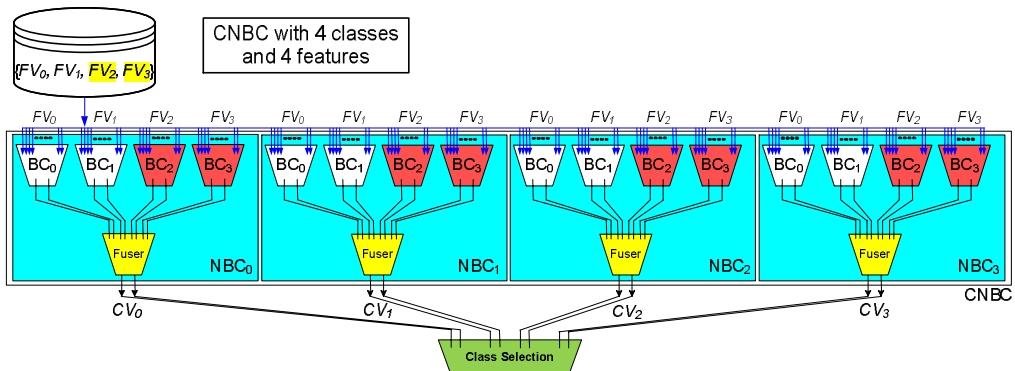
Figure 64 – Incremental evolution introducing new features to an existing CNBC topology.

4.2.4. Classification

In order to obtain the intermediate and final CNBC classification output, we apply a dedicated class selection technique using *1-of-n* encoding scheme in all BCs and NBCs, therefore, their respective output layer size is always two. Let $CV_{c,1}$ and $CV_{c,2}$ be the first and second output of the cth class vector (*CV*). The class selection in *1-of-n* encoding scheme can simply be performed by comparing the individual outputs, e.g., say a positive output if $CV_{c,2} > CV_{c,1}$, and vice versa for negative. This is also true for the fuser BC, the output of which makes the output of its NBC. *FV*s of each dataset item are fed to each NBC in the CNBC. Each *FV* is propagated through its corresponding BC in the input layer of the NBC. The outputs of these BCs are then fed to the fuser BC of each NBC to generate all *CV*s. The class selection block as for example shown in Figure 64 collects them and selects the positive class(es) of the CNBC as the overall outcome. This selection scheme, first of all, differs with respect to the dataset class type, where the dataset can be called as *uni-class*, if an item in the dataset can belong to *only* one class, otherwise called as *multiclass*. Therefore, in a uni-class dataset there must be *only* one class, the c^*, selected as the positive outcome whereas in a multiclass dataset, there can be one or more NBCs, $\{c^*\}$, with a positive outcome. In the class selection scheme the *winner-takes-all* strategy is utilized. Assume without loss of generality that a *CV* of {0, 1} or {-1, 1} corresponds to a positive outcome where $CV_{c,2} - CV_{c,1}$ is maximum. Therefore, in uni-class datasets, the positive class index, c^*, ("the winner") is determined as follows:

$$\{c^*\} = \arg \max_{c \in [0, C-1]} \left(CV_{c,2} - CV_{c,1} \right). \tag{19}$$

In this way the erroneous cases (false negatives and false positives) where no or more than one NBC exists with a positive outcome can be properly handled. However, for multiclass datasets the *winner-takes-all* strategy can only be applied when no NBC yields a positive outcome, $CV_{c,2} \leq CV_{c,1} \quad \forall c \in [0, C\text{-}1]$. Otherwise, for a set of *FV*s belonging to a dataset item, multiple NBCs with positive outcome may indicate multiple true-positives, which makes further pruning inapplicable. As a result, for a multiclass dataset the (set of) positive class indices, $\{c^*\}$, is selected as follows:

$$\{c^*\} = \begin{pmatrix} \arg\max_c \left(CV_{c,2} - CV_{c,1} \right) & \text{if } CV_{c,2} \leq CV_{c,1} \\ \left\{ \arg \left(CV_{c,2} > CV_{c,1} \right) \right\} & \text{else} \end{pmatrix} \quad \forall c \in [0, C-1]. \tag{20}$$

4.2.5. Experimental Evaluation

As a starting point for the experimental evaluation of CNBC, the same experimental setup of the previous color feature evaluation in Section 3.3 is used. Therefore, details regarding the classifiers, PolSAR images, and feature sets are similar as referred to in Sections 3.3.1 to 3.3.3. Here, changes and modifications made to the previous setup are described.

As for the classifiers, the combinations of multiclass topologies OvA and OvO with classifiers RF and SVM are extended by CNBC and feed-forward Artificial Neural Networks (ANNs), specifically Multi-Layer Perceptrons (MLP), as they are still a frequent choice as

classifiers within SAR LULC classification [4], [23], [29], [30], [49], [70], [104], [175]. Whenever MLPs are chosen as binary classifiers (BCs) within the CNBC, two different training/evolution methods are used namely the traditional backpropagation algorithm (BP) and the multi-dimensional particle swarm optimization (MD-PSO) as applied to evolutionary artificial neural networks (EANN) [107]. This allows evaluating the CNBC topology and application of different classifiers as underlying BCs with the previously obtained results in Section 3.3.4. As a summary, Table 4 lists the nine classification combinations used within the experiments.

Table 13 – Classifiers employed within the CNBC experimental setup.

Classifier abbreviation	Description
CNBC_MLP_BP	CNBC using MLP trained with BP
CNBC_MLP_PSO	CNBC using MLP evolved with MD-PSO
CNBC_RF	CNBC using RF
CNBC_SVM	CNBC using SVM
OvA_RF	One-versus-All using RF
OvA_SVM	One-versus-All using SVM
OvO_RF	One-versus-One using RF
OvO_SVM	One-versus-One using SVM
sANN_MLP_BP	MLP trained with BP

Within CNBC, the application of the EANN finds the weights of an ANN and simultaneously seeks for the *optimal* configuration in the architecture space. We shall also adapt similar strategies for all other used classifiers. This means, for classifiers such as SVM and RF, a sequential search over a pre-defined architecture space with multiple (parameter) configurations is performed and the best one will be used for testing the classification performance.

Whenever MLPs are considered as BCs within the CNBC, the predefined architecture space includes the single layer network $[N_{in} \times N_{out}]$ and 11 MLPs from $[N_{in} \times \{5\text{-}15\} \times N_{out}]$. For the standalone MLP, the architecture space consists of the single layer network $[N_{in} \times N_{out}]$, 11 one-layer perceptrons $[N_{in} \times \{10\text{-}20\} \times N_{out}]$, and 66 two-layer perceptrons $[N_{in} \times \{10\text{-}20\} \times \{5\text{-}10\} \times N_{out}]$. This is a large extension compared to the CNBC, however, necessary due to the more complex in- and outputs. For training, BP parameters are set as follows: the learning parameter is $\eta=0.005$, the iteration number is set to 1000, and as the activation function, the typical hyperbolic tangent is employed. MD-PSO parameters are set empirically as follows: swarm size is 100, $V_{max}=0.2$, $VD_{max}=10$ and the termination criteria are 1000 iterations and cutoff error $\varepsilon_c=10^{-4}$.

As in the color experiments' setup in Section 3.3.1, the same procedures are used to optimize the SVM's kernel type and respectable parameters, and similarly, search for the optimal number of trees within RF and feature split per tree. It is worth mentioning here that all configurations in an architecture space are enumerated with a hash function, which ranks the con-

figurations with respect to their complexity, e.g., associates higher hash indices to an individual ANN or RF with a higher complexity. This means that while sequentially searching for the best parameters / configurations for any classifier type, it will always favor less complex configurations as long as a similar performance level is achieved.

Table 14 – Different feature set combinations of PolSAR and image processing features considered for CNBC classification experiments.

Feature set	Feature vector	Dim.	Description
FS_1	FV_1	12	Elements from $\langle [T] \rangle$ and $\langle [C] \rangle$
FS_3	$FV_1 + FV_2$	23	Combination of FS_1 plus components of $H\alpha A$ and eigenvalue decomposition, RVI
FS_6	$FV_1 + FV_2 + FV_3$ $+ FV_4 + FV_5$	46	Combination of FS_3 plus components from target decompositions
FS_6^T	$FV_{1\text{-}5} + FV_6 + FV_7$ $+ FV_8 + FV_9$	127	Combination of FS_6 plus extracted texture features
FS_6^{CT}	$FV_{1\text{-}9} + FV_{11}$ $+ FV_{12} + FV_{13}$	187	Combination of FS_6^T plus extracted color features

The same features are applied as in the previous color experiments in Section 3.3. However, the feature sets are generated in such a manner that a previously used feature set is augmented with a new FV. Therefore, adding new FVs will systematically increase the feature set dimension. The generated feature sets (FSs), their included feature vectors, and total dimensions are listed in Table 14. The standalone MLP, OvO, and OvA topologies will take the entire feature set directly as the input. For the CNBC, the FS is split into predefined sub-features, namely the feature vectors, FV_n, as given in Section 2.6 and 3.2.5, and summarized in Table 14. Therefore, FS_1, FS_6, and FS_6^{CT} will generate one BC, five BCs, and 12 BCs, respectively. Note that as for the color experiments, the texture and color features are extracted over the Pauli color-coding (H,V) representation. Furthermore, a positive-to-negative ratio, PNR = 10, is used for negative sample selection for all classifiers, if necessary.

The classification results for the nine applied classifiers over the three PolSAR images (*Flevo_L*, *Flevo_C*, and *SFBay_C*) are presented in Figure 65; and results for *Flevo_L* are evaluated first before moving to the two RADARSAT-2 images.

The *Flevo_L* image is evaluated separately as it is a more complex classification task with its 15 classes. As a general observation, it is anticipated that classification results for the individual classifiers increase (3-8%) with the integration of new features such as $H\alpha A$ in FS_3 or components from various target decompositions in FS_6. Yet the improvements from FS_3 to FS_6 are smaller (2.5%) compared to the $H\alpha A$ addition (6%). However, the average difference in classification accuracies between FS_1 and FS_6 is 7.5%. This is related to the target decom-

positions components in FS$_3$ and FS$_6$ as they generally try to provide similar kinds of discrimination properties regarding the basic scattering mechanics.
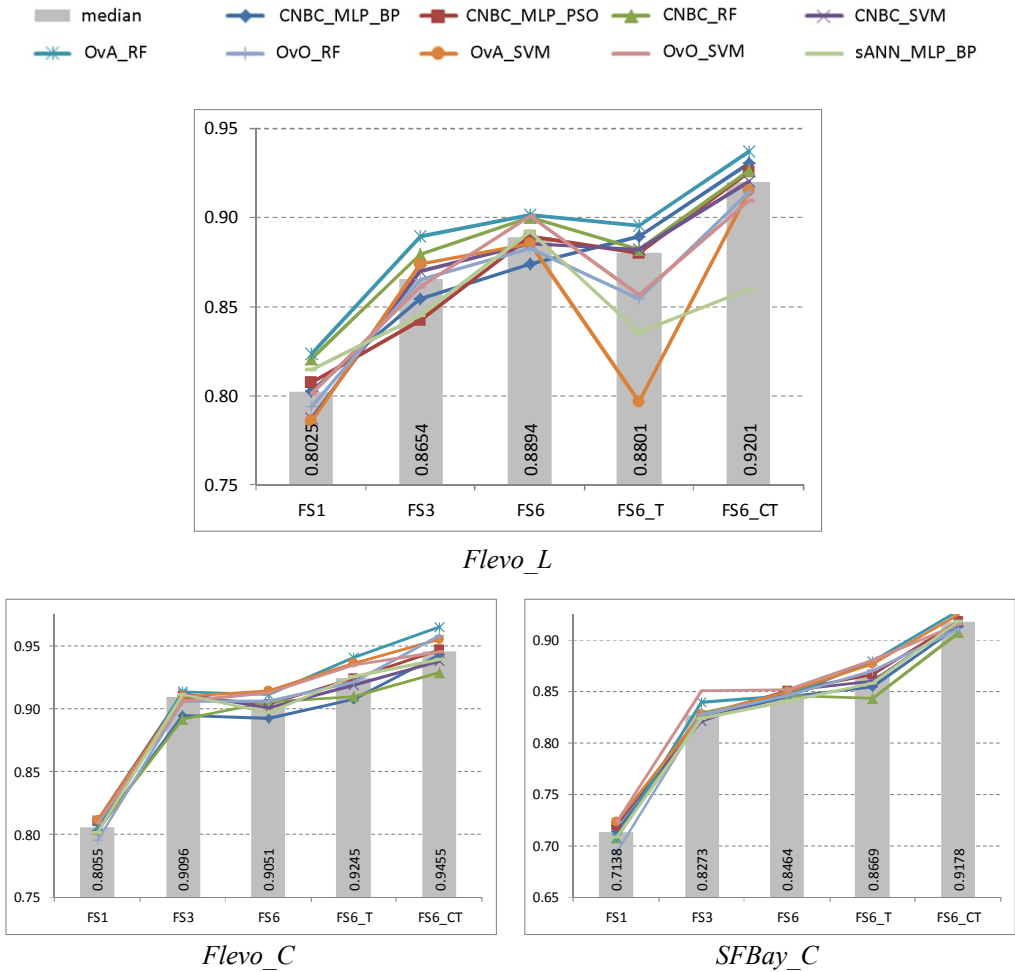
Figure 65 – Classification results of nine different classifiers for an increasing feature set over *Flevo_L*, *Flevo_C*, and *SFBay_C*. See Table 13 for classifier abbreviations.

After augmenting FS$_6$ with texture features, we can notice a significant drop in classification accuracies for OvA_SVM (down to FS$_1$ result), and sANN, OvO_RF, OvO_SVM (all down to FS$_3$ their results). The reason for these performance drops is that all of these classifiers use the entire FS as a single input. As discussed earlier in Section 3.3.4, this is probably due to the larger number of classes, majority of which being agriculture fields, thus no significant texture difference exists among them. Furthermore, such a low-resolution SAR image may not allow for major textural discriminations and the applied speckle filtering may further degrade the texture information. Therefore, such a degraded and fairly similar texture information among most of the classes rather introduce a "confusion" and makes it difficult to

learn individual class patterns when the majority of features are quite similar and noisy for many classes. Recall that the texture features have an overall dimension of 81 compared to the PolSAR features having a dimension of 46 for FS_6; hence, texture features will have a rather dominating influence. RF can better cope with this effect than SVM as little numbers of features are randomly selected within each tree in the forest, which in turn, will negate the influence of the large texture feature dimension. However, we can observe that RF is quite affected by the OvO multiclass topology due to its pairwise classification. Similarly, it can be noticed that the results using CNBC are not affected or only by minor drops due to the applied "divide and conquer" concepts to the features per class. In case of CNBC using MLP with BP (CNBC MLP_BP), the classification accuracies even increases with the integration of texture features. This MLP performance is below the median for FS_3 and the lowest for FS_6 with 2% shy of the median accuracy. With the integration of texture features, CNBC MLP_BP is able to achieve the same accuracy as the median in FS_6. While other classifiers drop because texture features do not provide additional discrimination to FS_6, the CNBC MLP_BP combination can benefit from the texture features yet just achieving the same classification accuracy most other classifiers obtained with just FS_6.

As previously observed by the color experimental results in Section 3.3.4, color features are able to provide an additional level of discrimination in this classification task of *Flevo_L*, increasing the median results by 3% compared to FS_6. All OvA, OvO, and CNBC type classifiers provide the best classification results exceeding their FS_6 and FS_6^T results, only the standalone MLP exhibits a classification accuracy below 89 percent with FS_6, i.e., a drop by 1% due to the increased feature set dimension.

Evaluations and observations regarding images *Flevo_C* and *SFBay_C* show only minor differences among topologies and used classifiers for the various feature set augmentations. The main reason is that both of them are simpler classification tasks with four and five classes of major terrain categories. In case of *Flevo_C*, results using FS_3 and FS_6 are pretty much the same due to the simpler four-class problem and the target decompositions in FS_6 do not yield more diversity among the features. Here, classes have unique texture pattern so that the texture features contribute discrimination among classes as previously observed, when adding the color features, all classifiers perform on similar level, achieving 4% higher accuracies than with FS_3 and FS_6. It can be noticed that CNBC results are at the low end of the accuracy range performing slightly lower (1%) than OvA due to the employed feature splitting per class. This is particularly the case for results obtained with CNBC and RF combination for this classification task. Recall that CNBC adopts a "divide and conquer" approach (assigning individual feature vectors to each binary classifier) and on top of that, RF employs its random feature selection over these individual feature vectors. By just selecting randomly a few feature elements from the individual FVs, there will be loss of information as the FV elements can be closely related with each other. The same effect can be observed for the addition of the texture features to FS_6. The difference can be seen when picking the random features from the entire feature set, as for OvA with RF, where no feature set division occurs, yet providing the highest accuracy among all classifiers. For the *SFBay_C* image, we can notice less diversity

among the classifiers so that most of them perform on a similar level. Overall, the same observations as for *Flevo_C* can be made when adding texture and color due to the small number of terrain classes, which are quite distinct.

4.3. CLASSIFICATION OF SINGLE POLSAR IMAGES

As for the application of CNBC, we demonstrate classification experiments over a single PolSAR image focusing on two use cases. The first case is sample-based incremental training where erroneous classification results are corrected after visual inspection, and the second case is class-based incremental training where new training data include unknown terrain classes. In both cases, the application of active learning is one scenario how the new training data are obtained. After visual inspection of the initial classification results, a user provides new training samples for the erroneous regions over the existing terrain classes. In the class-based case, this further includes the addition of new classes due to newly made observation or gained terrain knowledge. These two use cases are simulated in experiments over the San Francisco Bay (SFBay) AIRSAR image, which was preprocessed using speckle filtering as recommended by Lee et al. [117] over a 5×5 window.

4.3.1. Sample-Based Incremental Training

The focus is on the properties of the CNBC framework so that the used features are not regarded critical especially since visual classification results are considered. A traditional set of features is chosen consisting of elements from $\langle [T] \rangle$ and $\langle [C] \rangle$ combined with components of $H\alpha A$ and eigenvalue decomposition. Additionally, three complex correlation coefficients (ρ_{12}, ρ_{13}, ρ_{23}) between scattering matrix terms are derived. This results in a feature set of 28 dimensions with three sub-feature vectors (FVs):

$FV_1 = [T_{11}, T_{22}, T_{33}, C_{11}, |C_{12}|, \angle C_{12}, |C_{13}|, \angle C_{13}, C_{22}, |C_{23}|, \angle C_{23}, C_{33}]$,

$FV_2 = [Span, H, A, \bar{\alpha}, \bar{\beta}, \bar{\delta}, \bar{\gamma}, \lambda_1, \lambda_2, \lambda_3]$,

$FV_3 = [|\rho_{12}|, \angle \rho_{12}, |\rho_{13}|, \angle \rho_{13}, |\rho_{23}|, \angle \rho_{23}]$.

The CNBC created and evolved contains five NBCs, which is equivalent to the number of the five pre-defined classes for AIRSAR SFBay image. Recall that each NBC in the CNBC contains a certain number of BCs in the input layer, which is equivalent to the number of *FV*s. Therefore, each NBC has four BCs, i.e., three in the input layer for the applied feature vectors + one fuser BC to merge the feature BC outputs. Thus, a total of 4×5=20 classifiers are individually evolved using the major ANN type, MLP with MD-PSO and exhaustive BP [107] as the two *classifier and evolution types* (*CETs*) presented in Table 15. Furthermore, note that for each BC, its input size N_{in} is determined by the size of its FV_n, i.e., 12, 10, and 6 for the input layer BCs of all NBCs and naturally, 3×2 = 6 for all fuser BCs.

Table 15 – Classifier and Evolution Types (*CETs*) used within the CNBC experiments over a single PolSAR image.

CET-1	CET-2
MLP trained with Exhaustive BP	MLP trained with MD-PSO

Table 16 – The architecture space, of size $N_C = 17$, used for MLPs with one Single Layer and 16 Multi-Layer Perceptrons (one hidden layer) considered within CNBC for experiments over a single PolSAR image.

No.	Configuration	No.	Configuration	No.	Configuration
0	$N_{in} \times 2$	**4**	$N_{in} \times 4 \times 2$	**13**	$N_{in} \times 13 \times 2$
1	$N_{in} \times 1 \times 2$	**5**	$N_{in} \times 5 \times 2$	**14**	$N_{in} \times 14 \times 2$
2	$N_{in} \times 2 \times 2$	**...**	......	**15**	$N_{in} \times 15 \times 2$
3	$N_{in} \times 3 \times 2$	**12**	$N_{in} \times 12 \times 2$	**16**	$N_{in} \times 16 \times 2$

Positive-to-negative ratio with PNR = 10 is used for negative sample selection for all BCs. The evolution (and training) parameters and internal settings of the BCs are as follows: For MD-PSO, the termination criteria are the combination of the maximum number of iterations allowed (*iterNo* = 1000) and the cut-off error ($\varepsilon_c = 10^{-4}$). Other parameters were empirically set as the swarm size, $S = 100$, $V_{max} = x_{max}/5 = 0.2$, and $VD_{max} = 10$, respectively. For MLP, the learning parameters are set as $\eta = 0.002$ and iteration number as 1500 for BP applying the

typical activation function: *hyperbolic tangent* $\left(\tanh(x) = \dfrac{e^x - e^{-x}}{e^x + e^{-x}} \right)$.

The MLP architecture space includes the simple configurations with the following range arrays: $R_{min} = \{|FV_n|, 1, 2\}$ and $R_{max} = \{|FV_n|, 16, 2\}$, which indicate that besides the single layer perceptron (SLP), all MLPs have only a single hidden layer, $L_{max} = 2$, with no more than 16 hidden neurons. Table 16 lists all MLP configurations in the predefined architecture space (AS). Finally, for both training methods, the exhaustive BP and MD-PSO, $N_R = 10$ independent runs are performed. Note that for exhaustive BP, this corresponds to 10 runs for each configuration in the AS.

The training dataset contains only ~1000 pixels, which corresponds to merely ~0.1 percent of the entire PolSAR image; and its accuracy cannot be 100 percent guaranteed. For instance, due to the low resolution, when selecting training patches it might happen accidently that pixels are assigned erroneously caused by their proximity (e.g., patch of grass in forest might not be clearly recognizable visually). The *Urban* areas may also cover trees (planted alongside roads or gardens of houses). We can at best assume that the *majority* of those points belong to the terrain they are assigned to as in the *Urban* case some training samples might fall onto pixels with more grass or tree-like characteristics. Thus, classification is performed over the majority terrain type.

For the SFBay AIRSAR image, the training ground truth data and the classification results

with the two *CETs* are shown in Figure 66. From the figure can be seen that both *CETs* achieved quite similar classification results, yet the *CET-1* (MLPs trained with exhaustive BP) classification can be considered a little noisy or fractured compared to the other. However, both *CETs* in general managed to draw major boundaries between distinct terrains, such as *Sea – Beach, Urban – Forest, Sea – Urban*, as anticipated.
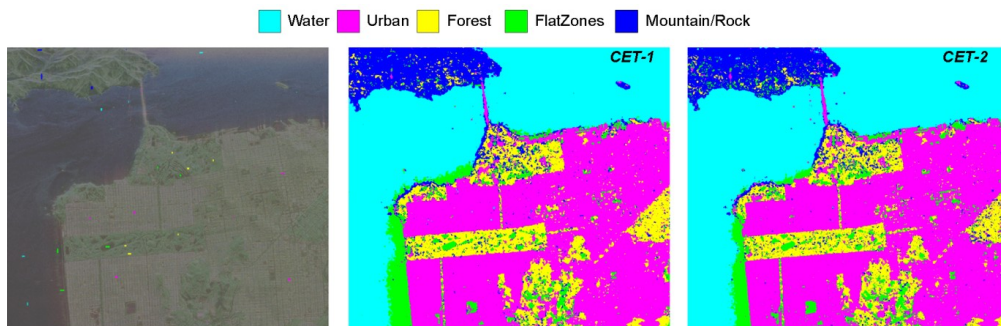
Figure 66 – San Francisco Bay AIRSAR image overlaid with training data (left) and CNBC classification results from the two *CETs*.
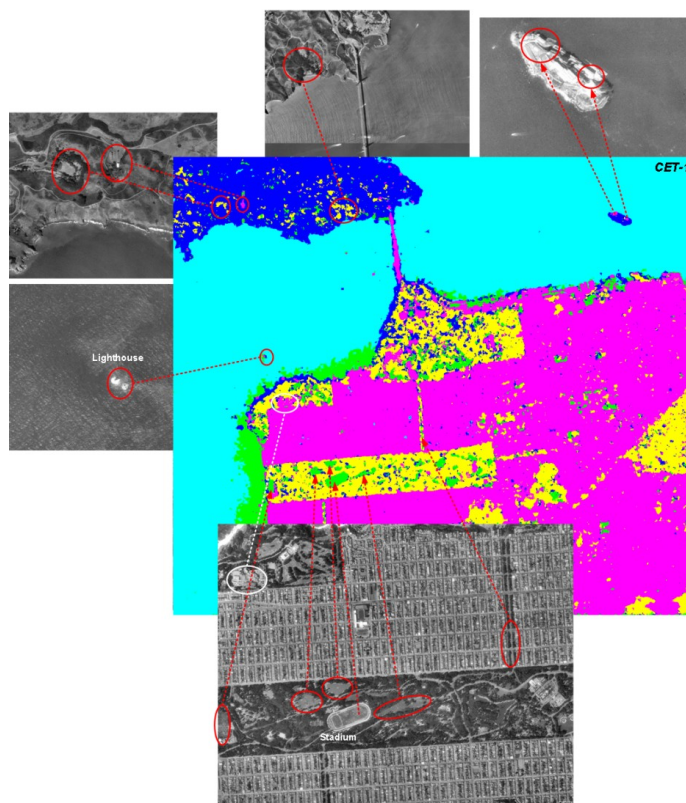
Figure 67 – The visual evaluation of the San Francisco Bay AIRSAR image classification result with an aerial photo taken from [193].
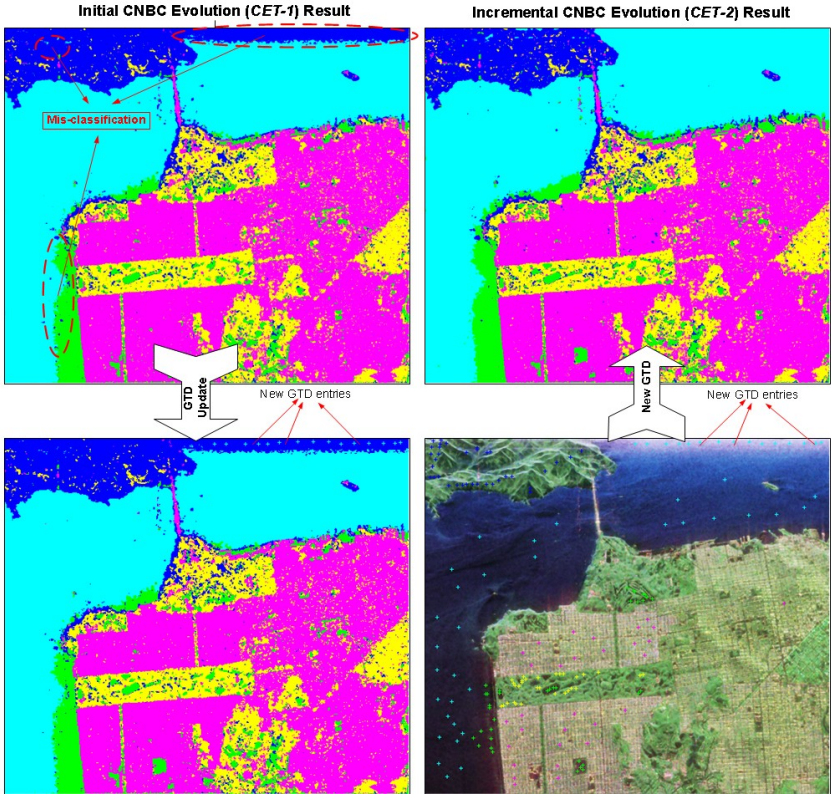
Figure 68 – Classification results from two distinct CNBC evolutions. The second (incremental) evolution corrects most of the misclassification errors.

Since the ground truth is not entirely available, we, therefore, evaluate the classification results visually using the output from *CET-1*, and aerial photo snapshots taken from [193]. Some parts are purposefully enlarged for enhanced visual clarity and compared with the *CET-1* classification result, as shown in Figure 67. It is visible that some large terrains, such as the bridge, the stadium, and the island or even some small areas such as the lighthouse and its island are all classified correctly. Nevertheless, the classification is not pixel accurate as some deformations are visible over some known geometrical constructions such as the bridge and the stadium. Note that SAR data are quite noisy and speckle filtering was initially performed, thus, accurate pixel-based classification is not expected in this particular case.

We shall show a sample utilization of the incremental evolution of the CNBC and evaluate its performance gain. We used the same experimental setup as before and only changed the initially training, which now contains around 150 points, approximately 30 points per terrain class. The BCs, having the same AS as before, are first evolved with exhaustive BP (*CET-1*). Then the user updates the training data with only 28 new entries (15 to *Water*, 10 to *Flat Zones,* and 3 to *Mountain*) over the erroneous zones and performs a new incremental evolution session with MD-PSO using 500 iterations and 5 runs. Recall that MD-PSO swarm was not fully randomized this time [107], instead, it *learns* from the previous experience of the

CNBC while using the new training data. Figure 68 shows classification results of the initial and incremental evolution by which the CNBC is able to correct the majority of classification errors. An important observation worth mentioning here is that the classification result obtained by the training data with only 178 points is just slightly worse than the one obtained with ~1000 points (see Figure 66). This shows that the CNBC can scale quite well with the size of the training data as long as it does not lack discriminative features.

4.3.2. Class- and Feature-Based Incremental Training

This section demonstrates the scenarios of class- and feature-based incremental training using the experimental setup with *CET-1*.

The first three-step class-based experimental setup is as follows. In the first step, a CNBC is created and evolved for three major terrain classes (*Water*, *Urban*, and *Forest*). Then, in step two, the *Flat Zone* terrain class is added and finally the *Mountain* terrain class is inserted in the third step. After each step the CNBC is incrementally evolved with the new training data available, which is also verified over the existing NBCs, i.e., whether the new data exceed a classification error of five percent based on procedures in Sections 4.2.3.2 and 4.2.3.1. If this is the case for a particular NBC then it will be subject to an incremental evolution; otherwise, it will be kept as is. Table 17 summarizes the classes used in each step with their number of class training samples and the classification error obtained *before* the upcoming incremental evolution process.

Table 17 – Incremental CNBC evolutions with the addition of new classes over the San Francisco Bay AIRSAR image (Orange denotes incremental evolution necessary, and green no changes).

Step	Terrain class	No. of training samples	Classification error during verification	
1	*Water*	171		
	Urban	195		
	Forest	180		
2	*+Flat Zone*	221	*Water*:	151 (68%)
			Urban:	46 (21%)
			Forest:	24 (11%)
3	*+Mountain*	188	*Water*:	9 (4.8%)
			Urban:	0 (0%)
			Forest:	95 (50.5%)
			Flat Zone:	84 (44.7%)
		955		

It can be seen that adding the *Flat Zone* class in step two required an incremental evolution for all NBCs (classes) whereas for the *Mountain* class addition in the third step NBCs for the *Water* and *Urban* class were kept untouched. Classification results of the incrementally

evolved CNBC over the SFBay image are shown in Figure 69 A, B, and C for three, four, and five classes, respectively.
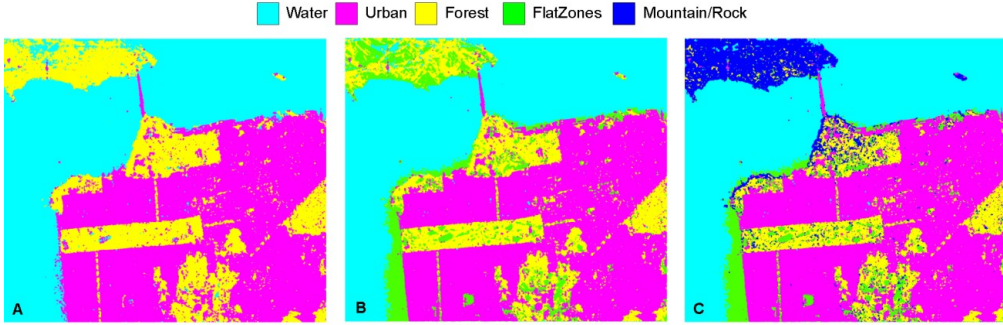
Figure 69 – Classification results of incrementally evolving one CNBC over the San Francisco Bay AIRSAR image starting with a three-class problem (A) and adding new ground truth classes in two additional steps (B) and (C).

Now let us consider two classification setups using MLP [225] and Optimization of Polarimetric Contrast Enhancement (OPCE) [222] over the same SFBay AIRSAR image. The MLP is applied over a 600×600 speckle filtered sub image version using initially a 19 dimensional feature vector ($H\alpha A + Span$ and four GLCM features over T_{11}, T_{22}, T_{33}), which has been reduced to 10 dimension by Principal Component Analysis. The MLP configuration has one single hidden layer with 10 nodes, 10 inputs, 3 outputs: *Water*, *Urban*, *Vegetation*, and is trained with approximately 30 000 samples using resilient backpropagation with 100 epochs. OPCE is applied over a 700×900 unfiltered sub image using three features namely T_{11}, T_{22}, and H. They considered a four-class problem *Water*, *Urban*, *Forest* (or woods) and quasinatural (equivalent of *Flat Zone* class in our definition) training with approximately 17 000 samples. Images, training sets used, and classification results for both approaches are shown in Figure 70. The two 600×600 and 700×900 sub-images with their individual training areas are referred to as scenarios S1 and S2, respectively.

CNBC is applied to both scenarios yet to reduce computational complexity the training sets are sub-sampled by around 10 and 3 for S1 and S2, respectively. This is done due to their number of classes and different number of originally provided training areas. Furthermore, the GLCM features were not extracted and only the PolSAR-based features are used. As for the CNBC configuration, the setup *CET-1* is trained with 500 epochs and initial classification results are shown for the sub-images and the full image in Figure 70, right hand side.

Due to their setups, the two scenario classification results show differences (left part of Figure 71) mainly in the top-right region and coastal areas related to the achieved pixel accuracy based on the given training data. The differences in S1 are probably related to the additional GLCM features used by the MLP, where one could argue that in case of the coastal regions they are more beneficial as for the top-right region.
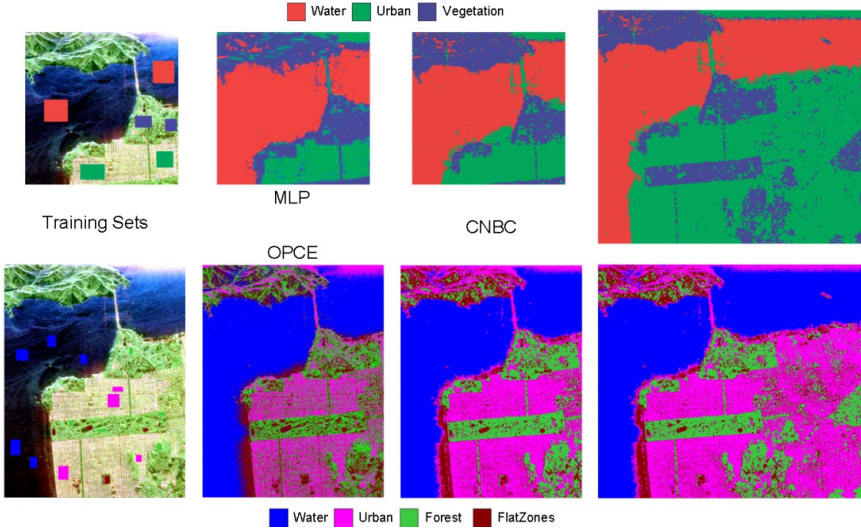
Figure 70 – Comparison of three classifiers (MLP [225], OPCE [222], and CNBC) over the San Francisco AIRSAR image.
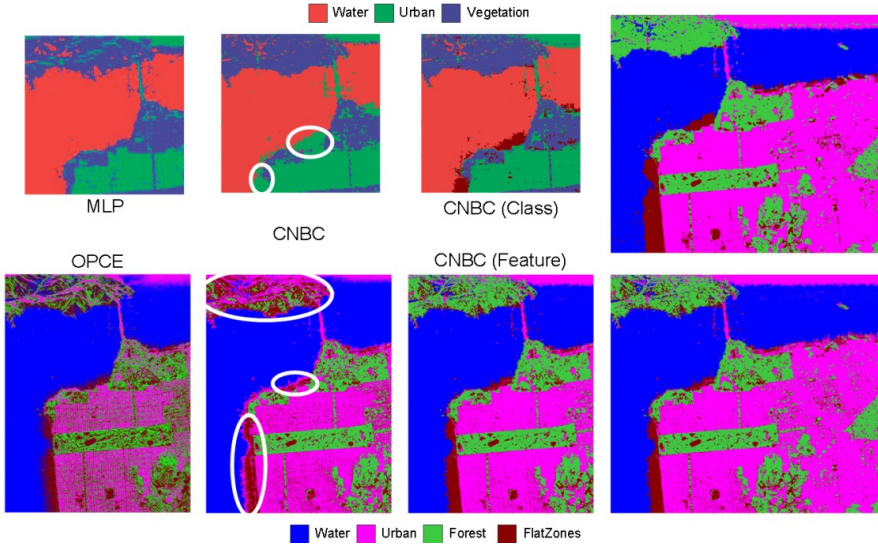
Figure 71 – Example of evolving initial CNBC results from two different tasks. Top row shows CNBC evolution by addition of a new class, bottom row shows CNBC evolution by addition of new features.

As we have the information available from S2, we know that these coastal regions are related to quasi-natural (i.e., *Flat Zone*) terrain. Therefore, the existing CNBC in S1 can be incrementally trained with the quasi-natural training data from S2. With adding this new class, the existing NBCs undergo verification with the new training data. The *Urban* class NBC fails this verification and needs to be updated using the new training data. On the other hand, the NBC related to *Water* and *Vegetation* stay untouched. This is confirmed by visual inspections

as the new *Flat Zone* class is primarily classified as *Urban*. In case of S2, the misclassifications will relate to the rather small number of features. Hence, we can extend the existing three-dimensional feature set by adding *HaA + Span* features as used in S1. This requires the evolution of one additional BC as well as updating the fuser BC.

The classification maps of the CNBC with the changes made to both scenarios are illustrated in Figure 71 (right hand side). Note that the previous misclassifications are corrected by employing active learning and adapting the existing CNBCs to the changes accordingly. In the last column of the figure, the same class legend is applied to both scenarios for better comparison. It can be observed that both scenarios achieve similar classification results, which are expected as now they both solve the same four-class problem with similar feature sets.

4.4.   CLASSIFICATION OF POLSAR DATASETS

Our main goal with the application of CNBC is the classification of a database of similar PolSAR images. For creation of such datasets, we consider collections of PolSAR images acquired by the same system. These PolSAR images are alike in the sense that they share similar land cover having a majority of common terrain classes. However, images may still have different number of classes, where class definitions may come in time from the user. Furthermore, certain terrain classes may have changed (absent or present) due to location or temporal differences among PolSAR images.
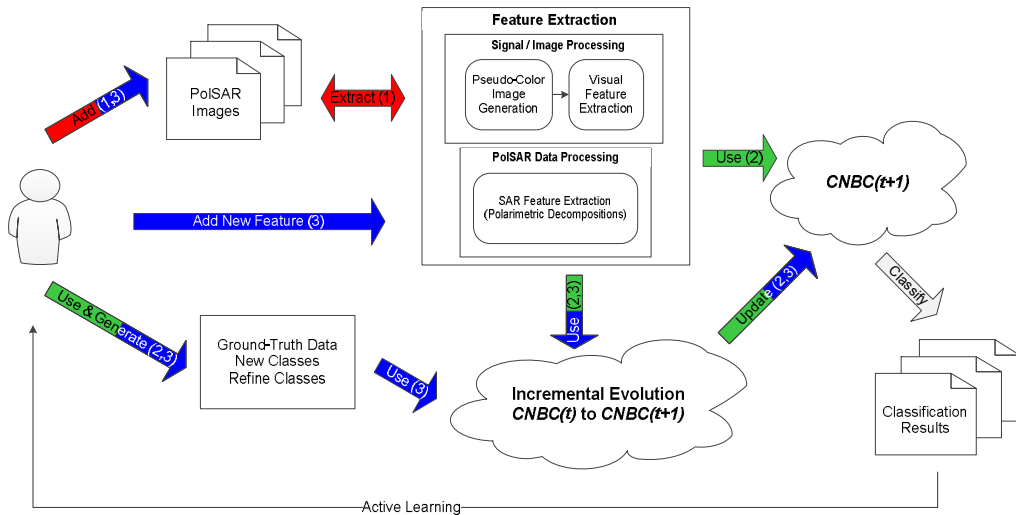
Figure 72 – Block diagram of overall classification process for multiple PolSAR images in three stages: 1. Feature extraction (red), 2. (incremental) CNBC evolution (green), and 3. Classification (blue).

The overall process is divided into three main stages. Firstly, there is the extraction of both common PolSAR and visual features from the provided PolSAR images. Secondly, CNBC is

used to evolve and classify an initial set of PolSAR images. Thirdly, incremental evolution is performed by CNBC with its adaptive and data-driven feature and class scalability, whenever a user provides ground truth for additional terrain classes appearing in new PolSAR images introduced later on. A block diagram of the overall process is shown in Figure 72. From the diagram is apparent that incremental evolution of the classification scheme utilizes active learning to use user feedback and interaction to provide new ground truth data or features to improve classification accuracy, if desired.

Two experimental setups are evaluated with the first one for visual demonstration of the process using CNBC and the second setup provides numerical evaluation. Both experiments are done over two different PolSAR image datasets from the NASA/Jet Propulsion Laboratory (JPL) Airborne SAR (AIRSAR) in L-Band obtained from the NASA AIRSAR website [147].

4.4.1. Visual Evaluation

In the image database for visual demonstration, eight different PolSAR images from three different regions are used namely Freiburg, Germany (two images), Oberpfaffenhofen (OPH), Germany (three images), and Wicomico, Maryland, USA (three image). The original four-look fully polarimetric SAR images, having a dimension of 1024×1279 pixels, provide coverage of four main terrain classes *Forest*, *Fields*, *Urban/Man-Made*, and *Water* where not all classes are present in all images or are not clearly distinguishable for a user. All images were pre-processed using speckle filtering by Lee et al. [117] over a 5×5 window.

As in the previous Section 4.3.1, the same feature set is chosen consisting of elements from $\langle[T]\rangle$ and $\langle[C]\rangle$ combined with components of $H\alpha A$ and eigenvalue decomposition as well as the three complex correlation coefficients (ρ_{12}, ρ_{13}, ρ_{23}) between scattering matrix terms. This results in a total feature set of 28 dimensions with the same three sub-feature vectors (FV_1, FV_2, FV_3) as in Section 4.3.1.

One CNBC is incrementally evolved starting with two classes (*Forest*, *Fields*) and gradually introduced more classes, in this case *Urban* and *Water*, respectively, as they appear on the other images. Due to the three FVs, each NBC has four BCs (three feature BCs in the input layer + one fuser BC); and Multi-Layer Perceptrons (MLPs) are used as the BCs that are individually evolved using exhaustive backpropagation (BP) [107] minimizing the training classification error. For the repetitive BP training of each network in the AS, the first step of an incremental evolution will simply be the initialization of the weights and biases with the parameters retrieved from the last record of the AS of that particular BC [107]. Considering this as the initial point, and using the current training dataset with the target labels, the BP algorithm can then perform its gradient descent in the error space. Furthermore, note the input layer size of each BC is determined by the dimension of its FV_n, i.e., 12, 10, and 6 for the input layer BCs of all NBCs, and naturally, 3×2 = 6 for the fuser BCs.

The BP parameters and internal settings of the MLPs were as follows: the learning parameter for BP is $\eta = 0.002$, the iteration number is 1000, and the activation function is the hyper-

bolic tangent. The MLP AS includes the simple configurations within the following range arrays: $R_{min}=\{|FV_n|,8,2\}$ and $R_{max}=\{|FV_n|,16,2\}$, which indicate that besides the single layer perceptron, all MLPs have only one hidden layer with 8 to 16 hidden neurons. Finally, seeking for the optimal configuration, 10 independent BP runs were performed for each configuration in the AS and the best one was selected. On the training samples the negative sample selection with PNR=10 is applied, if necessary, as described in Section 4.2.2.4.

Table 18 lists the images in the order they are used along with their different properties such as present terrain classes. The initial evolution starts with two classes on Wicomico1 using 900 and 495 training samples for *Forest* and *Fields*, respectively. After the initial evolution, other PolSAR images (Wicomico2 and Freiburg1) are classified until severe misclassification is encountered such as *Urban* regions in Freiburg1. After providing new training samples (360) for the *Urban* regions, the CNBC is evolved with the newly introduced *Urban* class and the NBC for *Forest* class is incrementally evolved since it failed the verification test on the new training samples due to the fact that the *Urban* regions overlapped mostly with *Forest* regions. For the NBC of the *Fields* class an incremental evolution operation was not necessary in this step. Classification using the CNBC evolved thus far over Freiburg2 and Wicomico3 presents good results, yet for OPH1, it obviously misclassified water bodies as *Fields* since a *Water* class has not yet been known by the existing CNBC. During the initial evolution of the NBC created for the *Water* class using 360 pixels as the training dataset, the existing NBC for the *Fields* class had to undergo another incremental evolution operation since it failed the verification test for the *Water* class. The resultant CNBC over the next PolSAR image, OPH2, classified most regions quite accurately but some misclassification occurred over the *Water* regions, so performing incremental evolutions for the NBCs that belong to both *Water* and *Fields* classes. The final CNBC classified the PolSAR image OPH3 accurately requiring no further incremental evolutions. Note that this process is mainly automated besides providing new training data per image, whenever required.

Table 18 – List of AIRSAR images and their properties within the CNBC experiment over multiple PolSAR images.

Process order	Dataset name	Date acquired	Terrain classes present
1	Wicomico1	02.06.1995	*Forest, Fields*
2	Wicomico2	02.06.1995	*Forest, Fields*
3	Freiburg1	15.06.1991	*Forest, Fields, Urban*
4	Freiburg2	16.07.1991	*Forest, Fields, Urban*
5	Wicomico3	02.06.1995	*Forest, Fields, Urban*
6	OPH1	12.07.1991	*Forest, Fields, Urban, Water*
7	OPH2	12.07.1991	*Forest, Fields, Urban, Water*
8	OPH3	12.07.1991	*Forest, Fields, Urban, Water*
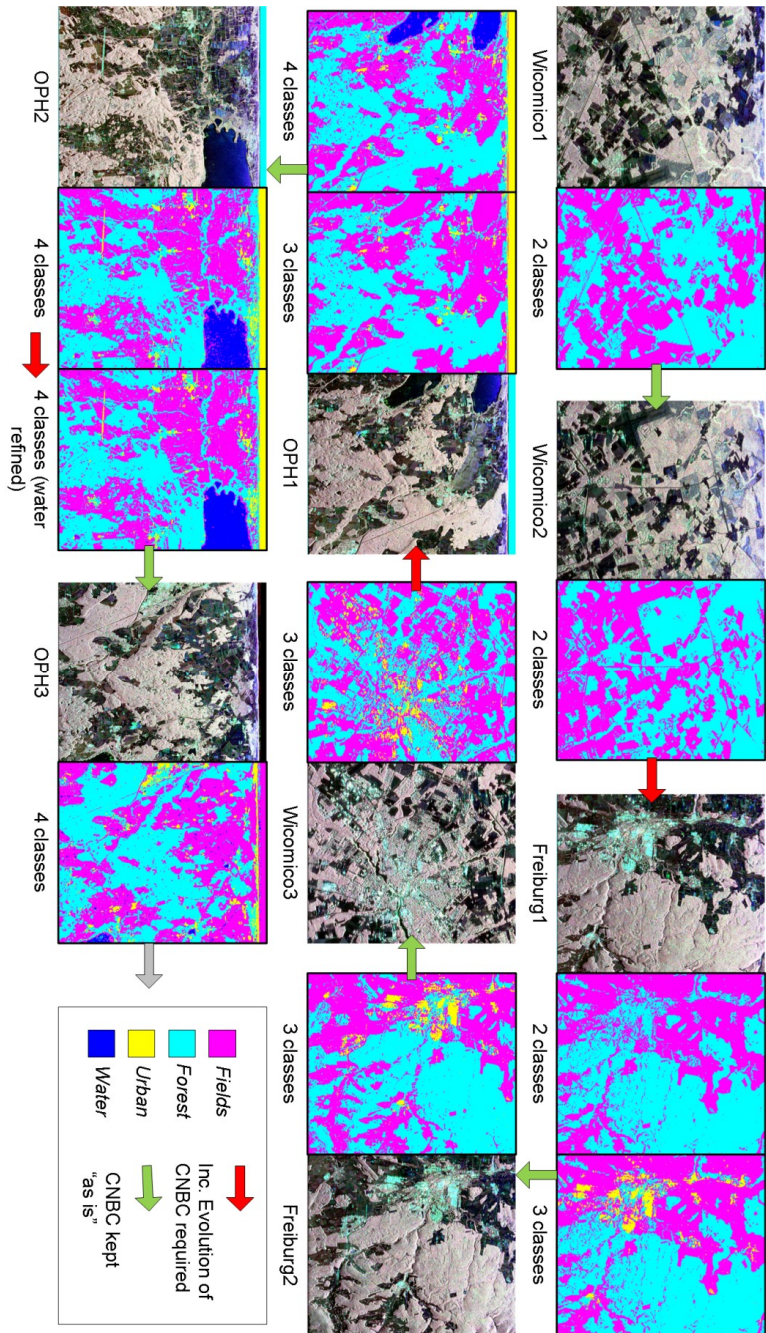
Figure 73 – Evolution steps for multiple AIRSAR images where red arrows indicates an incremental evolution operation with new class(es), and green arrows indicates that the CNBC is kept "as is".

Table 19 – Overview of CNBC evolution steps with new data (the numbers indicate positive/negative training from all previous classes samples) over multiple PolSAR images (Red indicates the insertion of a new class, orange denotes incremental evolution necessary, and green no changes).

Class	Initial Evolution (Wicomico1)	Incremental Evolution adding class *Urban* (Freiburg1)	Incremental Evolution adding class *Water* (OPH1)	Incremental Evolution refining class *Water* (OPH2)
Forest	900/495	900 / 855		
Fields	495/900		495 / 2115	495 / 2250
Urban		360 / 1395		
Water			360 / 1755	135 / 1350

Table 19 summaries the incremental evolution steps with the involved classes and their corresponding positive and negative training samples; and Figure 73 illustrates the processing steps with the PolSAR images and their corresponding classification results visually. In this experiment, since new training samples are only provided for the new classes, for (incremental) evolutions samples from the existing classes had to be used as well. It is also worth mentioning that the incremental evolutions of the NBC for the *Water* class have been performed over training datasets obtained from three and four different PolSAR images.

4.4.2. Numerical Evaluation

In the second experiment, the PolSAR image database contains images over the areas of San Francisco, Long Beach, and San Diego, California, USA. As detailed in Table 20, four medium to high-resolution PolSAR images were obtained, slant to ground range processed, speckle filtered [117], and then divided into equally sized non-overlapping sub-images (divided along the y-axis see Figure 74) where eight of them are selected to create a collection of PolSAR images for the experiment.

Table 20 – Data from NASA/JPL AIRSAR used for creating PolSAR image collection.

Name	Date acquired	AIRSAR Process ID	Ground resolution	No. of used sub-images
San Francisco	04/1995	cm5440	3584×4430	2
Long Beach 1	10/1998	cm5460	3432×5291	2
Long Beach 2	11/1998	cm5513	3368×6559	2
San Diego	11/1998	cm5489	3640×6644	2
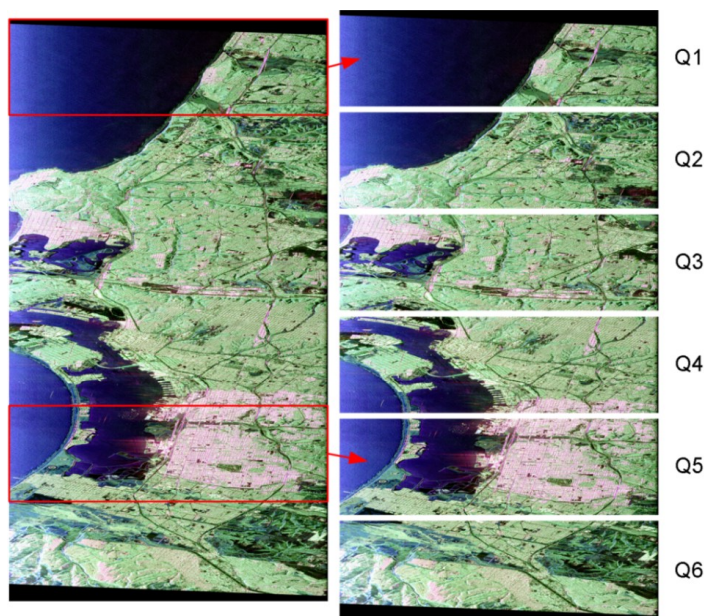
Figure 74 – Demonstration of splitting image into several sub-images for generating PolSAR image collection.

To evaluate the classification accuracy numerically, rather than manually identifying and labeling possible terrain classes we utilized the USGS National Land Cover Dataset (NLCD) 1992 [87] and 2001 [69] as an approximation for the ground truth data. Unfortunately, the NLCDs are not directly applicable, thus the following pre-processing steps were performed to map the NLCD to the AIRSAR images. First, the NLCDs from 1992 and 2001 have different number of terrain classes defined. NLCD from 1992 has 21 and NLCD 2001 includes 19 clas-ses, which also differ in naming and labeling conventions. Since the NLCDs and AIRSAR images have different acquisition dates, only the unchanged areas from NLCD 1992 to 2001 are used as ground truth data with the underlying assumption that if the specified terrain class has not changed from year 1992 to 2001, it has not changed in the years between 1995 and 1998, as shown in Figure 75. To do so, the NLCDs is reduced to four major terrain classes from the NLCD 92 Land Cover Class Definitions [87] resulting in *Water, Developed/Man-Made/Urban, Forest,* and *Grasslands*. The reason for such a selection scheme is to map the ground truth data to the AIRSAR images by warping (nearest neighbor approach) using 40-50 manual ground control points. To minimize the amount of false mapping, we have chosen those classes, which occupy at least five percent area of the eight selected sub-images so that when partitioning the training and test samples the influence of warping errors is reduced. The same approach was chosen to select the classes in each sub-image for partitioning between training and test datasets employing 200 pixels for training and 5000 pixels for testing per class in each sub-image. Table 21 lists the five sub-images used for incremental evolution in sequentially order with their corresponding training/test classes. For the evaluation in the test dataset, all eight sub-images with their test classes are given in Table 22.

Figure 75 – Ground truth generation for a collection of SAR images. Obtaining ground truth data (C) from NLCD 1992 (A) and NLCD 2001 (B) illustrated over the area of San Diego.

Table 21 – Training images in sequence applied to incremental CNBC evolution for PolSAR image collection experiments. The bold classes denote introduction of a new class.

Name	Abbr.	sub-image index	Training terrain classes	Training / Validation size
San Francisco	SF_Q3	3	*Developed*, *Water*	400 / 10000
San Francisco	SF_Q4	4	*Developed*, *Water*, *Grasslands*, *Forest*	400 / 20000
Long Beach 1	LB1_Q2	2	*Developed*, *Water*, *Grasslands*	600 / 15000
Long Beach 2	LB2_Q2	2	*Water* / (*Developed*, *Water*) – contains two major classes and only training samples for *Water* are provided	200 / 10000
San Diego	SD_Q2	2	*Water* / (*Developed*, *Water*, *Grasslands*)	200 / 15000
Total				1800 / 70000

Table 22 – Test images and their class information used for evaluation of final CNBC classifier for PolSAR image collection experiments. The highlighted rows denote images not considered during training stage.

Name	Abbr.	sub-image index	Test terrain classes	Test size
San Francisco	SF_Q3	3	All	20000
SanFrancisco	SF_Q4	4	All	20000
Long Beach 1	LB1_Q2	2	*Developed*, *Water*, *Grasslands*	15000
Long Beach 1	LB1_Q1	1	*Developed*, *Water*, *Grasslands*	15000
Long Beach 2	LB2_Q2	2	*Developed*, *Water*	10000
Long Beach 2	LB2_Q2	1	All	20000
San Diego	SD_Q2	2	*Developed*, *Water*, *Grasslands*	15000
San Diego	SD_Q5	5	*Developed*, *Water*, *Grasslands*	15000
Total				130000

Table 23 – Different sets of PolSAR, texture, and color feature vectors employed in training CNBC over a PolSAR image collection.

Feature set	Feature vectors	Dimension
FS_1	FV_1	3
FS_2	$FS_1 + FV_2 + FV_3$	23
FS_3	$FS_2 + FV_4 + FV_5$	36
FS_4	$FS_3 + FV_6 +…+ FV_9 + FV_{11} +…+ FV_{13}$	177

The incremental evolution process is applied as follows. One initial CNBC is created from the first PolSAR image SF_Q3 in the training dataset. After that, each one of the other Pol-SAR images in the training dataset is incrementally evolved to adapt the CBNC to changes (i.e., new classes, new training samples) introduced with the new PolSAR image. After all images have been considered for evolution (SF_Q3 $\Rightarrow$ SF_Q4 $\Rightarrow$ LB1_Q2 $\Rightarrow$ LB2_Q2 $\Rightarrow$ SD_Q2), the final CNBC can then be used to classify all images in the test dataset.

In order to evaluate the performance gain/loss that can be obtained by using different set of features, as enumerated in Table 23, four different feature sets (based on the different feature vector, FV_n, as described in Section 3) over two classification schemes are considered for this classification evaluation. Differences to previous feature vectors are that FV_1 represents a bare minimum of information containing only the three diagonal elements of $\langle[T]\rangle$, and FV_2 is the same as in Section 4.3.1 excluding *RVI*. Note that texture and color features (FV_6 to FV_{13}) are extracted over a local 11×11 pixels window of the pseudo color images generated by assigning T_{11}, T_{22}, and T_{33} to the R, G, and B components, respectively. Due to working with different number of images, besides normalizing the features to [-1, 1], we further apply histogram equalization to the feature vectors as a pre-processing step to overcome the contrast difference among the images due to different capturing times and possible incident angles.

The classification accuracies of the CNBC are evaluated against traditional MLPs and RBFs over incremental evolutions. The CNBCs created and evolved over the different AIR-SAR images contain the number of NBCs, which is equivalent to the number of pre-defined classes *C* (i.e., training classes per sub-image, see Table 21). Each NBC may have minimum of one (e.g., for FS_1) and up to 12 BCs (e.g., 11 in the input layer + one fuser BC for FS_4). Thus, a total of $C×1$ (FS_1) to $C×12$ (FS_4) BCs are individually evolved over the architecture spaces of two major ANN types (MLP and RBF) using the MD-PSO as the evolutionary technique, referred as MLP_PSO and RBF_PSO.

The evolution parameters and internal settings of the BCs are as follows: MD-PSO uses the termination criteria as the combination of the maximum number of iterations allowed (*iterNo* = 500) and the cut-off error $\varepsilon_c=10^{-3}$. Other parameters were empirically set as the swarm size, $S = 100$, $V_{max} = 0.2$ (velocity), and $VD_{max} = 10$ (dimension), respectively. Both MLP and RBF architecture spaces use the simple configurations with the following range arrays: $R_{min} = (|FV_n|,5,2)$ and $R_{max} = (|FV_n|,15,2)$. Finally, $N_R = 5$ runs are performed for MD-PSO.

Table 24 – Overview of incremental evolution steps with new data over the different training images and feature sets for PolSAR image collection experiments. (Red indicates the insertion of a new class, orange denotes incremental evolution is necessary, and green indicates no changes).

		Initial Evolution (SF_Q3)				Incremental Evolution adding class *Grassland and Forest* (SF_Q4)				Incremental Evolution refining class *Developed, Water, Grasslands* (LB1_Q2)				Incremental Evolution refining class *Water* (LB2_Q2)				Incremental Evolution refining class *Water* (SD_Q2)			
		2 classes				4 classes				3 classes				2 classes				3 classes			
Class / FS		**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
CNBC RBF_PSO	*Developed*	Red	Red	Red	Red	Green	Green	Green	Green	Green	Green	Orange	Orange	Green	Green	Green	Green	Green	Green	Green	Green
	Water	Red	Red	Red	Red	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange
	Grasslands					Red	Red	Red	Red	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange
	Forest					Red	Red	Red	Red	Orange	Orange	Orange	Orange	Green	Green	Green	Green	Green	Green	Green	Green
CNBC MLP_PSO	*Developed*	Red	Red	Red	Red	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Green	Green	Green	Green
	Water	Red	Red	Red	Red	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange
	Grasslands					Red	Red	Red	Red	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange	Orange
	Forest					Red	Red	Red	Red	Orange	Orange	Orange	Orange	Green	Green	Orange	Green	Green	Green	Green	Green
sMLP/sRBF	*Developed*	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red
	Water	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red
	Grasslands	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red
	Forest	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red

As for the comparison, the two well-known feed-forward ANN classifiers: Multi-Layer Perceptron (sMLP) and Radial Basis Function networks (sRBF) are selected despite the fact that they can only be applicable to static classification problems with fixed input and output dimensions. Since optimal BC configurations within each NBC are searched by the underlying evolutionary search method (MD-PSO), in order to provide a fair comparison, the best possible classifier architectures and/or parameters are also searched for sMLP and sRBF. For sMLP, this means that the best possible network configuration is searched within an architecture space encapsulating several MLPs with one and two hidden layers, where the first hidden layer may range from 10-20, and the second hidden layer from 5-10 neurons. This results in 50 configurations in the sMLP AS. For sRBF, a range from 10-50 neurons is employed for the single hidden layer. For both ANN types, sMLP and sRBF, MD-PSO is applied as the underlying training function, which searches for the best configuration within the predefined AS as well as the optimum network parameters (connections, weights, and biases) for each of the configurations.

The computational complexity in terms of incremental evolutions for each of the training images per feature set and classification scheme are shown in Table 24. Whenever new training data arrive, the ANNs, sMLP and sRBF, will become obsolete and a new classifier needs

to be created and re-trained from scratch to accommodate to the changes. For CNBC, recall that it will adapt its structure to the new training data by simply including a new NBC for each new class (*Grasslands* and *Forest* in SF_Q4). Furthermore, the existing NBCs will undergo a verification test (see Section 4.2.3.1) to validate if they are capable of classifying the new training data with a required accuracy (i.e., classification accuracy greater 95 percent).

For SF_Q4, the class NBC for *Water* fails with the new training data of classes *Grasslands* and *Forest* for both CNBCs whereas for MLP_PSO, the *Developed* class NBC also fails the verification for all feature sets. For LB1_Q2, LB2_Q2, and SD_Q2 no new classes are introduced but due to classification errors, especially on the class *Water*, new training data are provided to overcome these errors. This resulted in major incremental evolution for all classes of LB1_Q2, whereas the *Forest* class NBC for CNBC MLP_PSO in LB2_Q2 and *Developed* and *Forest* class NBCs for CNBC RBF_PSO/MLP_PSO in LB2_Q2 and SD_Q2 were kept untouched.

For the SF_Q3 and SF_Q4, as both subimages are obtained from the same image data, the main differences in classification results come from the changes in terrain class numbers. In case of LB1_Q2, the big performance drop is due to differences in capturing times (see Table 20) and possible incident angles. This will result in different backscattering characteristics of the underlying terrain, and recall that only a simple method (histogram equalization) is applied to overcome this. The performance over LB2_Q2 is mainly due to the two-class problem, where the applied features make the difference as *Water* can be easily distinguished due to its dominant surface scattering. Whereas for SD_Q2, the accuracy drops are mainly related to capturing times and terrain changes, as it is fairly urban dominated.

Moreover, with increasing number of features classification accuracies are improved as anticipated, particularly the addition of target decompositions to the basic FS_1 provide significant discrimination improvements. This is expected as classifiers using FS_1 do perform poorer since there are only three features, which are not enough to discriminate the classes properly. Furthermore, classification results achieved using FS_2 and FS_3 are rather similar due to the used target decomposition components as also observed in previous evaluations 3.3.4. Classification differences over the training images are illustrated in Figure 76 with the two CNBCs benefiting from FS_4 whereas both standalone ANNs did not gain from the additional features. This is most probably due to the limited AS applied and the larger dimension of FS_4 (177). Major differences can be noticed in Table 25 whenever the underlying classification task or newly provided data change (e.g., SF_Q4, LB1_Q2).

After the incremental evolution over the last training image, SD_Q2, the final CNBC is evaluated by its classification performance, shown in Table 26, over the ground truth data of the eight PolSAR images in the test dataset as listed in Table 22. It can be observed that using more features usually improves the classification accuracy. However, the classifier performance of the standalone ANNs (sMLP and sRBF) suffers when the feature dimension exceeds a certain limit that is basically the well-known "curse of dimensionality" phenomenon for such static classifiers. For smaller feature sets, sMLP and sRBF can perform equally or even better than CNBC.

Table 25 – Classification accuracies after incremental evolution over each of the training images for PolSAR image collection experiments. Differences for MLP and RBF inside CNBC regarding their incremental evolutions are highlighted according to Table 24.

		SF_Q3 (2 classes)				SF_Q4 (4 classes)			
		FS_1	FS_2	FS_3	FS_4	FS_1	FS_2	FS_3	FS_4
CNBC	MLP_PSO	0.996	0.996	0.996	0.996	0.698	0.809	0.825	0.844
CNBC	RBF_PSO	0.994	0.995	0.996	0.997	0.677	0.805	0.812	0.841
ANN	sMLP	0.995	0.996	0.989	0.990	0.688	0.820	0.811	0.744
ANN	sRBF	0.995	0.996	0.996	0.996	0.671	0.792	0.769	0.743

		LB1_Q2 (3 classes)				LB2_Q2 (2 classes)			
		FS_1	FS_2	FS_3	FS_4	FS_1	FS_2	FS_3	FS_4
CNBC	MLP_PSO	0.591	0.642	0.640	0.651	0.557	0.873	0.838	0.889
CNBC	RBF_PSO	0.530	0.631	0.636	0.659	0.385	0.855	0.854	0.866
ANN	sMLP	0.587	0.645	0.636	0.640	0.525	0.821	0.817	0.792
ANN	sRBF	0.617	0.641	0.638	0.601	0.465	0.908	0.809	0.886

		SD_Q2 (3 classes)			
		FS_1	FS_2	FS_3	FS_4
CNBC	MLP_PSO	0.474	0.569	0.579	0.628
CNBC	RBF_PSO	0.464	0.527	0.529	0.671
ANN	sMLP	0.494	0.568	0.599	0.600
ANN	sRBF	0.496	0.563	0.556	0.537
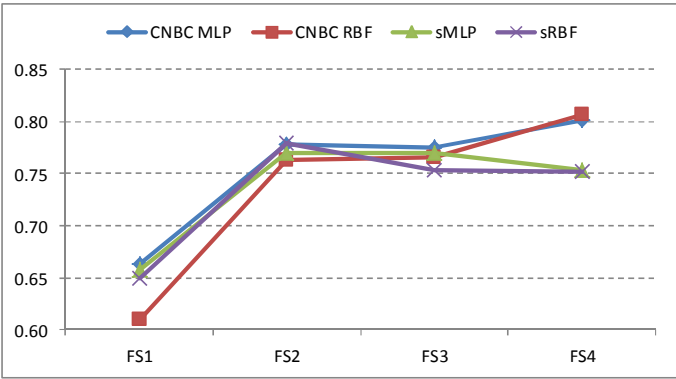
Figure 76 – Classification accuracies averaged over the five training AIRSAR images for the four different feature sets.

However, the performance drops for sMLP and sRBF are more severe than observed over the training stages in Table 25. This is most probably due to the dimensions of FS_4 and the architecture space. Additionally, combining the training data of the different images might not

be sufficient to handle the differences in capturing times and incident angles and their resulting differences in backscattering characteristics.

Figure 77 shows the average classification accuracies over the eight test images based on the four different feature sets. It is visible that all four classifiers have a boost (15-20%) from FS_1 to FS_2 and a further, even though marginal increase, with FS_3. This small increase is probably related to the addition of FV_4 and FV_5, which provide similar features (incoherent decomposition) as FV_3. It also shows that from FS_3 to FS_4 the classification performance of both sMLP and sRBF, drop 7% and 9%, respectively whereas both CNBC variants, MLP_PSO and RBF_PSO, can further improve their classification performances (by ~2% and ~3%).

Table 26 – Classification performances over the test dataset at the end of the five (incremental) evolution sessions for the PolSAR image collection experiments.

		SF_Q3 (2 classes)				SF_Q4 (4 classes)			
		FS_1	FS_2	FS_3	FS_4	FS_1	FS_2	FS_3	FS_4
CNBC	MLP_PSO	0.690	0.788	0.784	0.669	0.643	0.811	0.809	0.829
CNBC	RBF_PSO	0.672	0.767	0.789	0.678	0.599	0.759	0.796	0.828
ANN	sMLP	0.688	0.761	0.754	0.658	0.664	0.784	0.787	0.762
ANN	sRBF	0.682	0.655	0.727	0.496	0.639	0.625	0.711	0.454

		LB1_Q2 (3 classes)				LB1_Q1 (3 classes)			
		FS_1	FS_2	FS_3	FS_4	FS_1	FS_2	FS_3	FS_4
CNBC	MLP_PSO	0.724	0.832	0.840	0.857	0.527	0.594	0.601	0.600
CNBC	RBF_PSO	0.579	0.818	0.820	0.848	0.457	0.618	0.619	0.624
ANN	sMLP	0.729	0.813	0.854	0.724	0.596	0.614	0.675	0.543
ANN	sRBF	0.695	0.803	0.763	0.751	0.562	0.622	0.595	0.538

		LB2_Q2 (2 classes)				LB2_Q1 (4 classes)			
		FS_1	FS_2	FS_3	FS_4	FS_1	FS_2	FS_3	FS_4
CNBC	MLP_PSO	0.572	0.861	0.839	0.895	0.389	0.543	0.557	0.641
CNBC	RBF_PSO	0.414	0.882	0.855	0.888	0.333	0.555	0.590	0.654
ANN	sMLP	0.591	0.844	0.825	0.762	0.432	0.574	0.570	0.573
ANN	sRBF	0.629	0.910	0.820	0.890	0.383	0.536	0.577	0.469

		SD_Q2 (3 classes)				SD_Q5 (3 classes)			
		FS_1	FS_2	FS_3	FS_4	FS_1	FS_2	FS_3	FS_4
CNBC	MLP_PSO	0.474	0.569	0.579	0.628	0.572	0.610	0.615	0.691
CNBC	RBF_PSO	0.464	0.527	0.529	0.671	0.511	0.624	0.636	0.663
ANN	sMLP	0.494	0.568	0.599	0.600	0.584	0.641	0.720	0.616
ANN	sRBF	0.496	0.563	0.556	0.537	0.564	0.582	0.612	0.550

Thus, in this scenario of classifying a collection of PolSAR images, the application of a data-driven and adaptive topology is advantageous as changes in number of classes and features can be accommodated easier rather than completely re-designing and optimizing the entire classifier each time changes occur.
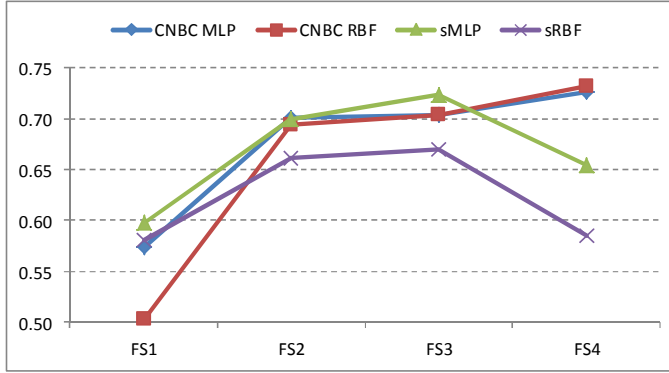
Figure 77 – Classification accuracies averaged over the eight test AIRSAR images for the four different feature sets.

4.5. SUMMARY

To address the challenge of larger-scale PolSAR image classification, we considered a data-driven and adaptive classification approach. To achieve this, the Collective Network of Binary Classifier framework is adapted with the primary objectives of maximizing efficiency and accuracy. The proposed framework mainly adopts a "divide and conquer" type of approach, so as to efficiently handle indefinite number of PolSAR features and terrain classes, which is otherwise a difficult, if not infeasible, problem for a single classifier due to the well-known "curse of dimensionality" phenomenon. In the CNBC framework approach, compact classifiers, which can be evolved and trained in a much more efficient way than a single but complex classifier, are conveniently considered in a predefined architecture (configuration) space, in which the optimal classifier for the classification problem at hand can be sought using evolutionary techniques such as MD-PSO. At any given time, this allows creating a dedicated binary classifier (BC) for discriminating a certain terrain type from the others with the use of a single feature. Each incremental evolution session *learns* from the current best classifier and can improve it further, possibly with another configuration in the architecture space. Moreover, with each incremental evolution, new classes and features (such as previously proposed color features) can also be introduced which signals CNBC to adjust to such change. In this way, the CNBC can dynamically adapt itself to a classification problem while striving for maximizing classification accuracy.

The two main benefits for this learning approach are, on one hand, computational complexity with changes in the underlying classification task and, on the other hand, due to the employed topology and "divide and conquer" concept. Thus, the application of numerous fea-

tures may lead to an improved classification performance addressing the "curse of dimensionality" phenomenon. However, for smaller number of features, it is expected that classification accuracies will be slightly in favor of static methods without dividing the feature sets. The capabilities of CNBC are demonstrated over an extensive set of experimental results regarding the adaptation and data-driven classification of single as well as collections of PolSAR images. This revealed that with the ongoing incremental evolutions, a generic CNBC can be created that can adapt itself according to the PolSAR image and classes presented over time. Compared to static classifiers such as standalone feed-forward ANNs, creation and extension of an existing classifier is beneficial in terms of classification accuracy and time without discarding existing knowledge and past training efforts. Additionally, while static ANNs and other regular classifiers may suffer from the "curse of dimensionality" phenomenon without an accompanying feature selection or dimension reduction technique, CNBC addresses this with its "divide and conquer" strategy.

Our experimental results demonstrated that the evolutionary classifier framework, CNBC, provides an efficient solution for the problems of scalability and dynamic adaptability by allowing both feature space dimensions and the number of terrain classes in PolSAR image collections to vary dynamically. Whenever the CNBC is evolved in batch mode, it can compete and even surpass other static classifiers especially when the feature space dimension increases. This is an expected outcome since the CNBC framework can take advantage of any visual feature as long as it has the discrimination power for only one or few classes. Moreover, experiments over collections of different NASA/JPL AIRSAR images showed that once the CNBC achieves a certain level of maturity, it can classify new PolSAR images with similar terrain classes requiring no or only minimal incremental evolutions.

Chapter 5

Semi-Supervised Learning in Ill-posed PolSAR Image Classification

M achine learning approaches are commonly used for classification of remote sensing data in various applications. Generally, supervised learning (SL) approaches are able to achieve better results than unsupervised learning (UL) methods due to incorporating prior knowledge in the form of ground truth data. Yet at the same time, this can be considered a drawback since SL requires labeled training data from a human expert, which can be time consuming and error prone manual labor. Even though this is probably the situation for the majority of fields SL is applied to, it is a particularly difficult process for classification of remote sensing image. The reason is that on-site surveys are ideally conducted to the locations from which the remote sensing data have been acquired, especially keeping in mind that SL benefits from larger number of labeled data during training. With a rather limited amount of training data available, the classification task easily becomes ill-posed due to the small sample size problem, where the number of training samples is noticeably smaller compared to the feature dimension. Due to this, the underlying classifier will lack discrimination and generalization capabilities. This becomes evident when the classification problems are of complex nature such as multiclass classification tasks where the number of classes might be equal or higher than the dimensions of the employed features.

To utilize the vast amount of unlabeled data while avoiding the human interaction as in active learning, the interest over semi-supervised learning (SSL) approaches has increased since it can, if properly applied, combine supervised and unsupervised learning approaches. The overall idea behind SSL is to start from a set of labeled data and then utilize the large amount of unlabeled data to improve the initial learner. Therefore, the crucial part in this process is the automatic selection of reliable and informative training samples from the unlabeled data. As mentioned in Section 4.1.4, common concepts explore *unsupervised learning*, *self-training* by single classifier, and *co-training* by multiple classifiers. In addition, several assumptions

are generally exploited such as *local smoothness / consistency assumption*, *global cluster assumption* and *low-density separation* as well as the *fitting constraint*.

In the area of remote sensing image classification, SSL has recently attracted a lot of attention. Two decades ago, the early investigations showed how unlabeled samples could be beneficial for classification applications [172]. In this work, Shahshahani and Landgrebe studied techniques to address the small sample size problem by using unlabeled observations and their potential advantages in enhanced statistic estimation. Their main conclusion was that more information could be obtained and utilized with the additional unlabeled samples. Based on these observations, a self-learning and self-improving adaptive classifier [96] using generative learning was proposed to mitigate the small sample size problem that can severely affect the recognition accuracy of classifiers. To accomplish this in [96], they iteratively utilized a weighted mixture of labeled and semi-labeled samples.

Following these pioneer works, there have been various SSL approaches over remote sensing image data, for instance generative learning as semi-supervised versions of a spatially adaptive mixture-of-Gaussians model were proposed in [99] and [100]. Another approach uses graph-based methods, which rely upon the construction of a graph representation [77], where vertices are the labeled and unlabeled samples and edges represent the similarity among samples in the dataset including, for example, contextual information via composite kernels [25]. Furthermore, this graph-based approach was also employed within self-training, where the graph is used to assure reliability of newly added training examples [32], [122]. However, the general issue of graph-based methods is that the label propagation relies on the inversion of a large matrix with a size equivalent of the total number of labeled and unlabeled pixels, which limits their application for remote sensing applications.

One of the most basic semi-supervised learning approaches is to use the output of an unsupervised learning method as the input of a supervised learning approach. This has been applied to SAR images where an unsupervised clustering approach named Deterministic Annealing was used as the training input for a Multi-Layer Perceptron [80]. This type of combined approach has also been applied to other classifier types such as Support Vector Machines (SVMs) using the output of the fuzzy C-means (FCM) clustering, which was further extended by Markov Random Fields exploiting contextual information from multiple SVM-FCM classification maps [3]. A similar approach to the combination of supervised and unsupervised learning algorithms is the application of cluster kernels [187], [188] employing SVM, where so-called bagged kernels encode the similarity between unlabeled samples obtained via multiple runs of unsupervised k-means clustering.

Furthermore, SVM has been used within the context of self-training, where a binary transductive SVM has been adapted in a one-against-all topology [22]. Besides that, one-class SVM has been applied to detect pixels belonging to one of the classes in the image and reject the others [146]. Yet another semi-supervised SVM approach utilizes the so-called context-pattern in a form of 4- or 8-connected pixel neighborhoods to identify possible misleading initial training labels [21]. Besides its popularity, the application of SVMs in the semi-supervised learning context has some shortcomings such as particularly high computational

complexity, utilization of a non-convex cost function, and the usage of multiclass SVMs. These shortcomings have been addressed by using a semi-supervised logistic regression algorithm [62] and by replacing the SVMs with an artificial neural network [164] offering much better scalability than SVM-based methods.

The focus within this chapter is the challenge of ill-posed PolSAR image classification due to the small sample size problem. Therefore, first an evaluation of traditional supervised learning methods is conducted in Section 5.1 over polarimetric SAR images with respect to their classification performances for various training set sizes. In particular, attention is concentrated on rather small training data sizes per class as authored in [194]. After that, the application of semi-supervised learning using an ensemble learning approach is presented in Section 5.2 based on the author's publication [202]. This includes two basic unsupervised approaches by enlarging the initial labeled training set as well as an ensemble-based self-training method. Particularly, different strategies within the ensemble self-training are proposed on how to select reliable candidates from the pool of unlabeled samples to speed-up the learning process and to improve the classification performance of the underlying classifier ensemble.

5.1. CLASSIFIER EVALUATIONS OVER DIFFERENT TRAINING SET SIZES

One common expectation is that supervised learning will perform better with larger amounts of training data. Yet how small can the training data actually be and still achieve satisfactory generalization and classification results?

With this in mind, four commonly applied classifier approaches to PolSAR images are assessed namely k-Nearest Neighbor, Artificial Neural Networks, Support Vector Machine, and Decision Tree classifiers. Nine different variations of them are studied over two fully polarimetric SAR images from the Flevoland region corresponding to an easier and more challenging LULC classification problem with 4 and 15 terrain classes, respectively. Classification performances are evaluated over different training set sizes from up to five percent going down to mere 0.25-0.1‰ using features from the $H\alpha A$ decomposition as described in Section 2.1.

5.1.1. Experimental Setup

This evaluation uses the two fully PolSAR images as described in Section 2.5 namely Flevoland, AIRSAR, L-Band and Flevoland, RADARSAT-2, C-Band as a complex and simple classification task, respectively. The underlying ground truth information used during the experiments is presented in Table 27. Both images are speckle filtered [117] with a 5×5 window before extracting feature corresponding to FV_2 as described in Section 2.6 consisting of entropy H, anisotropy A, the average angles averages $\bar{\alpha}$, $\bar{\beta}$, $\bar{\delta}$, $\bar{\gamma}$, the three eigenvalues, *Span*, and *RVI*. These features are the components of $H\alpha A$ and eigenvalue decomposition and com-

monly used as features in PolSAR image classification. Furthermore, in the evaluation in Section 3.3.4 related to the color features, this feature combination demonstrated superior performances compared to the covariance matrix and various other target decomposition.

Table 27 – Polarimetric SAR image data and their ground truth for experiments regarding classifier evaluation using different training set sizes.

Name	No. classes	Test size	Abbr.
Flevoland	15	208186	*Flevo_L*
Flevoland	4	200000	*Flevo_C*

Table 28 – Predefine sets for finding the best classifier configurations for training set size evaluations. ([N] number of nodes, [T] number of trees).

Classifier abbr.	Classifier configuration parameter range
KNN	k=2n+1; n=0,…,6
ELM	[N] = 12, 25, 50n; n=1,…,10
MLP	1 hidden layer: [N] = 11n; n=1,…,4
RF	[T]=50n; n=1,…,4 / split=3,4,5,6,7
SVM1	(linear kernel) C (2^n; n=0,...,8)
SVM2	(polynomial d=3) C (2^n; n=0,...,8), γ (2^{-n}; n=0,…,-8)
SVM3	(Gaussian) C (2^n; n=0,...,8), γ (2^{-n}; n=0,...,-8)

As for the training set sizes, seven different variations are chosen with the smaller and medium sized training sets (TSs) TS_1-TS_4 (0.1‰, 0.25‰, 0.5‰, and 0.1%) will be evaluated with respect to their classification performances. The larger ones TS_5-TS_7 (0.5%, 1%, and 5%) on the other hand, can be regarded more challenging considering the computational performances as classification accuracies are expected to be high. In both PolSAR images, the same amount of labeled data are used so that the total number of pixels in the training sets are around 20, 50, 100, 200, 1000, 2000, and 10000 randomly drawn. Every individual combination of classifiers and training sets are run 100 times using MATLAB® (R2010b, The Math-Works Inc.) on an Intel Corel2 Quad Q9400 @ 2.66GHz with 3GB RAM running Microsoft Windows XP operating system.

Regarding the nine different classifier variations, they include the k-Nearest Neighbor (*KNN*), Multi-Layer Perceptron (*MLP1*), Support Vector Machine (pairwise multiclass *SVM* [2] with linear, polynomial, and RBF kernel), basic Extreme Learning Machine (*ELM* [3]) [92], a CART Decision Tree (*DT*), and Random Forest (*RF* [4]) that is basically an ensemble of *DT*s. For each classifier, the best possible parameters, configuration, or architecture is sought, which achieve the highest classification accuracies from a predefined range as shown in Table

---

[2] http://www.csie.ntu.edu.tw/~cjlin/libsvm/
[3] http://www.ntu.edu.sg/home/egbhuang/ELM_Codes.html
[4] http://code.google.com/p/randomforest-matlab/

28. For the two MLP classifiers, two MATLAB® in-built backpropagation training functions are considered namely `traingd` (gradient descent $\Rightarrow$ *MLP1*) and `trainscg` (scaled conjugate gradient $\Rightarrow$ *MLP2*) with their default settings and 100 epochs.

5.1.2. Experimental Results

We start with the *Flevo_C* RADARSAT-2 image as the easier classification problem due to its clearer distinctions of the underlying four terrain classes with representing classification accuracies in Figure 78 and computational complexity evaluations in Table 29. Generally, higher classification accuracies are expected with more training data and better generalization capabilities among all classifiers can thus be achieved at the price of higher computational complexity. However, over just four terrain classes, all classifiers besides *DT* and *MLP1* achieved 85-88 percent classification performance using TS_2 with only 50 training samples in total (~12 samples per class). This 3% margin among the classifiers even shrinks for the other larger training sets yielding higher classification accuracies over TS_5. On this performance level, increasing the size of the training set only provides insignificant gains on the classification performance. Furthermore, when comparing TS_3 and TS_4 as well as TS_4 and TS_5, their difference in classification performance over all classifiers is rather marginal with ~1.1% especially with TS_4 and TS_5 being *2x* and *5x* larger than TS_3 and TS_4, respectively. However, the differences in their respective training and testing complexities are negligible regarding the real execution times in seconds. Anyhow, providing labels for *5x* larger TS can be significant. The main noticeable difference is that TS_1 is with only five samples per class. *RF*, *SVM1*, and *SVM3* achieve the highest accuracies of ~ 85 percent, whereas others suffer from the limited number of samples in TS_{1-2}. The achieved classification performances among the classifiers will eventually level out for training sets larger than TS_3.
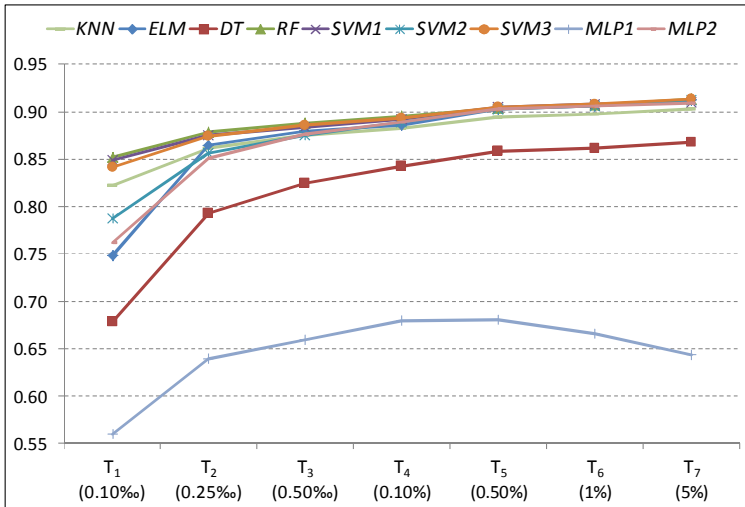
Figure 78 – Average classification accuracies for seven different training set sizes over the *Flevo_C* RADARSAT-2 image.

Table 29 – Average training and testing times in seconds for seven different training set sizes over the *Flevo_C* RADARSAT-2 image. Best performing configurations are in parentheses.

		KNN	ELM	DT	RF	SVM1	SVM2	SVM3	MLP1	MLP2
TRAINING SETS	TS$_1$ (0.10‰)	<0.01 (3)	<0.01 (12)	0.01	0.01	<0.01	<0.01	<0.01	0.66 (33)	0.44 (11)
	TS$_2$ (0.25‰)	<0.01 (7)	<0.01 (12)	0.01	0.02	<0.01	<0.01	<0.01	0.78 (44)	0.47 (11)
	TS$_3$ (0.50‰)	<0.01 (7)	<0.01 (12)	0.01	0.03	<0.01	<0.01	0.01	0.86 (44)	0.57 (22)
	TS$_4$ (0.10%)	<0.01 (9)	0.01 (50)	0.01	0.05	0.01	0.01	0.01	1.01 (44)	0.72 (22)
	TS$_5$ (0.50%)	<0.01 (13)	0.02 (50)	0.01	0.27	0.36	0.10	0.16	2.17 (44)	2.94 (44)
	TS$_6$ (1.00%)	<0.01 (13)	0.11 (100)	0.02	0.61	1.85	0.31	0.60	3.87 (44)	7.66 (44)
	TS$_7$ (5.00%)	<0.01 (13)	2.02 (400)	0.07	4.26	18.68	6.07	12.50	15.76 (44)	64.99 (44)
TEST SETS	TS$_1$ (0.10‰)	0.79	0.22	0.08	1.53	0.27	0.30	0.53	0.56	0.34
	TS$_2$ (0.25‰)	1.23	0.23	0.08	1.64	0.38	0.51	1.00	0.66	0.34
	TS$_3$ (0.50‰)	1.60	0.23	0.08	1.75	0.51	0.71	1.49	0.67	0.46
	TS$_4$ (0.10%)	2.53	0.62	0.08	1.92	0.72	1.04	1.75	0.67	0.46
	TS$_5$ (0.50%)	8.28	0.62	0.08	2.51	1.95	3.24	5.15	0.69	0.72
	TS$_6$ (1.00%)	14.19	1.10	0.08	2.84	3.42	5.45	8.94	0.67	0.69
	TS$_7$ (5.00%)	57.06	3.69	0.08	3.73	17.25	23.31	36.65	0.65	0.67

KNN, the simplest classifier among all, achieves a comparable classification performance yet its computational complexity over the test set is still dependent on the amount of training data. *DT* performs below average among all classifiers for all TSs as a weaker classifier, however, its performance can be boosted when it is used within an ensemble topology such as *RF,* which generally perform better than a *single* classifier system [58], [158]. In this case, RF with 200 trees and three feature splits is employed for all TSs. *RF*, *SVM1*, and *SVM3* achieve the highest performances among all TSs while *RF* is marginally better for TS$_{1-3}$ by 0.4%, which is an expected outcome as they are among the most common classifier choices in Pol-SAR image classification applications. Superiority of *RF* is due to being an ensemble classifier of *DT*s, which is reflected in the training and testing times compared to a single *DT. SVM2* matches the performance of *RF*, *SVM1*, and *SVM3* for TS$_{4-7}$ while having the least training times among the SVMs, yet suffers slightly for TS$_{1-3}$ probably due to the fixed polynomial degree. SVMs aim for separation in a *kernel space* so that the applied kernel as well as the number of training samples influence the training time (kernel mapping + finding support vectors). Similar to *KNN*, the number of support vectors will eventually dictate the complexity during classification phase. Therefore, *RF* significantly outperforms *SVM1 and SVM3* in training and testing complexity for TS$_{6-7}$ as more support vectors are generated for them.

For ANN classifiers, the *ELM* achieves 1-3% better classification accuracies with significantly faster training time compared to the *MLP2*. However, *ELM* testing complexity can be substantially slower (*2-6 times*) depending on its number of hidden nodes while the time complexity of the MLPs is consistently below one second over the test set. For the MLPs, there is a considerable difference between the two training functions where *MLP2* outperforms *MLP1*. The inferior performance of *MLP1* for TS_{6-7} is probably due to either over-training or the network configuration might be too limited. This is also visible in the best performing architecture with *MLP2* achieving higher accuracies with simpler networks particularly for the smaller TSs, yet it still has the highest complexity during training. Some observations can be made for *ELM* and *KNN*, when more neighboring samples are included into the decision-making. The reason is again the simplicity of the four-class problem, where the classifier complexity scales well to the number of training samples.

As for the *Flevo_L* AIRSAR image, the classification task is more complex as there are 15 LULC classes present with the classification performances and complexities shown in Figure 79 and Table 30, respectively. Results for *MLP1* are omitted from Figure 79 as they were around 10% of accuracies for all TSs. Such a low performance is probably due to the local minima entrapment of the gradient descent algorithm. The remaining classifiers show larger variations between minimum and maximum classification accuracies achieved particularly for TS_{2-5} with gaps of 23%, 17%, 13%, and 8%, respectively.

Similar to the *Flevo_C* image, it is observed that with larger TSs better overall classification performances are achieved where the gap shrinks to 6-4% among the classifier results for the larger TS with *DT*, *KNN*, *ELM*, and *MLP2* particularly benefiting from the larger TS size. *RF* achieves 78 percent accuracy on TS_2 outperforming the second best classifiers by 7% due to its ensemble nature with 100 trees and feature split of 7 which is related to the larger number of classes. Yet, SVMs reduce this gap by half with two times the size of TS_2 in TS_3. For the applied kernels in the SVMs, they all perform rather constant for the smaller TSs and only a 1-2% difference can be observed for *SVM1*, which is probably related to its linear kernel, which is not sufficient to separate the larger amount of training data for all class pairs in the mapped kernel space.

Compared to the *Flevo_C* image, it can be observed for *Flevo_L* that *ELM*, *MLP2,* and *KNN* resulted in larger networks and smaller numbers of nearest neighbors to accommodate the larger number of terrain classes. Training and testing complexity for *KNN* is the same as for the *Flevo_C* images as number of training and testing samples are approximately the same. This is similar for *ELM* as it is mainly related to the number of nodes in the hidden layer and number of training samples. As expected, the complexity of training *MLP2* obviously increased as hidden layer and, of course, output layer sizes changed.

For the SVMs, computational complexities of the training process are lower particularly for TS_{5-7} compared to the four-class problem, which might be related to the smaller number of training samples per class. This makes it easier in the pairwise multiclass SVM approach to solve the optimization problem even though the larger numbers of classes result in more bina-

ry classifiers. Yet for this reason, testing complexity increases due to a larger numbers of overall support vectors that need to be compared against the test samples.
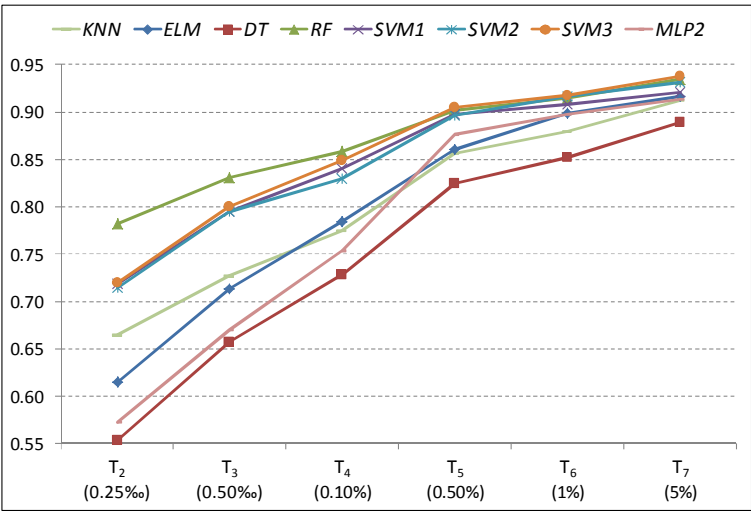
Figure 79 – Average classification accuracies for six different training set sizes over the *Flevo_L* AIRSAR image.

Table 30 – Average training and testing times in seconds for six different training set sizes over the *Flevo_L* AIRSAR image. Best performing configuration are in parentheses.

		KNN	ELM	DT	RF	SVM1	SVM2	SVM3	MLP1	MLP2
TRAINING SETS	**TS$_2$ (0.25‰)**	<0.01 (1)	<0.01 (25)	0.01	0.02	0.01	0.01	0.02	0.88 (44)	0.68 (44)
	TS$_3$ (0.50‰)	<0.01 (1)	0.01 (50)	0.01	0.03	0.02	0.02	0.02	1.01 (44)	0.92 (44)
	TS$_4$ (0.10%)	<0.01 (3)	0.01 (50)	0.01	0.06	0.03	0.04	0.04	1.3 (33)	1.77 (44)
	TS$_5$ (0.50%)	<0.01 (5)	0.29 (400)	0.01	0.27	0.24	0.21	0.26	3.07 (33)	12.25 (44)
	TS$_6$ (1.00%)	<0.01 (5)	0.47 (400)	0.02	0.30	0.53	0.47	0.68	5.72 (33)	36.61 (44)
	TS$_7$ (5.00%)	<0.01 (5)	2.11 (400)	0.08	3.82	8.48	4.69	8.52	30.15 (44)	426.82 (44)
TEST SETS	**TS$_2$ (0.25‰)**	0.44	0.45	0.18	1.49	1.40	1.55	2.08	0.85	0.87
	TS$_3$ (0.50‰)	0.75	0.63	0.18	1.50	2.05	2.34	3.27	0.84	0.87
	TS$_4$ (0.10%)	2.48	0.69	0.17	1.53	2.68	3.02	4.55	0.73	0.88
	TS$_5$ (0.50%)	7.78	4.09	0.18	1.71	5.08	6.42	10.53	0.72	0.86
	TS$_6$ (1.00%)	13.85	4.13	0.17	0.99	10.80	10.10	21.01	0.71	0.86
	TS$_7$ (5.00%)	59.73	4.07	0.17	2.16	30.96	39.04	49.79	0.82	0.87

Overall, SVMs and *RF* are able to outperform other classifiers when it comes to small training data sizes. Particularly the ensemble approach, *RF*, performs best with regard to classification accuracies and computational complexity.

5.2. SMALL TRAINING SIZE PROBLEM AND SEMI-SUPERVISED LEARNING BY ENSEMBLE OF CLASSIFIERS

In the case of supervised learning, combining multiple classifiers to a committee or ensemble has demonstrated to improve classification performance over single classifier systems [158] and its effectiveness has also been shown for remote sensing data [58]. Generally, ensemble learning tries to improve generalization by combining multiple learners, whereas semi-supervised learning attempts to achieve strong generalization by exploiting the unlabeled data. Hence, fusing these two learning paradigms, even stronger learning systems can be generated by leveraging unlabeled data and classifier combination [230]. Zhou and Li proposed the Tri-training approach [229], which can be seen an extension of the co-training algorithms, where three classifiers are used and when two of them agree on a label of an unlabeled sample while the third disagrees; then, under a certain condition, the two classifiers will label this unlabeled sample for the training of the third classifier. Later, Tri-training was extended to Co-forest [123] including more base classifiers adopting the "majority teaches minority" strategy. Additionally, semi-supervised boosting methods have been proposed such as Assemble [15], which labels unlabeled data by the current ensemble and iteratively combines semi-labeled samples with the original labeled set to train a new base learner which is then added to the ensemble. The more generic SemiBoost [133] combines classifier confidence and pairwise similarity to guide the selection of unlabeled examples. Bagging and boosting based ensemble approaches became popular within SSL, particularly self-training, with a general outline illustrated in Table 31; however, they are not as often applied to remote sensing data as for other areas.

Table 31 – General outline of SSL bagging ensemble approach.

- Start with an empty ensemble $\Pi = \varnothing$
- Train a base learner h_0 with labeled data and add h_0 to Π_0
- For each iteration $t = 1 \rightarrow N$:
    - Compute confidence and semi-labels for unlabeled samples using existing ensemble Π_{t-1}
    - Select semi-labeled samples based on a confidence threshold
    - Train new base learner h_t with labeled and semi-labeled samples
    - Add h_t to ensemble $\Pi_t = \Pi_{t-1} \cup h_t$

The general ensemble-based outline as given in Table 31 was utilized by an approach named Semi-labeled Sample Driven Bagging using Multi-Layer Perceptron [33] and k-Nearest Neighbor [34] classifiers over multispectral data. Furthermore, ensembles have been applied to the concept of unsupervised learning where the Cluster-based ENsemble Algorithm

[36] applies Mixture of Gaussians (MoG) and support cluster machine to attack the quality problems of the training samples. In this case, the ensemble technique is used to find the best number of components going from coarse to fine to generate different sets of MoG. A self-trained ensemble with semi-supervised SVM has been proposed in [137] for pixel-based classification where fuzzy C-Means clustering is employed to obtain confidence measures for unlabeled samples, which are then used in an ensemble of SVMs. Here, each SVM classifier starts with a different training set, which might be difficult within a small sample size problem when the initial labeled training data cannot be divided into multiple partitions.

There have been quite many SSL investigations over spectral-based remote sensing data where only a few particularly focused on ill-posed classification of the small sample size problem, which makes the selection of the initial training dataset more critical [24]. However, SSL has not yet been studied in such a high scale that PolSAR data reside particularly when it comes to the evaluation of the classification performance. In this section, the main questions addressed are:

1) How small can the initial training dataset be to achieve still good results, with and without SSL?

2) While applying SSL initially with small size training data, is it possible to reach similar accuracies to a SL approach that is trained over a significantly larger dataset regardless from the number of iterations or unlabeled samples?

With these two questions in mind, focus is on three main investigations regarding the small sample size problem over PolSAR data. Firstly, before applying self-training two unsupervised approaches are considered to enlarge the initial user-annotated training data as an initial stage of the SL. Secondly, a bagging ensemble approach is investigated combining the advantages of a multi-classifier system with semi-supervised learning. Thirdly, different strategies within the self-training procedure are studied on how to select from the pool of unlabeled samples to speed-up the learning process and also to improve both generalization and classification performance.

5.2.1. SSL Ensemble-Driven Approach

In general, semi-supervised learning approaches employing ensemble classifiers are straightforward and proven effective. We adopt the bagging ensemble approach similar to Chi and Bruzzone [33], [34] as the underlying supervised learning approach since such systems are generally classifier independent and advantageous against SVM and graph-based methods regarding memory requirements especially for larger data.

The general outline of a bagging ensemble approach is presented in Table 31. The first step, a base learner h is trained with labeled training dataset, L, and added to the ensemble Π. At step t, the unlabeled data, U, is classified and as a result, semi-labels based on confidence values from Π_{t-1} are obtained. As in [33] and [34], a subset SL_t from U is then extracted, containing the pixels that are (randomly) selected from the unlabeled samples over the entire image for a better spatial distribution. The pixels should have a confidence score above a certain

level (e.g., 85 percent for the multi-layer perceptron, MLP) and a total number of twice the amount of labeled samples is selected. Then a new base learner h is trained using L and SL_t, with h added to Π_t, i.e., $\Pi_t = \Pi_{t-1} + h(L, SL_t)$. This is an iterative process until a predefined number (such as 20) of classifiers in Π is reached. They employed k-Nearest Neighbor and MLPs as the base learners and penalized the unlabeled samples in step t using the confidence values obtained from the previous ensemble Π_{t-1}. This is done so the semi-labels selected among the unlabeled samples do not have the same influence during training as the labeled samples. For the k-Nearest Neighbor, the penalty is applied when the nearest neighbors are compared while classifying a sample, whereas in case of Multi-Layer Perceptrons, they modified the mean squared error cost function instead.

Table 32 – The outline of the adapted semi-supervised bagging ensemble approach, where red highlights the modifications made to the general approach.

- Start with an empty ensemble $\Pi = \varnothing$
- [1. Extend initial training data using spatial consistency assumption around the labeled data]
- Train a base learner h_0 with labeled data and add h_0 to Π_0
- [2. Use a small ensemble as the base learner itself]
- For each iteration t=1→N:
  - Compute confidence and semi-labels for unlabeled samples using existing ensemble Π_{t-1}
  - [3. Select unlabeled samples only from a certain search neighborhood]
  - Select semi-labeled samples based on the search neighborhood and confidence threshold
  - [Modify search neighborhood based on growing criterion]
  - Train a new base learner h_t with labeled and semi-labeled samples
  - Add h_t to the ensemble $\Pi_t = \Pi_{t-1} \cup h_t$

The aforementioned general bagging ensemble approach within a self-training process is executed and extended in this semi-supervised learning setup. Particularly, three modifications as highlighted in Table 32 are proposed to improve the classification accuracy and to reduce the number of self-training iterations required. Figure 80 illustrates the four-stage iterative SSL approach with each stage and contribution detailed in the following subsections.
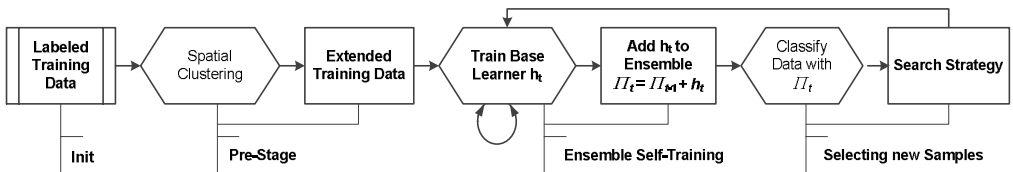
Figure 80 – Flowchart illustrating the four-stage SSL approach.

5.2.1.1 Init: Labeled training data

The initial labeled training dataset is critical for semi-supervised learning techniques [24]. To validate this, the used training sets T_i are generated in the following manner. Training set T_1 has exactly one (i=1) labeled sample per class. In the Flevoland AIRSAR image (Section 2.5.1) are 15 terrain classes (nC=15), so that T_1 has a total size of 15. Furthermore, 10 instances of T_1 are randomly generated. Now the instances of training set T_2 are created by randomly adding a new and different labeled training sample to each of the 10 instances of T_1 so that $T_1 \subset T_2$ for all instances. Every instance of T_2 now includes two (i=2) labeled samples per class. This process continues up to i=10 so that $T_1 \subset T_2 \subset \ldots \subset T_9 \subset T_{10}$ until 100 training datasets with sizes from 15 to 150 are generated.

5.2.1.2 Pre-stage

As a pre-stage, unsupervised clustering is employed to tackle the small training set problem, which is a regular starting point in a SSL scenario. The main idea is that any option that is able to extend the training set accurately would be highly beneficial since a better generalization and hence a superior classification performance can be achieved over a larger training set as also observed in the evaluations of Section 5.1. Here straightforward approach is to use the contextual information within the pixel neighborhood of the labeled samples and assign the same label to the neighbors. By employing this contextual information in form of the 4- or 8-connected neighbors, local spatial smoothness and consistency among the image pixels is exploited. This way the initial training set can be easily enlarged by 4- or 8-times with a high probability of the semi-labeled neighbors having the correct label. To further increase the number of training samples, a dense over-segmentation of an image is computed applying a *superpixel* [120] segmentation approach. This segments the image into small homogenous regions, the so-called superpixels [120], respecting local image boundaries, while limiting under-segmentation through a compactness constraint. Again, this is a spatial smoothness and consistency among pixel intensities. Compared to the connected neighbors approach, this may properly extend the initial training set by an order of magnitude depending on the average size of the obtained superpixels. However, the outcome will be parameter dependent with respect to the size and compactness of the superpixel algorithm, which might also affect the accuracy of the semi-labels.

During this pre-stage, each T_i is enlarged by the 8-connected neighbors around each labeled pixel (NN_i, i.e., see left side of Figure 81) and by the pixels belonging to the same superpixel as the labeled pixels (SP_i). For the superpixel segmentation, TurboPixels algorithm [120] is employed with the (maximum) number of superpixels set to 8000, which is empirically determined to get compact and homogeneous superpixels. With this parameter setting, the algorithm will produce 7434 super pixels with an average size of around 103 pixels with the outcome shown in Figure 81 on the right side. As an alternative approach, a more recent algorithm [1] could also be used instead where one just needs to specify the *desired* superpixel size and its compactness rather than the number of superpixels. Note that the choice of the

superpixel algorithm is not critical for this study. Figure 82 shows an example of the extended training sets NN_i and SP_i on an instance of T_5 over a selected area.

Figure 81 – The contextual 8-connected pixel neighborhood (left) and (right) the result of the applied superpixel algorithm (Turbopixels [120]) over Flevoland AIRSAR image.

Selected Area for Illustration   Pixels   NN   SP
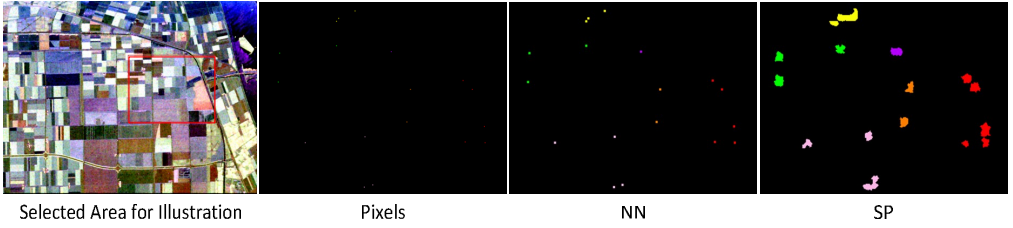
Figure 82 – Example of the different training sets over a selected area (red box) based on contextual information (NN) and superpixel (SP) extensions.

5.2.1.3 Ensemble-based Self-Training

Within this self-training process, a bagging ensemble is employed as the underlying supervised learning approach, which relies on a base classifier. It is commonly known that employing a strong classifier is advantageous for any supervised learning. Therefore, the fact can be exploited that combining multiple classifiers to a committee or ensemble has shown to improve classification performance over single classifier systems [158]. As a base classifier within the self-training stage, a rather weak classifier is employed specifically a Decision Tree (DT) algorithm [19] due to its simplicity and parameter independence. Additionally, Random Forest (RF) [19] is used as a multi-classifier system of DTs in order to obtain diversity due to its employed feature splitting. Since the overall approach is already based on an ensemble of classifiers, RF will only include three DTs to keep computational complexity low.

Due to its simplicity, the employed DT algorithm provides binary class decisions d_c for an individual sample as $d_c=1$ if class c is chosen, otherwise $d_c=0$. As an ensemble, RF uses the individual DT class decisions to provide its final class predictions via *majority voting* [158]:

$$D_c = \max_{j=1}^{nC} \sum_{n=1}^{M} d_j \,, \qquad (21)$$

where nC is the number of classes and M is the number of classifiers, in this case $M=3$. Mathematically, this approach can also be applied to DT with $M=1$. Now during the self-training, the class confidence values of ensemble Π, at iteration t, is the combination of all individual ensemble member predictions D_c via the *mean rule* [158]:

$$\mu^t = \frac{1}{t} \sum_{i=1}^{t} D_c^i \,. \qquad (22)$$

As for the iterative self-training procedure, a confidence threshold, THR, shall be used for the class confidence values, where THR indicates the minimum class confidence value a sample should have assigned by the previous ensemble classifier to select it from the unlabeled samples. On one hand, if THR is too high, limited or no new information is introduced into the learning process, which results in no or limited learning during each self-training iteration. On the other hand, with THR too low, there is a risk of introducing too many erroneous semi-labeled samples to the classifiers. This whole approach is contrary to active learning, where samples with confusing class membership values are selected (i.e., samples lying on or close to the decision boundary) since a human expert will provide a correct label. Within self-training, one has to weigh the risk of adding new information and classifier's confidence. Moreover, related to the small sample size investigation, when selecting the number of unlabeled samples, N_{SL}, the same number as originally labeled training samples is considered per class during self-training iterations. This guarantees that the number of labeled samples is always equal or greater than the added semi-labeled samples; and it is not biased towards possible erroneous semi-labeled samples particularly during the earlier self-training iterations. Accordingly, the following procedure is adapted:

1. If no unlabeled sample has a class membership value higher than THR then no unlabeled samples are selected for that particular class.
2. If the number of unlabeled samples with a class membership value equal or higher than THR is less than N_{SL}, all of them are selected. Hence, N_{SL} is the maximum number than can be selected per class from the unlabeled samples.
3. Otherwise, N_{SL} number of samples is selected among the unlabeled samples with class membership values higher than THR.

The semi-labeled samples are randomly selected with uniform distribution among the samples fulfilling the criterion of confidence values higher than THR. The reason for random selection is twofold: By selecting samples from the top of the class confidence values, we would only select samples that we can already classify correctly. Alternatively selecting samples with class confidence values slightly higher than THR, there is obviously a higher chance of adding new information into the learning process yet also a higher probability of introducing erroneous samples. However, making errors in earlier stages of the self-training process may cause accumulation of errors over time. Random selection among the unlabeled samples combines the advantages of selecting samples with different class confidence values while reduce

the risks of the two aforementioned selection scenarios. Moreover, the random selection will add certain diversity among the samples that can enhance the learning process. Note, that it is also a common practice to include other measures, such as clustering samples in feature space or determine diversity among samples to avoid the selection of redundant samples. However, this has not been considered to avoid the high computational complexity and large memory requirements.

Both labeled and semi-labeled samples are treated equally during the training. Such a treatment is acceptable since the class confidence value threshold is kept reasonably high; however, when the semi-labeling is wrong even with a high confidence value then nothing can indeed be done to cure this. On the other hand, the ensemble approach can still compensate for few erratic individual classifiers when some semi-labeled samples are introduced with wrong semi-labels. Moreover, base learners can be applied "as is" without any need of modification to make up the erroneous semi-labeling. Overall, we have chosen 50 iterations during the self-training procedure as one of the objectives is to investigate the effects of how many iterations and classifiers (i.e., how many unlabeled samples) can be added to the ensemble while still providing additional information to the learning process compared to predefined numbers such as 17 and 25 in [33] and [34], respectively.

5.2.1.4 Selection of New Training Samples

In any self-training approach, a significant improvement over the classification performance can usually be achieved only by selecting a reliable set of new training samples from the large pool of unlabeled samples. Therefore, rather than selecting them from the entire image excluding the ground truth, the search neighborhood can be limited to the vicinity of the provided labeled samples. This neighborhood exploits a spatial smoothness constraint at the beginning and the area of which can be increased with each iteration. In the proposed approach the *how* and *where* are discussed to select unlabeled samples in the following way.

The *how* is usually measured by the (class) confidence values provided by the classifier, i.e., computing the confidence score of each sample belonging to a particular class. Via a confidence threshold (THR) applied to the class confidence values it can be determined which unlabeled samples should be selected among the unlabeled data. The rule of thumb is that samples above a certain class confidence value are selected to ensure not adding and accumulating too much error as previously discussed.

The *where* indicates the search for the location to select new training data *after* applying the confidence threshold, THR = 0.8. This is usually performed over all available unlabeled samples to exploit information presented in the entire data or image. This is a valid approach; however, due to the large amount of unlabeled samples, the selection of the most reliable candidates becomes more difficult especially when we want to add new information to the (self-training) learning process. Due to nature of remote sensing data, the spatial location in the close vicinity of the provided labeled data can be exploited rather than performing some feature clustering methods with unknown distance metrics, both of which can create further uncertainties or erroneous training data selections in the process.

Thus, the main idea is to grow the search neighborhood with the notion that the provided labeled data are correct exploiting the smoothness and consistency assumptions while focusing the selection process among the unlabeled data within vicinity of the labeled data. This search strategy has the advantage that the initial classifier may find such unlabeled data that do not necessarily have the highest class confidence values but are able to provide more diversity among the training samples. Especially at the beginning of the self-training process, we try to increase the size of the training set with *reliable* and *informative* semi-labeled samples. Therefore, two options are considered: 1) the search neighborhood is limited to the vicinity around all labeled samples to determine those semi-labeled samples among the unlabeled samples per class - NH_L. This can be seen as a localized version of selecting from the entire image. In addition, 2) setting the search neighborhood around the labeled samples of a particular class to determine the semi-labeled samples per class - NH_C. The search area of those two strategies can iteratively grow with the number of SSL iterations to cover the entire image eventually.

To choose unlabeled samples, three strategies according are considered and evaluated with regard to THR and N_{SL}. Firstly, the basic strategy uses all unlabeled samples for selection (*full*). The other two strategies exploit, as mentioned above, the spatial location of the labeled data so that the unlabeled samples are limited by the neighborhood of the labeled samples (NH_L) or by the labeled samples of a particular class (NH_C). Two different methods are applied to generate NH_L and NH_C. The first applies an initial circular pixel neighborhood with radius *rad* growing around each labeled sample. Its radius is gradually increased by *radInc* pixels in self-training iteration t based on the equation: $NH^t = rad + (t \cdot radInc)$. The second method utilizes the available superpixel segmentation with the following idea: instead of growing the search neighborhood by a *radInc*, it now grows by merging adjacent superpixels to the previous neighborhood starting from the superpixel belonging to the labeled samples. In the end, NH_L and NH_C actually cover the same area with the difference that NH_C is further separated into the individual classes. Synthetic examples of NH_L and NH_C of the initial search neighborhoods are shown in Figure 83 with *rad*=10 and *radInc*=1 due to the limited resolution of the test image. Furthermore, using the superpixel approach as demonstrated in Figure 84, the neighborhood grows quite rapidly, therefore, reaching the equivalent of *full* within around t=10 ST iterations. Accordingly, an option is considered where the superpixel-based search neighborhood only grows per n-th ST iteration to limit the exponential growth. However, utilizing superpixels could be seen more generic without having to "tweak" parameters *rad* and *radInc* of the circular growing search neighborhood.

Within the evaluation, the effects of circular NH_L and NH_C are investigated considering four combinations of *rad* 10 and 20 with *radInc* of 1 and 2, namely 10_1, 10_2, 20_1, and 20_2. For the superpixel NH_L and NH_C, two tests are conducted, where besides growing the search neighborhood with each ST iteration, the SP search neighborhoods is only updated every 2nd (odd) iteration.

| full | Circular NH$_L$ | Circular NH$_C$ | Superpixel NH$_C$ |
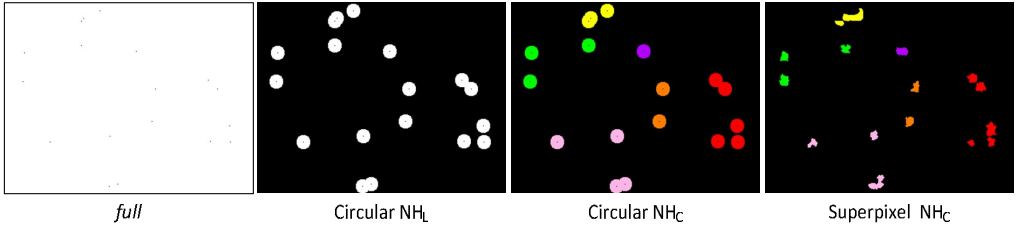
Figure 83 – Examples of the different initial search neighborhoods for the unlabeled sample selection over the selected image area. The colors for NH$_C$ indicate the class label.

Iteratively Growing (Circular NH$_L$)   Iteratively Growing (Superpixel NH$_C$)   Example of Iteratively Growing via Superpixels
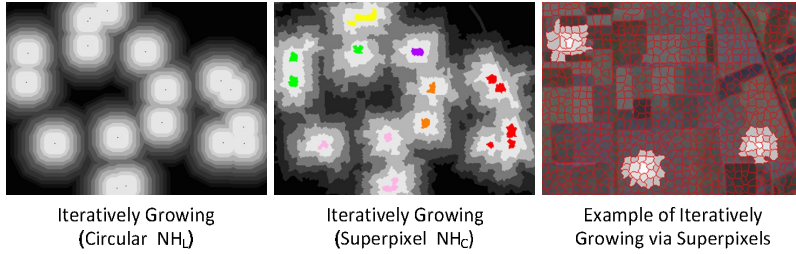
Figure 84 – Examples of the growing process for the circular- and superpixel-based search over the selected image area. The colors for NH$_C$ indicate the class label. The darker shades of gray indicate the growth in the ST iterations with black areas not considered for selection at all.

5.2.2. Experimental Results

In the following experiments, the main questions are:

1) Can a SSL method starting with a limited training set, T_i, manage to reach a similar performance of a classifier trained over a larger training set?,

2) How does self-training over T_i measure compared to the SSL approaches by enlarging the training set in an unsupervised manner using NN and SP?,

3) What is the influence of search neighborhoods *full* versus NH$_L$ versus NH$_C$ and the relation of number of unlabeled versus labeled samples over such search neighborhoods?

To find answers to these questions, the experimental setup is similar to Section 5.1.1, extracting the same 11 dimensional feature vector as described in Section 2.6 - FV$_2$ = [*Span, H, A, $\bar{\alpha}$, $\bar{\beta}$, $\bar{\delta}$, $\bar{\gamma}$, λ_1, λ_2, λ_3, RVI*] over the Flevoland AIRSAR image.

Before presenting the classification results over the initial training sets, T_i, using self-training with the bagging ensemble approach, we shall first investigate the effects of the confidence threshold, the different search neighborhoods and the number of unlabeled samples selected. Furthermore, we shall evaluate and compare the performances of the enlarged training sets via the unsupervised SSL approaches, NN, and superpixels, SP. Along with numerical performance evaluations represented as average classification accuracies over the 10 instance for the different training sets size T_i; we shall also present visual classification results.

For evaluation of the effects with respect to the confidence threshold THR, a basic setup is used with the three search neighborhoods, *full*, circular NH_L, and circular NH_c employing DT and RF within the ensemble based self-training. Three values are used for THR: 0.7, 0.8, and 0.9. Individual results are shown in Figure 85 as average classification accuracies achieved over the smaller and larger sized training sets. Using RF, the different THRs perform on a similar level and differences among search neighborhoods with minor variations in the final classification accuracies, due to being a stronger classifier providing better class predictions over smaller training sets. However, the weaker classifier DT is more affected by the choice of THR as one probably would expect. The main observation is that the THR has an effect on the final classification performance with respect to the underlying search neighborhood. The DT performance regarding THR seems to be proportional to the size of the used search neighborhood. With a smaller THR is seems beneficial to have a smaller search neighborhood, whereas with higher THR the size of the search neighborhood does not seem to have major effects as performances vary just within a small margin. This is expected as the weaker learner DT is not able to learn and generalize too well from such tiny to small training sets. Due to these observations and as the overall classification performances for the different THRs average out over the different search neighborhoods, so that for the remainder of the experiments THR is equal to 0.8.
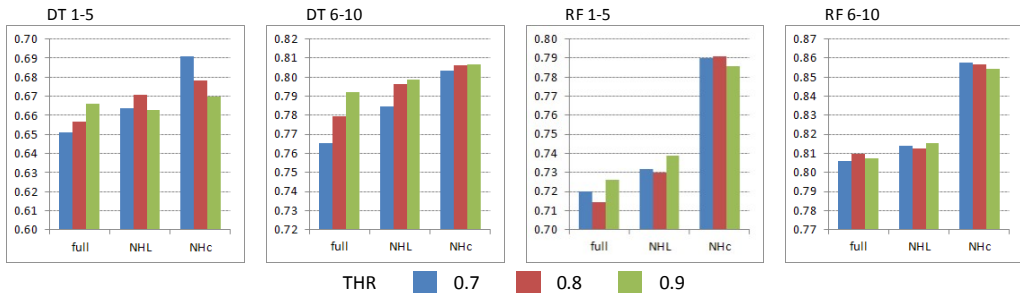
Figure 85 – Illustrating the effect of different confidence threshold (THR) values as average classification accuracies over the three search neighborhoods using DT and RF as base learners within self-training.

Regarding the evaluation of the different search neighborhoods (SNHs), the overall differences in total gain after 50 iterations of ST is minimal with all SNHs reaching similar classification results (Figure 86) and ST improvements (Figure 87) for different T_i. We can see that especially with the small T_i (T_1-T_5) larger ST improvements (30-40%) can be obtained compared to their initial lower classification accuracies indicating that there are large potentials for improvements. It is also anticipated that the ST improvements with the larger T_i (T_6-T_{10}) are no longer that significant (only around 10-15%) as their initial classification accuracies are already around 60-70 percent due to larger training data. In general, differences among the SNHs *full*, NH_L, and NH_C are observed as expected. Applying NH_L results in slightly better classification performances than using *full* since NH_L is a smaller subset of *full* whereas NH_C limits the SNH for one class to the spatial proximity of its particular labeled samples. The performance difference between *full* and NH_L disappears for the larger sets, T_{5-10}, since NH_L suf-

fers from the same problem, i.e., no or limited amount of new information is available due to larger number of labeled samples.

Figure 86 – Classification accuracies for different training sets T_i using the three selection methods for the unlabeled samples over the four circular combinations and two superpixel methods using Random Forest as base learner in the ensemble self-training.

For the circular NH_L and NH_C, the initial size rad and the incremental expansion by *radInc* seems to have marginal effects since in either case the SNH area will not significantly change per iteration. Overall, the four different combinations result in rather similar outcomes with marginal variations due to the random sample selection. Based on the made observations, the influence of parameters rad and *radInc* are related to the resolution of the used AIRSAR im-

age. Considering the two SP growing methods, in both cases results using NHL are similar to full due to the fast rate the SP SNH grows. However, main differences can be observed for NHC. Firstly, both methods results are below the ones obtained by circular growing, and, in either case, no ST improvements for larger Ti are achieved after 10 iterations. Hence, within the first 10 iterations, the best ST performances are achieved. Thus, afterwards no further benefits of adding new samples can be made with NH_C since the SNH area is the same size as *full*.
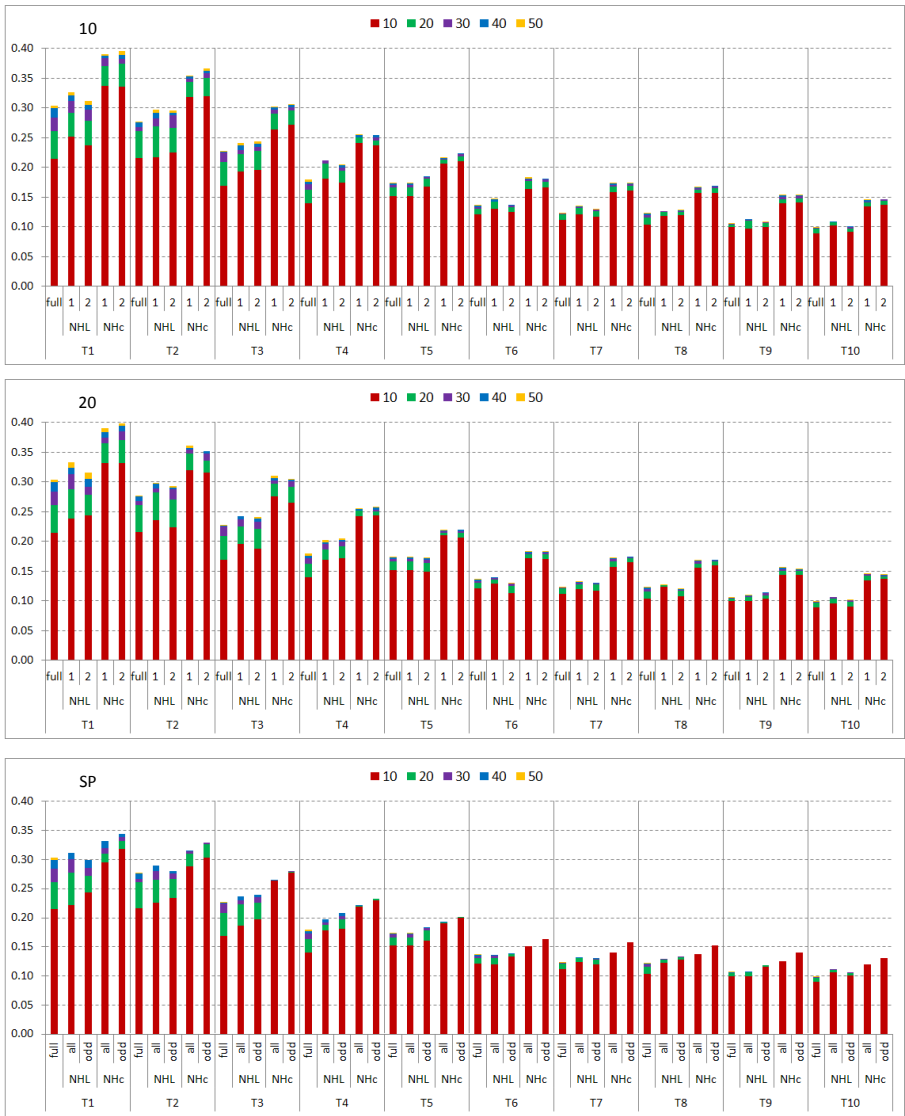
Figure 87 – Improvements of self-training per 10th iteration for different training sets T_i using the 3 selection methods for the unlabeled samples over the four circular combinations and two superpixel methods using Random Forest as base learner in the ensemble self-training.

For the main classification performance evaluations, the NH_C approach is used for the circular combination with *rad*=10 and *radInc*=1. This mimics a slower growth while the superpixel approach will grow much faster while expanding the SNH every odd ST iteration. For the remainder of this evaluation, two SNH approaches are abbreviated with NH_o for the circular and NH_{sp} for the superpixel methods.

Next, the effect of the number of unlabeled samples that are added per ST iteration is investigated. Due to small sample number per class, the same amount of unlabeled samples with regard to the number of labeled samples per class is considered for initial pixel training sets T_i. This avoids the possible bias towards erroneous semi-labeled samples particularly during the earlier self-training iterations. Thus, we evaluated the effect of adding different number of unlabeled samples per ST iteration for enlarging T_i with NN and SP to see if they would able to handle such bias. For this, multiples (*xL*) of labeled samples per class are considered, namely *x1*, *x2*, *x4*, and *x6* that can potentially added in numbers of unlabeled samples. This means, for example, that in case of *x4*, unlabeled data size is up to four times the number of originally labeled samples per class. Thus, with i=2, this results in eight possible candidates if their confidence scores are higher than THR.
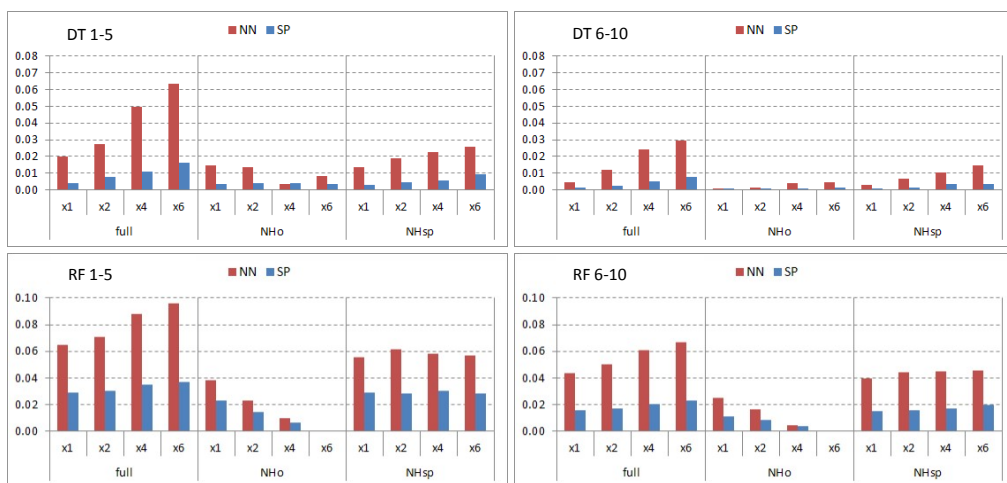
Figure 88 – Influence of number of unlabeled samples added per self-training iteration using the class-based SNH in a circular (NH_o) and superpixel-based (NH_{sp}) growing approach for the two enlarged training sets NN_i and SP_i.

Figure 88 demonstrates the differences of search neighborhoods (NH_o and NH_{sp}) and number of *xL* combinations with respect to the best classification result achieved among all combinations. The general observation confirms the expected result, that is, adding more unlabeled samples than the initially labeled samples will bias the learning process towards the unlabeled samples particularly in case of the smaller training sets, i.e., T_{1-5}, as shown in the plot for Decision Tree (DT) over NN_i using SNH mode, *full*. However, this effect is reduced for T_{6-10} due to the relatively larger size of the initial training data. When using NH_{sp}, it shows the same behavior for different *xL* combinations as it will reach the area of *full* since the search

process quickly suffers from the same issues. Yet note that the effects are not as severe since the SNH still has to grow within the first ST iterations, which reduces the chance of the weak DT learner to add erroneous or uninformative samples during the first ST iterations. Applying NH_o appears to provide best results with $x4$ combination being a trade-off to $x2$ and $x6$ as a balance between the number of labeled and unlabeled samples. Yet classification accuracy differences among them are rather small, within one percent. For the larger training sets of $NN_{6\text{-}10}$ using NH_o, there seems to be no benefit of adding more unlabeled samples due to larger number of labeled samples that are already available and providing initial classification accuracy level higher than 70 percent. In that regard, performance differences for $x4$ and $x6$ to $x1$ and $x2$ combinations are negligible. Concerning the superpixel enlarged initial training sets, results using SNHs, *full* and NH_{sp} seem to follow similar behavior for different xL combinations. However, classification accuracies are achieved within a one percent range due to the larger training set sizes and this makes it easier to compensate for larger number of unlabeled samples. In case of SP_i, the number of unlabeled samples does not significantly affect the results, as the labeled data size during the initial training iterations is so large that only minor ST improvements can be made.

Regarding the evaluation using Random Forest (RF), we can observe similar behavior for the different training sizes. As for DT with NN_i using the SNH mode as *full*, more unlabeled samples result in higher probabilities of selecting erroneous or uninformative samples due to the larger SNH. For NH_{sp}, due to the homogeneous superpixels, the size of unlabeled samples has a marginal effect because the additional samples over the same superpixel area will have a rather similar feature structure especially with a stronger classifier that is capable of providing higher confidence scores. Nonetheless, we can observe that for NH_o the opposite effect is visible, where more samples can now make a significant influence. One of the reasons is that the circular NH grows with respect to the spatial distance to a labeled sample location - not as in case of superpixels by sample homogeneity. More important than this is the fact that with NH_o the SNH grows slower so that not all similar samples are added within the same ST iteration. The addition of similar samples is spread over time and many ST iterations. For the enlargement of T_i by superpixels as for DT, similar observations can be made for $NN_{6\text{-}10}$ and SP, where the effect of the unlabeled samples is reduced with higher number of labeled samples.

After these two initial evaluations, the classification performances of T_i and their corresponding self-training (ST) improvements for search neighborhoods *full*, NH_o, and NH_{sp} are presented in Figures 89 and 90. Using DT as the base classifier, we can observe that similar levels of classification accuracies can be achieved using self-training over T_1, T_2, and T_3 compared to the initial results of T_4, T_7, and T_{10}, respectively. Similarly, when using RF with the sets T_1 and T_2 employing self-training, classification results comparable to T_5 and T_{10} can be realized. This is not surprising because a stronger base classifier such as RF can achieve higher classification accuracies particularly for small training sets such as T_1 and T_2 as observed in the previous Section 5.1. This will eventually result in better label predictions of the unlabeled samples in the first ST iterations. Note that for both base classifiers, the classification accuracies achieved by self-training are similar to the ones obtained with its SL counter-

part while using 3-5 times less amount of manually labeled data. Such a crucial reduction on the manually labeled data for training is indeed a noteworthy accomplishment of the ST approach; however, further investigations shall be carried out to evaluate different options and to maximize the gain.

Evaluating the three different methods on *how* and *where* to pick the unlabeled samples from, we can observe clear differences for the two base classifiers employed. For the weaker classifier DT in Figure 89, the *full* search neighborhood provides slightly better or similar results than NH_o and NH_{sp} for T_1 and T_2, whereas for the larger sets, T_{3-10}, NH_o and NH_{sp} yield in higher accuracies. The reason lies in the fact that DT, as a rather weak learner is not capable of learning from such small sample sizes with one and two labeled samples per class. For T_1, note that DT still benefits from new samples during the iterations, $t=40$ and $t=50$. However, er, ST improvements of 18-22% are achieved due to the extremely low initial classification accuracies on the labeled samples while the overall classification accuracies being below 50%.

Figure 89 – Top row: Classification accuracies for different training sets, T_i using the three selection methods for the unlabeled samples. Dashed lines show results for enlarging T_i with NN and superpixel (SP). Bottom row: Improvements of self-training (ST) per 10th iteration for the Decision Tree base learner.

As for the larger sets, T_{3-10}, the larger training data size results in a greater search neighborhood while increasing the number of possible unlabeled candidates within NH_o and NH_{sp}. Yet the number of these candidates is still significantly smaller than the one for *full*, and this

yields a higher chance to semi-labeled samples with lower class confidence values providing more diversity but in the same time, higher risk of erroneous semi-labeling. Yet, performance differences observed among different search neighborhoods are minimal, i.e., using NH_o provides a mere ~2% higher classification accuracies. As for RF in Figure 90, classification performances over T_{1-4} using NH_o are the highest among all other alternatives. This is related to the stronger classifier that has superior learning capabilities with less number of samples so that the smaller search neighborhood of NH_o and NH_{sp} becomes beneficial in providing further diverse samples into the learning process. Both base classifiers indeed benefit from selecting samples closer to the initially labeled samples while having a stronger classifier with a better learning ability is obviously more advantageous.

Figure 90 – Top row: Classification accuracies for different training sets, T_i using the three selection methods for the unlabeled samples. Dashed lines show results for enlarging T_i with NN and superpixel (SP). Bottom row: Improvements of self-training (ST) per 10th iteration for the Random Forest base learner.

Furthermore, when looking into results of the self-training iterations in Figures 89 and 90, we can observe that using NH_o and NH_{sp} can yield further improvements particularly during the earlier iterations (i.e., t=10). With the *full* search neighborhood, a minimum number of 20-30 iterations are needed to achieve the same results or for a larger T_i, similar accuracies cannot be achieved even after 50 iterations when using DT as the base classifier. The same behavior can also be observed for RF, where results obtained after 10 iterations applying NH_o and NH_{sp} outperform accuracies achieved after 50 iterations applying *full* as the search neigh-

borhood. Note further that performing self-training over these initial training sets achieves similar or better results than using NN_i or SP_i, respectively. This is because the neighborhood, *full*, can result in larger numbers of unlabeled samples with high classifier confidence scores that are greater than THR. Hence, there is a greater chance for the highly accurate semi-labeled samples to be selected; however, this yields adding no or less new information into the self-training process. In case of NH_o and NH_{sp}, the number of possible unlabeled samples as new candidates is limited, therefore, giving a higher chance of adding more diversity due to the samples with a lower confidence score. When using NH_{sp}, the best performance is achieved within the first 10 iterations, after that its SNH area grows beyond *full*. Since the classification accuracy with NH_{sp} after 10 iterations is already better than the one achieved with *full* using 50 iterations, no further benefits of adding new samples can be observed with NH_{sp}.

As each training set T_i consists of 10 different instances, the crucial effect of the starting samples for the two smallest sets T_1 and T_2 is illustrated in Figure 91 over the first 20 ST iterations. The plots show that for various instances DT struggles with the small number of labeled samples to achieve improvements via ST. As mentioned earlier, the initial labeled samples are critical to determine the success of applying SSL particularly for such a weaker learner as DT. It can be noticed that using RF (as a mini-ensemble of three DTs) overcomes this problem and significant improvements are achieved within the first 10 ST iterations. Furthermore, as T_1 is a subset of T_2, it can be seen that the one additional sample per class has a positive influence yet will not always overcome the weakness of the first sample or might even have a negative effect.
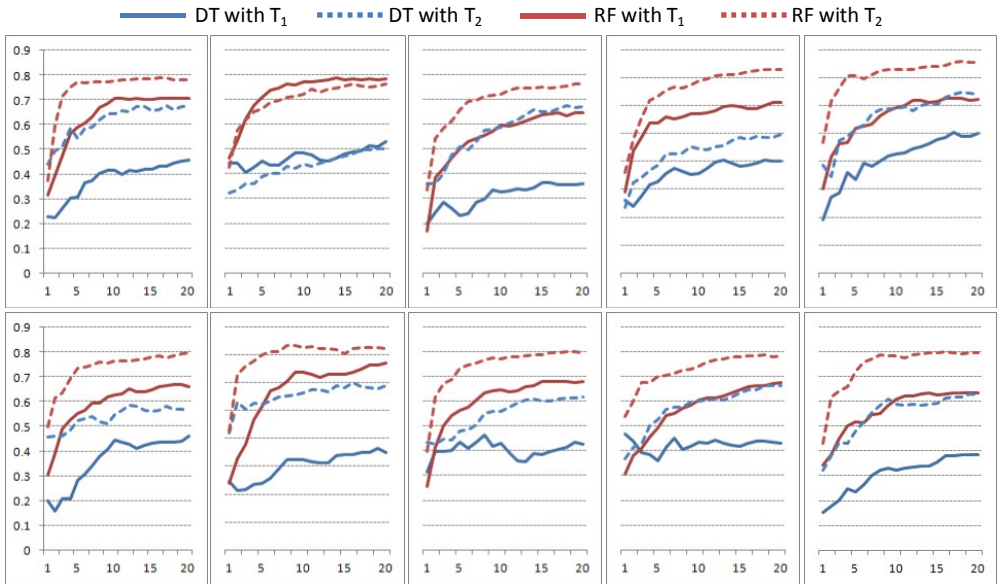
Figure 91 – Classification accuracies of the 10 instances of training sets T_1 and T_2 over 20 iterations of self-training for the two base learners Decision Tree (DT) and Random Forest (RF).

When enlarging T_i to NN_i, the initial classification performance is improved as illustrated in Figure 92, which is anticipated due to the availability of more samples. As the ground truth is available, it could be verified that 96.3 percent of the NN_i samples are correctly labeled whereas the majority of the remaining 3.7% is unknown since no ground truth is available on those areas. Even though, it is expected that NN_1 and T_9 results are not comparable because the NN_1 samples will have a similar feature structure providing less diversity among the samples compared to the labeled nine samples in T_9. This is also observable for the NN_{2-4} results as they are not able to match the initial classification accuracy of T_{10} besides NN_{2-4} being significantly larger.

The initial classification improvements on the average are ~18% and ~10% for NN_{1-5} and NN_{6-10}, respectively. Note that employing self-training is still able to provide an increase in the classification accuracy over all NN_i yet the improvements are getting insignificant as the initial accuracies are higher (see Figures 92 and 93). At the end, this results in classification performance employing NH_o/NH_{sp} being just marginally better by 0.5% for DT and ~2.5% for RF, on the average than using *full*.
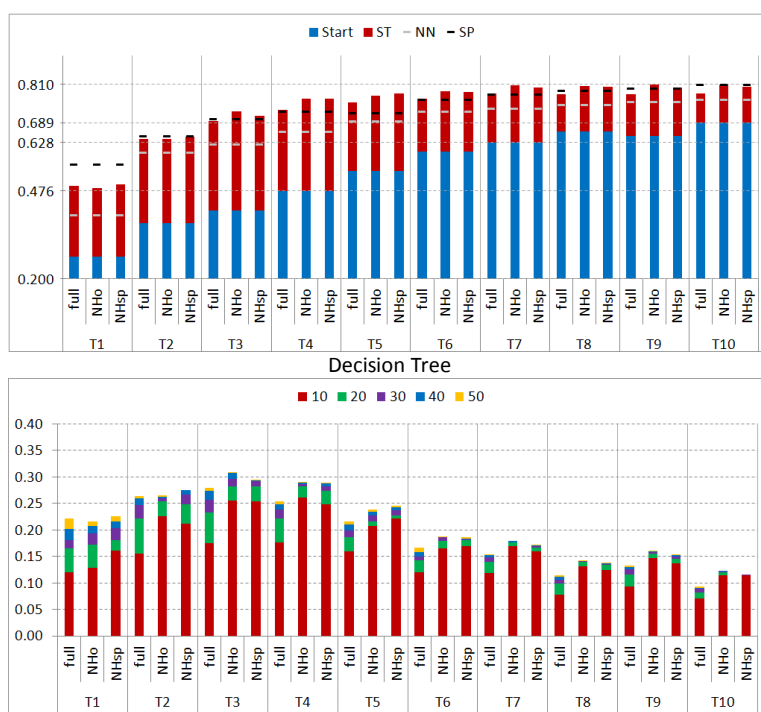

Figure 92 – Top row: Classification accuracies for different training sets, NN_i, using the three selection methods for the unlabeled samples. Dashed lines show results for enlarging T_i superpixel (SP). Bottom row: Improvements of self-training per 10th iteration for the Decision Tree base learner.

When DT is used as the base classifier (Figure 92), the main performance difference compared to the results with T_i is that NN_i is highly beneficial for labeled training set with i=2

when improving the final classification accuracy by 6-7%. Similar observation can be made for RF (Figure 93) trained over the NN_1 where an accuracy improvement of around 6% is visible for the search neighborhood *full* compared to their T_1 results. However, employing NH_o and NH_{sp} over both training sets T_1 and NN_1, the difference shrinks to 2%. For training sets larger than T_3, the performance difference between the application of T_i and NN_i is minimal. The reason for this is that both NN_i and NH_o/NH_{sp} enhance the initial training set T_i based on the same idea: by selecting unlabeled samples from the close neighborhood of the provided labeled samples.
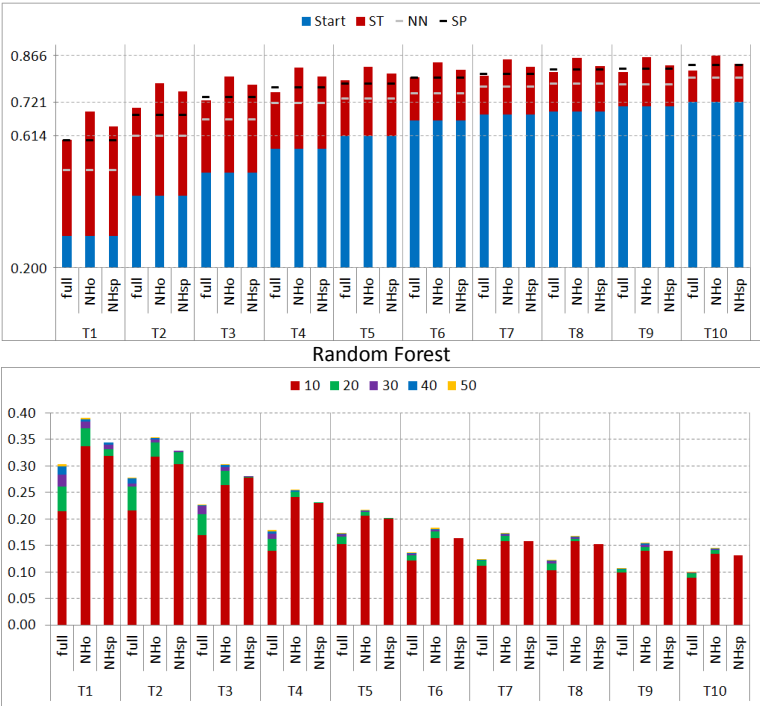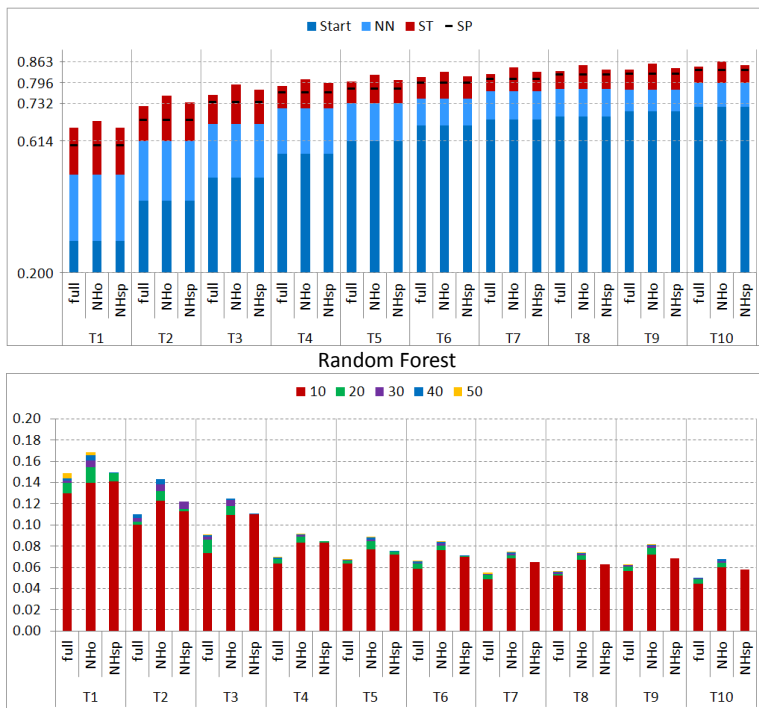

Figure 93 – Top row: Classification accuracies for different training sets, NN_i, using the three selection methods for the unlabeled samples. Dashed lines show results for enlarging T_i superpixel (SP). Bottom row: Improvements of self-training per 10th iteration for the Random Forest base learner.

Similar observations and comparative evaluations between NN_i and the enlarged training sets SP_i via superpixels can be made. As visible in the plots given in Figures 94 and 95, the classification accuracy over the SP_1 can be improved by 15% and 11% for DT and RF, respectively, compared to NN_1 whereas performance differences for the other training set sizes are getting less. Similar to NN_i, such improvements occur because of significantly larger training set size for self-training, i.e., around 100 semi-labelings per labeled pixel. However, when verified with the ground truth, it can be noted that the accuracy of the correctly labeled superpixel samples SP_i is lower than the one for NN (around 10%), yet there are still at least 86 of 100 samples with the correct labels whereas the other 14% are mainly unknown. It is

obvious that the significantly larger number of samples compensates the relative drop in the semi-labeling accuracy. Moreover, the superpixel method already covers most potential candidates within the vicinity of the initially labeled samples and thus reduces the effect of NH_o and NH_{sp}. When the search strategies *full* is employed the amount of new information is quite limited and note that it is now further reduced due to the larger diversity already introduced by the larger training data. Hence, this significantly reduces any potential performance gain. The same effect can also be observed for the larger T_i or NN_i training sets. This means that an upper bound of classification accuracy can be achieved by employing different sized training sets enlarged by connecting neighbors or superpixels.
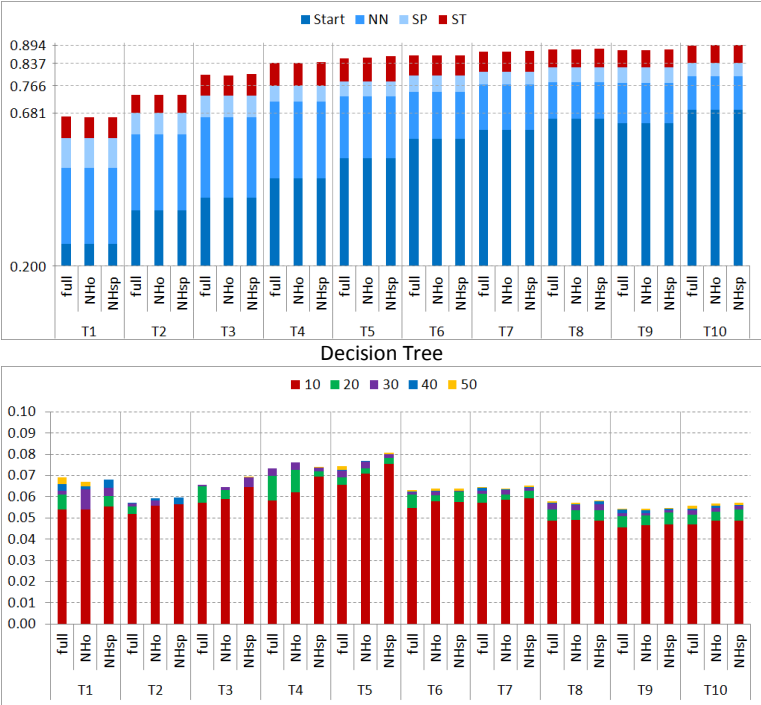
Figure 94 – Top row: Classification accuracies for different training sets, SP_i using the three selection methods for the unlabeled samples. Bottom row: Improvements of self-training (ST) per 10th iteration for the Decision Tree base learner.

Over both neighborhood approaches NH_o and NH_{sp} of the proposed self-training method, comparative evaluations are performed among the three training set types. Figure 96 presents the plots that sum up the differences of the classification accuracies obtained by individual NN_i and SP_i approaches with respect to their initial training set, T_i. When using DT as the base classifier, the neighborhood superpixel approach contributes by at least 5% to the classification accuracy for the most of the training sample sizes used whereas the highest gain is achieved when the two smallest training sets are enlarged by their 8-connected neighbors. When using RF as the base classifier, improvements are observed for all sets over the search neighborhood *full* where accuracy differences over NN_i and SP_i are around 2% and 6%, respectively. It can be observed that when exploiting the closer spatial neighborhood of labeled

samples via NH_o, neither the 8-neighbor contextual information approach nor the superpixels approach leads to any significant performance improvement due to aforementioned reasons.
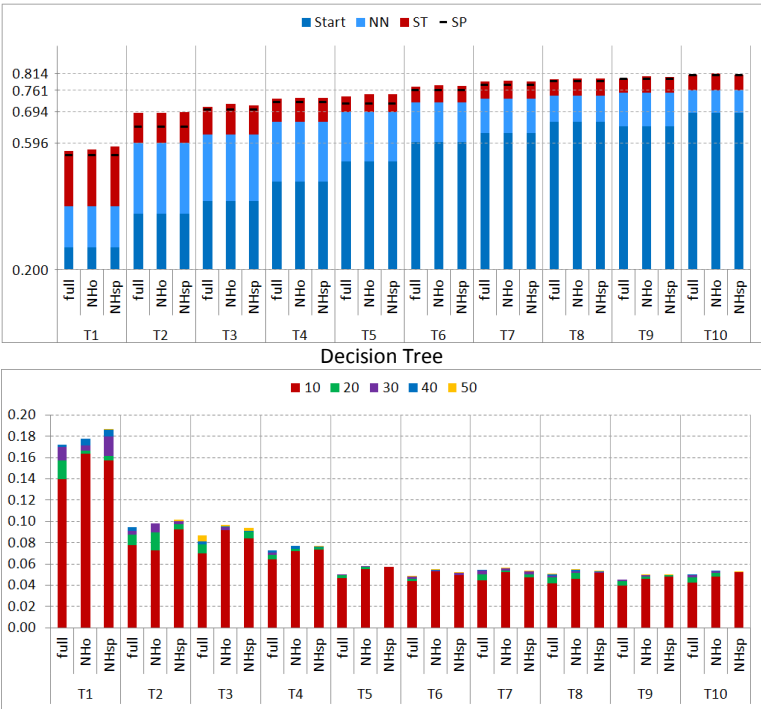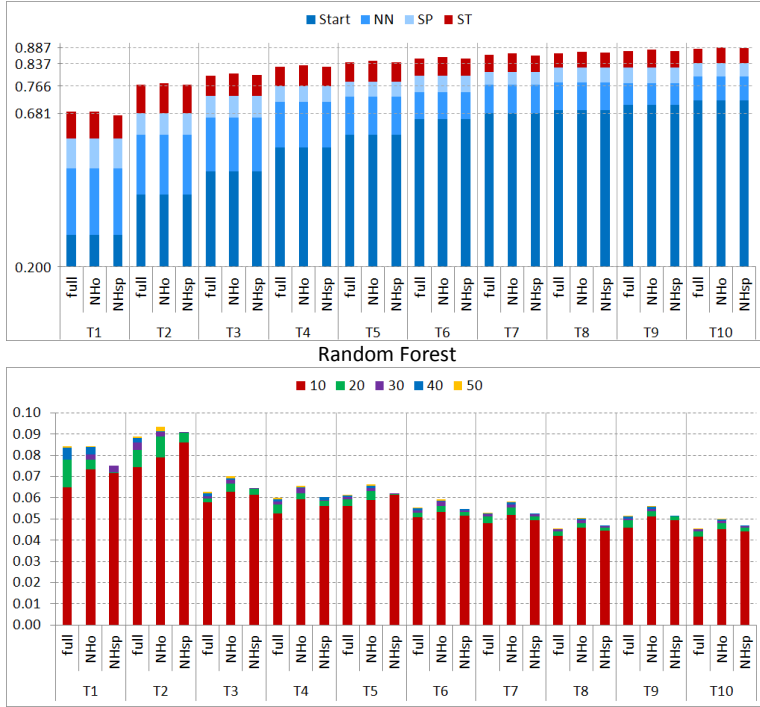

Figure 95 – Top row: Classification accuracies for different training sets, SP_i using the three selection methods for the unlabeled samples. Bottom row: Improvements of self-training (ST) per 10th iteration for the Random Forest base learner.

Along with the numerical evaluations, visual classification results are presented from the initial to the final classification output. Figure 98 illustrates the sample classification maps using RF as the base classifier within the bagging ensemble approach. In the figure, visual comparison can be made regarding the effects of search neighborhoods, *full* and NH_o, over the classification performance and the top row shows the initial results over an instance of T_2. The images displaying the difference to ground truth show the major misclassification of a particular class while white indicating correct labels. Thus, larger white areas represent a better match with the ground truth. The green circles for the classification results annotated with the ST iteration number 10 in row two indicate the classification difference between the two search neighborhoods, *full* and NH_o, compared to the initial results from the labeled samples in the first row. In particular, this row shows the difference between the classification performances achieved using two neighborhood approaches, NH_o and *full* both visually and numerically. For the following rows, the green and red circles indicate higher improvements or degradations, respectively, compared to the corresponding previous row. Overall, it is visible from the figure that major improvements after 10 iterations are achieved by applying NH_o and during the rest of the 20 iterations, only minor improvements are observed. As observed in the final numerical classification results, the application of the neighborhood approach, *full,*

yields an improvement for the major areas after 20 iterations, yet does not achieve better results.
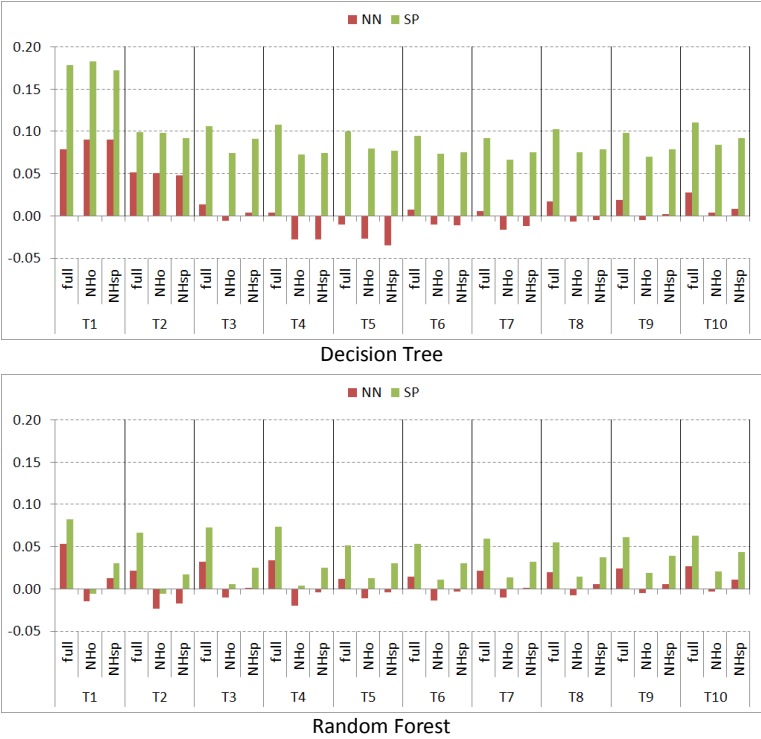
Figure 96 – The plots of classification accuracy differences of final self-training results comparing NN_i and SP_i to T_i for the two base learners.

For visual and numerical comparisons of the different SSL approaches, Figure 99 shows results for the initial T_i, NN_i, SP_i, T_i in self-training, and initial T_{10} training sets. The first four are based on an instance of T_2 while T_{10} is chosen based on the numerical results for RF. It is worth noting that the visual classification results achieved by RF with SL over the set T_{10} and with SSL by self-training over T_2 applying NH_o are quite similar. The green circles mark the best classification performance in a particular area among all classification results. The comparison shows that the classification over T_2 with the application of SSL employing NH_o produces the best classification map. This is a significant accomplishment achieved by SSL along with a classifier initially trained with a small-sized training set particularly when compared to the classifier trained over the set T_{10} and thus having five times more user-labeled samples to form the training set. In this example, the visual results favor NN_2 over SP_2; however, this is vice versa when numerical results are compared. This is related to the particular T_2 instance and the corresponding superpixels. This shows that the starting point can be particularly critical for SSL especially when small sample sized training sets are used for the initial training.

Finally, a brief comparison is made against various supervised classifiers as evaluated in the previous Section 5.1 over different training set sizes 52, 104, 208, and 1041 samples,

which correspond to 0.25‰, 0.5‰, 1%, and 5% of the 208 000 pixel ground truth, respectively. All classifier parameters were optimized for the best classification performance (see Section 5.1.1). Their classification accuracies are averaged over 100 runs using $H\alpha A$ features and shown in Table 33. As an additional comparison, the classification results using the covariance matrix $\langle [C] \rangle$ are added to Table 33.

Table 33 – Results using the covariance matrix $\langle [C] \rangle$ and $H\alpha A$ features with supervised classifiers.

	DT		ELM		MLP		KNN	
	$\langle [C] \rangle$	$H\alpha A$	$\langle [C] \rangle$	$H\alpha A$	$\langle [C] \rangle$	$H\alpha A$	$\langle [C] \rangle$	$H\alpha A$
52	0.42	0.55	0.31	0.62	0.23	0.57	0.48	0.66
104	0.53	0.66	0.53	0.71	0.28	0.67	0.54	0.73
208	0.60	0.73	0.63	0.78	0.35	0.75	0.59	0.77
1041	0.68	0.82	0.75	0.86	0.50	0.88	0.69	0.86

	RF		SVM (linear)		SVM (polynomial)		SVM (rbf)	
52	0.63	0.78	0.54	0.72	0.53	0.71	0.54	0.72
104	0.70	0.83	0.63	0.79	0.61	0.79	0.62	0.80
208	0.75	0.86	0.70	0.84	0.68	0.83	0.67	0.85
1041	0.81	0.90	0.80	0.90	0.75	0.90	0.75	0.91
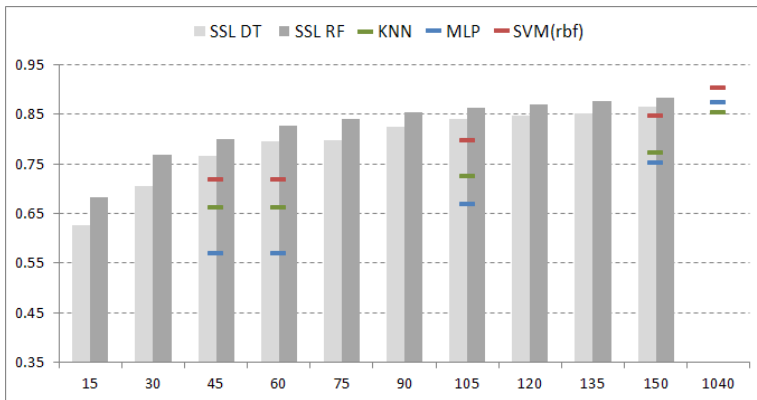
Figure 97 – Classification accuracy plots for comparison of SL versus the proposed SSL self-training approach using traditional SL classifiers.

Our previous experiments and evaluations have shown that SSL and ST using small-labeled data are able to achieve similar classification performances compared to supervised learning with larger labeled training sets using the same underlying classifier. The same observation can be made for typical classifiers such as KNN, MLP, and SVMs as illustrated in Figure 97. The most interesting fact is that training sets with six or more samples per class using superpixels to enlarge the initial labeled training data combined with the ensemble-based self-training is able to achieve comparable classification performances with various SL methods using as high as 1000 labeled samples.

Classified image with labeled samples

Overlaid with ground truth regions

difference to ground truth

53.70%

NH_o neighborhood

Overlaid with ground truth regions

difference to ground truth

74.43%

78.35%

full neighborhood

Overlaid with ground truth regions

difference to ground truth

68.33%

72.48%

t=10

t=20

t=30   80.26%

t=40   80.40%

    74.05%

    74.83%

    79.97%

t=50

    74.66%

Classified images after 50 iteration

Figure 98 – A sample set of classification maps obtained by different self-training iterations over an instance of labeled set T_2 using Random Forest as the ensemble base classifier. White color indicates a match between classification results and the ground truth. Green circles for results in the 10th iteration of the self-training (t=10) indicate difference among the two search neighborhoods, $full$ and NH_o, compared to the initial results from the labeled samples in the first row. For the following rows, the green and red circles indicate higher improvements or degradation, respectively, to the corresponding previous row. Percentages are the respective classification accuracies obtained per iteration.

difference to ground truth

overlaid ground truth regions

entire classified image

T_2 (53.70%)

NN_2 (65.66%)

SP_2 (60.65%)
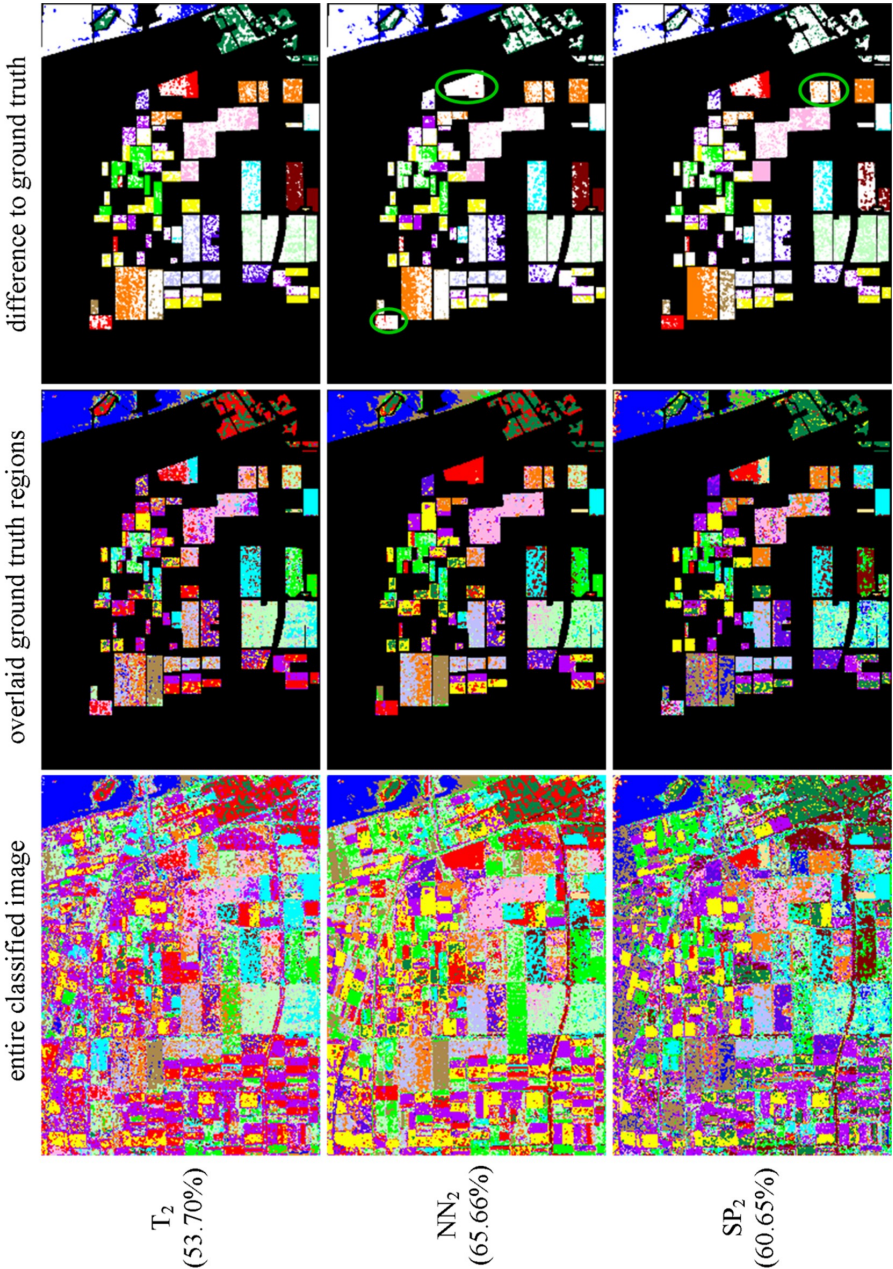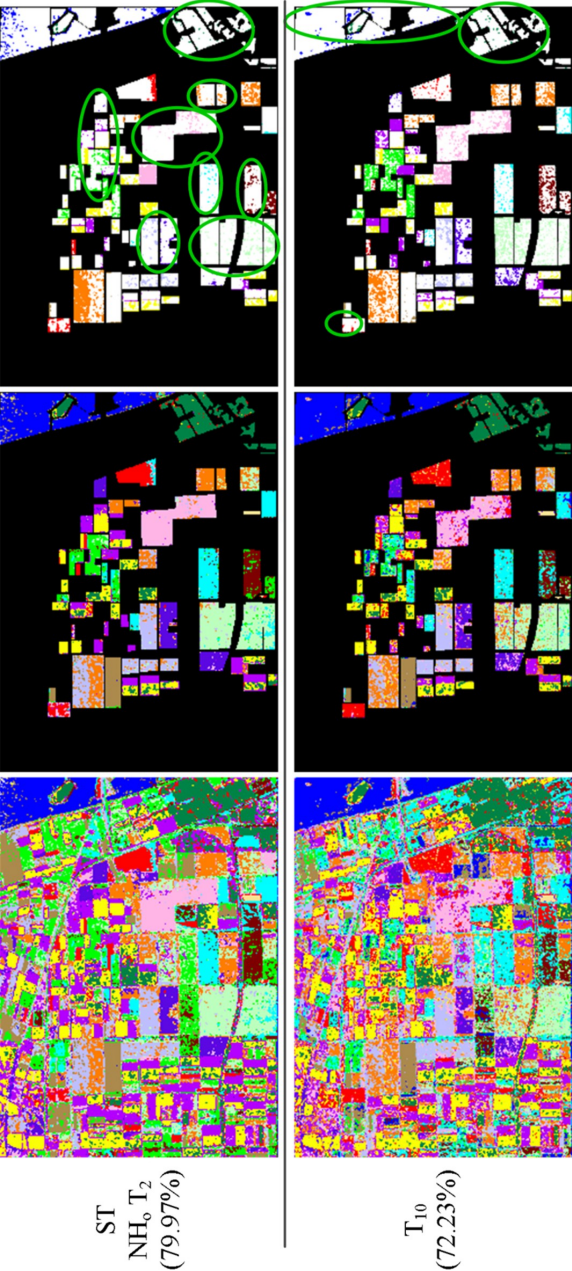
Figure 99 – Differences of distinct SSL approaches using unsupervised + supervised methods (NN, SP) and bagging ensemble self-training (using Random Forest) over a T_2 instance. White color indicates a match between classification results and ground truth. The green circles mark the best classification performance achieved in a particular area over all classification maps. Percentages are the respective classification accuracies.

5.3. SUMMARY

In this chapter, different approaches of semi-supervised learning over polarimetric SAR data were investigated with the focus drawn on the small sample size problem. Unsupervised methods such as contextual information (i.e., connected neighbors) or clustering/segmentation approaches (i.e., superpixels) to enlarge the initial labeled training set have shown promising results to address the small sample size problem in the right direction. Additionally, the employed self-training using an ensemble-based approach has proven beneficial especially in cases when it can achieve similar classification performances over small training sets compared to the classification results of the same classifiers trained over significantly larger training sets in a supervised manner.

Furthermore, it has been principally shown that different strategies on how to select reliable candidates from a large set of unlabeled samples can speed-up and improve the classification performance. In particular, for a remote sensing application such as polarimetric SAR image classification, it is advantageous to exploit the location-based information from the labeled training data. The choice of the applied confidence threshold can be critical particularly for weaker classifiers, where an adaptive approach can be applied starting with larger values and slowly decreasing it over time. However, we could observe that this approach alone cannot guarantee to achieve a classification performance that is beyond a certain level since the initial smaller training set sizes are critical. Nevertheless, in accordance with the number of base classifiers in the ensemble approach, it will still help to decrease the number of semi-supervised learning iterations by achieving similar or even better results when compared to common supervised learning approaches over different training set sizes.

Chapter 6

Conclusions

Analysis and interpretation of remote sensing data has been and will continue to be a major challenge within Earth Observation. Yet these observation activities do require an efficient remote sensing process starting with data acquisition, data storage, and data processing. With such data management, analysis and interpretation applications of observing and monitoring the earth's environment require accurate information extraction and efficient computational intelligence procedures. However, there is no global approach to address information extraction and machine learning applications partly due to various types and sources of remote sensing data each having own challenges in the process.

One remote sensing technique to aid Earth Observation is radar-based imaging gaining major interests due to advances in its imaging techniques. These are particularly *synthetic aperture radar* (SAR) and *polarimetry* with SAR providing the capability of high-resolution radar imaging and polarimetry introducing unique characteristics of the obtained scattering information. This resulted in an increase of such radar imaging platforms providing significantly more data. Additionally, as the underlying radar technology is capable of acquiring images in all-weather conditions as well as during day and night, it provides an increasing interest in various applications within the areas of geology, agriculture, forestry, urban, oceanography, military, and disaster management.

The majority of these applications are related to observation, detection, and classification to support humans within their efforts of monitoring and decision-making with respect to Earth's environment. In the process of analysis and interpretation, they commonly apply computational- and machine-learning techniques. With the increasing amount of data available, this does become also a large-scale learning and classification challenge due to the constantly ongoing process of acquiring new data. This thesis addressed this challenge by improving the classification process with special focus on land use and land cover applications using polarimetric SAR data. To achieve this, main contributions are made in superior feature extraction and advanced machine-learning techniques in form of classifier topologies, principles, and data exploitation.

For SAR image classification, the integration of texture features from the image-processing field was first considered decades ago and provided extra discrimination among classes. Since

then, texture features played a significant role in the analysis and interpretation stages of the SAR classification process.

Therefore, the introduction of new visual descriptors (i.e., color features) was investigated for polarimetric SAR land use and land cover classification. The evaluations demonstrated that color features provided additional discrimination among terrain classes on top of the conventional scattering information and texture features. For partially PolSAR image data the generation of pseudo color images to extract color features is different because the full polarimetric information is not available to apply mapping of the polarization matrices to the individual RGB components. For dual-pol images, combinations of the two polarization matrices assigned to the RGB color components provide a similar straightforward option to obtain such pseudo color images. However, this is obviously not possible considering single-pol or one intensity channel images. Therefore, this thesis considered the mapping of intensity values based on a color scale or transforming the intensity values into a color space to remedy this drawback.

Experiments over fully and partially polarimetric SAR images demonstrated that with the integration of the color features, significant improvements can be achieved over the classification performances either with the common PolSAR features alone or also the combination of PolSAR and texture features. With the color features used during all classification experiments and regardless from the PolSAR data, it was observed that additional texture features contributed to the classification performance. Yet in comparison, the performance gains and losses caused by the texture features vary according to the terrain types and the SAR data over which they were extracted.

We have particularly observed that the application of the considered color features can provide noteworthy improvements in man-made, foliage, and tree-type dominated vegetation terrain classes for fully PolSAR images as well as agricultural, forestry, and vegetation terrains for the partially PolSAR images. In addition to this, experiments validated that individual color features alone are able to provide a high discrimination mainly considering pseudo color images generated using the fully available polarization matrices. Furthermore, it could be observed that the color features work over intensity images yet obviously not as effectively as over color images. However, they were still able to provide comparable results in relation to results with the applied texture features, which might be related to specific terrain classes in certain classification setups. Moreover, similar or better results may be achieved regarding the dual-pol image by extracting texture and/or color features from a single pseudo color image reducing the overall feature dimension rather than processing the two available polarization matrices individually. As for the single-pol image, it is not anticipated that the considered color features extracted over a color transformed intensity image will significantly outperform traditional texture features yet we could observe that such color features were able to provide additional discrimination for classification of such data.

To address the large-scale learning challenge with the ever-growing amount of PolSAR data, a scalable and data-driven supervised classification framework, called Collective Network of Binary Classifiers (CNBC) was adapted with the primary objectives of maximizing learn-

ing efficiency and classification accuracy. This framework incorporates the active learning principle to support human users regarding the analysis and interpretation of polarimetric SAR data with particular focus on collections of images. When processing such image collections in a classification scenario, changes or updates to the existing classifier might be required frequently due to surface, terrain, and object changes as well as variations in capturing time and position.

The CNBC framework mainly adopts a "divide and conquer" type of approach, so as to efficiently handle indefinite number of PolSAR features and terrain classes, which may otherwise be a difficult, if not infeasible, problem for a single classifier due to the well-known "curse of dimensionality" phenomenon. Within the CNBC approach, compact classifiers, which can be evolved in a much more efficient way than a single but complex classifier, are conveniently considered in a predefined architecture (configuration) space, in which the optimal classifier for the current classification problem can be sought using evolutionary techniques. At any given time, this allows creating a dedicated binary classifier (BC) for discriminating a certain terrain type from the others with the use of a single feature. Each incremental evolution session *learns* from the current best classifier and can improve it further, possibly with another configuration in the architecture space. Moreover, with each incremental evolution, new classes and features (such as previously proposed color features) can also be introduced which signals CNBC to adapt dynamically to such changes.

The capabilities of CNBC have been demonstrated over an extensive set of experimental results regarding the adaptation and data-driven classification of single as well as collections of PolSAR images. This revealed that with the ongoing incremental evolutions, a generic CNBC can be created that adapted itself according to the PolSAR image and classes presented to it over time. Compared to static classifiers such as ANN, building up on and extending an existing classifier is beneficial in terms of classification accuracy and time without discarding the existing knowledge and recreating a new classifier from scratch. Additionally, while static and regular classifiers may suffer from the "curse of dimensionality" phenomenon without an accompanying feature selection or dimension reduction technique, CNBC addresses this well with its "divide and conquer" strategy. The experimental results verified that the CNBC provided an efficient solution for the problems of scalability and dynamic adaptability allowing both feature space dimensions and the number of terrain classes in PolSAR image collections to vary dynamically. Whenever the CNBC is evolved in batch mode, it can compete and even surpass other static classifiers especially when the feature space dimension increases. This is an expected outcome since the CNBC framework can take advantage of any visual feature as long as it has the discrimination power for only one or few classes.

As most classification problems are addressed by supervised machine learning, they generally outperform their unsupervised counterparts. However, the performance of any supervised learning is limited to the availability of labeled data. To benefit from the unlabeled data, supervised and unsupervised learning principles are combined into semi-supervised learning approaches. Hence, the application of semi-supervised learning is motivated by ill-posed classification tasks particularly the small training size problem. To address this, two semi-

supervised approaches were proposed based on unsupervised extension of the training data and ensemble-based self-training. The considered unsupervised methods such as contextual information (i.e., connected neighbors) or clustering/segmentation approaches (i.e., superpixels) to enlarge the initial training data have shown promising results to address the small sample size problem. Additionally, the employed self-training using an ensemble-based approach has proven beneficial especially in cases when it is able to achieve similar classification performances over small training sets compared to the results achieved with supervised classifiers trained over significantly larger training sets.

Furthermore, it has principally been shown that different strategies on how to select reliable candidates from a large set of unlabeled samples can speed-up and improve the classification performance. In particular, for a remote sensing application such as polarimetric SAR image classification, it is advantageous to exploit the location-based information from the labeled training data. However, we have also observed that this approach alone cannot guarantee to achieve a classification performance that is beyond a certain level since the initial smaller training set sizes are critical. Nevertheless, in accordance with the number of base classifiers in the ensemble approach, it will still help to decrease the number of semi-supervised learning iterations by attaining similar or even better results when compared to common supervised learning approaches over different training set sizes. We can foresee that there is an imminent need to investigate further different strategies to select unlabeled samples reliably including measures for sample diversity as well as to consider semi-supervised learning within the application of domain adaptation and transfer learning.

Demand for high-level classification increases to achieve fine-grain categorization of the earth terrain and surface for fine-detail land use and land cover (LULC) differentiation such as various types of crop, forest, water, and urban areas. The advances made in polarimetric SAR are important steps towards this direction, particularly very-high-resolution imaging radars achieving resolutions closer to the optical counterparts. Lower resolution imagery will lack the necessary detail needed to discriminate the fine-grain categorization desired. Moreover, polarimetric SAR alone might not be enough to achieve such high-level LULC differentiation, as for similar surface types the backscattering differences will significantly decrease and may not even be distinguishable. Therefore, combination and fusion with other SAR data such as interferometry or even from other remote sensing sources such as optical, spectral, or lidar will be highly beneficial.

By applying very-high-resolution data to accomplish fine-grain classification, the main question will be if the traditional SAR features in form of backscattering coefficients and target decompositions as well as texture and color are still applicable. Generally, to achieve this, features are need that are able to describe and discriminate characteristics among classes. With an increasing amount of classes with fine-detail differences, the traditional hand-made features might not be sufficient. Hence, the automatic extraction and unsupervised learning of features will become relevant in the future. The possible applications of Feature- and Deep Learning are to be explored, which are currently considered as the state-of-the-art in computer vision and image classification such as coarse- and fine-grain object categorization. This in-

cludes further investigation if color features are suitable for fine grain classification for different crop, forest, water, and urban types.

The concept of fusing different features has been demonstrated by CNBC. Furthermore, the CNBC topology is generally capable of handling large number of classes, yet some adaptation to fine-grain categorization is probably required. This could be achieve in a hierarchical process by first differentiate LULC classes as best as possible among all and in a second stage find tune among the most confusing ones. Adding new classes and features is expected to work, as long as there is at least one feature good enough to distinguish new class from the others.

Other potential extensions and future works related to the proposed techniques within this thesis relate to feature transformation and domain adaptation for the classification of image collections by further improving generalization capabilities and classification performances. Moreover, other forms of learning principles such as online learning can be considered within the application of CNBC. In addition, color features and the classification topology presented in Chapter 4 and 5, respectively, can be applied and investigated with other remote sensing data such as multi- and hyperspectral data.

The techniques proposed and investigated within this thesis have shown valuable contributions to the field of land use and land cover polarimetric SAR image classification. There is still a lot of potential for further research to reduce the amount of manual labor regarding improvements with respect to analysis and interpretation of polarimetric SAR and other remote sensing data. Such contributions are essential in support of Earth Observation and to enhance the understanding and management of the earth itself and its environment.

Bibliography

[1]   R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC Superpixels Compared to State-of-the-art Superpixel Methods", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, November 2012.

[2]   T.L. Ainsworth, J.P. Kelly, and J. Lee, "Comparisons between dual-pol , compact polarimetric and quad-pol SAR imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 64, no. 5, pp. 464–471, September 2009.

[3]   N. Alajlan, Y. Bazi, H. Al Hichri, and E. Othman, "Robust classification of hyperspectral images based on the combination of supervised and unsupervised learning paradigms" *IEEE International Geoscience and Remote Sensing Symposium*, Munich, Germany, 22-27 July 2012.

[4]   V. Alberga, G. Satalino, and D.K. Staykova, "Comparison of polarimetric SAR observables in terms of classification performance," *International Journal of Remote Sensing*, vol. 29, no. 14, pp. 4129–4150, July 2008.

[5]   W. An, Y. Cui, and J. Yang, "Three-component model-based decomposition for polarimetric SAR data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 6, pp. 2732–2739, June 2010.

[6]   F. Argenti, A. Lapini, L. Alparone, and V. S. Marta, "A Tutorial on Speckle Reduction in Synthetic Aperture Radar Images", *IEEE Geoscience and Remote Sensing Magazine*, vol. 1, no. 3, pp. 6–35, September 2013.

[7]   M. Arii, J.J. van Zyl, and Y.J. Kim, "Adaptive model-based decomposition of polarimetric SAR covariance matrices," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 3, pp. 1104–1113, March 2011.

[8]   M. Arii, J.J. van Zyl, and Y. Kim, "A General Characterization for Polarimetric Scattering From Vegetation Canopies," *IEEE Transactions on Geoscience and Remote* Sensing, vol. 48, no. 9, pp. 3349–3357, September 2010.

[9]   T. Arvidson, J. Gasch, and S.N. Goward, "Landsat 7's long-term acquisition plan – an innovative approach to building a global imagery archive", *Remote Sensing of Environment*, vol. 78, no. 1–2, pp. 13–26, October 2001.

[10]   E. Attema, M. Davidson, P. Snoeij, B. Rommen, and N. Floury, "Sentinel-1 mission overview," *IEEE International Geoscience and Remote Sensing Symposium,* Capetown, Soth Africa, 12-17 July 2009.

[11]   Ö. Aytekin, M. Koç, and I. Ulusoy, "Local Primitive Pattern for the Classification of SAR Images", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 4, pp. 2431–2441, April 2013.

[12]   S.K. Babey and C.D. Anger, "A Compact Airborne Spectrographic Imager", *IEEE International Geoscience and Remote Sensing Symposium, Vancouver, Canada*, 10-14 July 1989.

[13] Y. Ban and H. Hu, "RADARSAT Fine-Beam SAR Data for Land-Cover Mapping and Change Detection in the Rural-Urban Fringe of the Greater Toronto Area", *Urban Remote Sensing Joint Event*, Paris, France, 11-13 April 2007.

[14] A. Baudoin, E. Boussarie, P. Damilano, G. Rum, and F. Caltagirone, "Pléiades: a Multi Mission and Multi Cooperative Program," *52nd International Astronautical Congress,* Toulouse, France, October 1-5, 2001.

[15] K.P. Bennett, A. Demiriz, and R. Maclin, "Exploiting unlabeled data in ensemble methods", *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Edmonton, Alberta, Canada, July 23-26 2002.

[16] J.C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Kluwer Academic Publishers, 1981.

[17] C.M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1996.

[18] G. Brassard and P. Bratley, *Fundamental of Algorithmics*, Prentice-Hall, 1996.

[19] L. Breiman, "Random Forests", *Machine Learning*, vol. 45, no. 1, pp. 5–32, October 2001.

[20] G. Brown, A.Pocock, M.Lujan, and M.-J.Zhao, "Conditional Likelihood Maximisation: A Unifying Framework for Mutual Information Feature Selection", *Journal of Machine Learning Research*, vol. 13, pp. 27-66, January 2012.

[21] L. Bruzzone and C. Persello, "A novel context-sensitive semisupervised SVM classifier robust to mislabeled training samples", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 7, pp. 2142–2154, July 2009.

[22] L. Bruzzone, M. Chi, and M. Marconcini, "A novel transductive SVM for semisupervised classification of remote-sensing images", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no 11, pp. 3363–3373, 2006.

[23] L. Bruzzone, M. Marconcini, U. Wegmuller, and A. Wiesmann, "An advanced system for the automatic classification of multitemporal SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 6, pp. 1321–1334, June 2004.

[24] L. Bruzzone, C. Persello, "Recent trends in classification of remote sensing data: active and semisupervised machine learning paradigms", *IEEE International Geoscience and Remote Sensing Symposium*, Honolulu, Hawaii, USA, 25-30 July 2010.

[25] G. Camps-Valls, T. Marsheva, and D. Zhou, "Semi-supervised graph-based hyperspectral image classification", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 10, pp. 3044–3054, October 2007

[26] F. Cao, C. Deledalle, J. Nicolas, F. Tupin, L. Ferro-famil, E. Pottier, L. Carlos, O. De Lyon, and C. Cral, "Influence of speckle filtering of Polarimetric SAR data on different classification methods", *IEEE International Geoscience and Remote Sensing Symposium*, Vancouver, BC, Canada, 24-29 July 2011.

[27] C.C. Chang and C.J. Lin, "LIBSVM: a library for support vector machines", *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, 27 pages, May 2011.

[28] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning*. MIT Press. Cambridge, MA, 2006.

[29] C. Chen and K. Chen, "The use of fully polarimetric information for the fuzzy neural classification of SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 9, pp. 2089–2100, September 2003.

[30] K. Chen, W. Huang, D. Tsay, and F. Amar, "Classification of multifrequency polarimetric SAR imagery using a dynamic learning neural network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 34, no. 3, pp. 814–820, May 1996.

[31] L. Chen, W. Yang, Y. Liu, and H. Sun, "Feature evaluation and selection for polarimetric SAR image classification", *IEEE International Conference On Signal Processing*, Beijing, China, 24-28 October 2010.

[32] S. Cheng, Q. Huang, J. Liu, and X. Tang "A novel inductive semi-supervised SVM with graph-based self-training", *Intelligent Science and Intelligent Data Engineering: Lecture Notes in Computer Science*, vol. 7751, pp. 82–89, 2013.

[33] M. Chi and L. Bruzzone, "A semilabeled-sample-driven bagging technique for ill-posed classification problems", *IEEE Geoscience and Remote Sensing Letters*, vol. 2, no. 1, pp. 69–73, January 2005.

[34] M. Chi and L. Bruzzone, "An ensemble-driven k-NN approach to ill-posed classification problems", *Pattern Recognition Letters*, vol. 27, no. 4, pp. 301–307, April 2006.

[35] M. Chi, R. Feng, and L. Bruzzone, "Classification of hyperspectral remote-sensing data with primal SVM for small-sized training dataset problem", *Advances in Space Research*, vol. 41, no. 11, pp. 1793–1799, November 2008.

[36] M. Chi, Q. Qian, and J.A. Benediktsson, "Cluster-Based Ensemble Classification for Hyperspectral Remote Sensing Images", *IEEE International Geoscience and Remote Sensing Symposium*, Boston, Massachusetts, USA, 6-11 July 2008.

[37] E. Christensen and J. Dall, "EMISAR: A dual-frequency, polarimetric airborne SAR", *IEEE International Geoscience and Remote Sensing Symposium*, Toronto, Canada, June 24 – 28, 2002.

[38] A. Chukhlantsev, *Microwave Radiometry of Vegetation Canopies*, Springer, 2006.

[39] S.R. Cloude and E. Pottier, "A review of target decomposition theorems in radar polarimetry," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 34, no. 2, pp. 498–518, March 1996.

[40] R.G. Congalton and K. Green, *Assessing the Accuracy of Remotely Sensed Data: Principles and Practices*, CRC Press (Taylor and Francis Group), 2008.

[41] R.G. Congalton, R.G. Oderwald, and R.A. Mead, "Assessing Landsat classification accuracy using discrete multivariate analysis statistical techniques", *Photogrammetric Engineering and Remote Sensing*, vol. 49, no. 12. pp. 1671–1678, December 1983.

[42] X. Corine Land Cover 2006, Online: http://sia.eionet.europa.eu/CLC2006/

[43] C. Cortes and V. Vapnik, "Support-Vector Networks", *Machine Learning*, vol. 20, no. 3, pp. 293–297, September 1995.

[44] M.M. Crawford, D. Tuia, and H.L. Yang, "Active Learning: Any Value for Classification of Remotely Sensed Data?", *Proceedings of the IEEE*, vol. 101, no. 3, pp. 593–608, March 2013.

[45] M. Dabboor and V. Karathanassi, "A knowledge-based classification method for polarimetric SAR data," *Proceedings of SPIE*, vol. 5980, 59800C-13, 2005.

[46] D. Dai, W. Yang, and H. Sun, "Multilevel Local Pattern Histogram for SAR Image Classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 2, pp. 225–229, March 2011.

[47] M.R. de Leeuw and L.M. Tavares de Carvalho, "Performance evaluation of several adaptive speckle filters for SAR imaging," *Anais XIV Simpósio Brasileiro de Sensoriamento Remoto*, Natal, Brasil, 25-30 April 2009.

[48] R. Dekker, "Texture analysis and classification of ERS SAR images for map updating of urban areas in the Netherlands", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 9, pp. 1950–1958, September 2003.

[49] F. Del Frate, G. Schiavon, D. Solimini, M. Borgeaud, D.H. Hoekman, and M.A.M. Vissers, "Crop classification using multiconfiguration C-band SAR data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 7, pp. 1611–1619, July 2003.

[50] F. Del. Frate, F. Pacifici, and D. Solimini, "Urban Land Cover in Rome, Italy, monitored by single-parameter multi-temporal SAR images", *Urban Remote Sensing Joint Event*, Paris, France, 11-13 April 2007.

[51] F. Dell'Acqua and P. Gamba "Texture-based characterization of urban environments on satellite SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 1, pp. 153–159, January 2003.

[52] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm", *Journal of the Royal Statistical Society B*, no. 39, vol. 1, pp. 1–38, 1977.

[53] Q. Deng, Y. Chen, W. Zhang, and J. Yang, "Colorization for Polarimetric SAR Image Based on Scattering Mechanisms", *Congress on Image and Signal Processing*, Sanya, China, 27-30 May 2008.

[54] H. De-Yong and L. Xiao-Juan, "Texture Analysis and its Application for Single-Band SAR Thematic Information Extraction", *IEEE International Geoscience and Remote Sensing Symposium,* Boston, MA, 6-11 July 2008

[55] DigitalGlobe Inc., [Online], www.digitalglobe.com

[56] M.C. Dobson, F. Ulaby, and L.E. Pierce, "Land-cover classification and estimation of terrain attributes using synthetic aperture radar", *Remote Sensing of Environment*, vol. 51, no. 1, pp. 199-214, January 1995.

[57] L.J. Du and J.S. Lee, "Polarimetric SAR image classification based on target decomposition theorem and complex Wishart distribution," *IEEE International Geoscience and Remote Sensing Symposium*, Lincoln, Nebraska, USA 27 – 31 May 1996.

[58] P. Du, J. Xia, W. Zhang, K. Tan, Y. Liu, and S. Liu, "Multiple Classifier System for Remote Sensing Image Classification: A Review", *Sensors*, vol. 12, no. 4, pp. 4764–4792, April 2012.

[59] P. Dubois-Fernandez, O. Ruault du Plessis, D. Le Coz, J. Dupas, B. Vaizan, X. Dupuis, H. Cantalloube, C. Coulombeix, C. Titin-Schnaider, P. Dreuillet, J. Boutry, J. Canny, L. Kaisersmertz, J. Peyret, P. Martineau, M. Chanteclerc, L. Pastore, and J. Bruyant, "The ONERA RAMSES SAR System", *IEEE International Geoscience and Remote Sensing Symposium*, Toronto, Canada, June 24- 28, 2002.

[60] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification*, Wiley, 2000.

[61] M. Duquenoy, J. Ovarlez, L. Ferro-Famil, and E. Pottier, "Supervised Classification Of Scatters On SAR Imaging Based On Incoherent Polarimetric Time-Frequency Signatures," *European Signal Processing Conference*, Glasgow, Scotland, 24-28 Agust 2009.

[62] A. Erkan, G. Camps-Valls, and Y. Altun, "Semi-supervised remote sensing image classification via maximum entropy", *IEEE International Workshop on Machine Learning for Signal Processing*. Kittilä, Finland, 29.Aug – 1.Sep. 2010.

[63] K. Ersahin, I.G. Cumming, and R.K. Ward, "Segmentation and classification of polarimetric SAR data using spectral graph partitioning", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 1, pp. 164–174, January 2010.

[64] K. Ersahin, B. Scheuchl, and I. Cumming, "Incorporating Texture Information into Polarimetric Radar Classification Using Neural Networks," *IEEE International Geoscience and Remote Sensing Symposium*, Anchorage, Alaska, USA, 20-24 September 2004.

[65] G.M. Foody, "Status of land cover classification accuracy assessment". *Remote Sensing of Environment*, vol. 80, no. 1, pp. 185–201, April 2002.

[66] A. Freeman and S.L. Durden, "A three-component scattering model for polarimetric SAR data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 36, no. 3, pp. 963–973, May 1998.

[67] A. Freeman, "SAR Calibration: An Overview", *IEEE Transactions on Geoscience and Remote Sens,* vol. 30, no. 6, pp. 1107–1121, November 1992.

[68] R. French, "Catastrophic interference in connectionist networks", *Trends in Cognitive Sciences,* vol. 3, no. 4, pp. 128–135, April 1999.

[69] L. Fry, G. Xian, S. Jin, J. Dewitz, C. Homer, L. Yang, C. Barnes, N. Herold, and J. Wickham, "Completion of the 2006 National Land Cover Database for the Conterminous United States", *Photogrammetric Engineering and Remote Sensing*, vol. 77, no. 9, pp. 858–864, September 2011.

[70] L. Gao and Y. Ban, "Investigating the performance of SAR polarimetric features in land-cover classification," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences.* vol. XXXVII, 2008.

[71] T. Gevers, A. Gijsenij, J. Weijer and J. Geusebroek, *Color in Computer Vision: Fundamentals and Applications*, Ed. New York: Wiley, 2012.

[72] G. Gigli, R. Sabry, and G. Lampropoulos, "Classifier combination and feature selection methods for polarimetric SAR classification," *Proceedings of SPIE*, vol. 6571, 2007.

[73] R.C. Gonzalez, R.E. Woods, *Digital Image Processing*, Prentice Hall: 3rd Edition, 2008.

[74] L.C. Graham, "Synthetic interferometer radar for topographic mapping" *Proceedings of the IEEE*, vol. 62, pp. 763–768, June 1974

[75] R.O. Green, M.L. Eastwood, L. Michael, C.M. Sarture, T.G. Chrien, M.Aronsson, B.J. Chippendale, J.A. Faust, and B.E. Pavri, "Imaging Spectroscopy and the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS)", *Remote Sensing of Environment*, vol. 65, no 3, pp. 227–248, September 1998.

[76] D. Greene, P. Cunningham, and R. Mayer, "Unsupervised Learning and Clustering", *Machine Learning Techniques for Multimedia in Cognitive Technologies*, pp 51–90, 2008.

[77] Y. Gu and K. Feng, "L1-graph semisupervised learning for hyperspectral image classification", *IEEE International Geoscience and Remote Sensing Symposium*, Munich, Germany, 22-27 July 2012.

[78] S. Guan, C. Bao, and R. Sun, "Hierarchical Incremental Class Learning with Reduced Pattern Training", *Neural Processing Letter,* vol. 24, no. 2, pp. 163–177, Octover 2006.

[79] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection", *Journal of Machine Learning Research*, vol. 3, pp. 1157-1182, March 2003.

[80] R. Hänsch and O. Hellwich, "Semi-supervised learning for classification of polarimetric SAR-data", *IEEE International Geoscience and Remote Sensing Symposium*, Cape Town, South Africa, 12-17 July 2009.

[81] R. Hänsch, "Complex-Valued Multi-Layer Perceptrons - An Application to Polarimetric SAR data*, Photogrammetric Engineering and Remote Sensing*, vol. 76, no. 9, pp. 1081–1088, September 2010.

[82] Y. Hara, R.G. Atkins, S.H. Yueh, R.T. Shin, and J. Kong, "Application of neural networks to radar image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 32, no. 1, pp. 100–109, January 1994.

[83] R. Haralick, K. Shanmugam, and I. Dinstein, "Textural Features for Image Classification", *IEEE Transactions* on *Systems, Man and Cybernetics,* vol. 3, no. 6, pp. 610–621, November 1973.

[84] W. He, M. Jaeger; A. Reigber, O. Hellwich, "Building Extraction from Polarimetric SAR Data using Mean Shift and Conditional Random Fields", *7th European Conference on Synthetic Aperture Radar*, Friedrichshafen, Germany, 2-5 June 2008.

[85] M. Hellmann, G. Jager, E. Kratzschmar, and M. Habermeyer, "Classification of full polarimetric SAR-data using artificial neural networks and fuzzy algorithms," *IEEE International Geoscience and Remote Sensing Symposium*, Hamburg, Germany, 28 June - 2 July1999.

[86] D.H. Hoekman and M.A.M. Vissers, "A new polarimetric classification approach evaluated for agricultural crops," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 41, no. 12, pp. 2881–2889, December 2003.

[87] C. Homer, J. Dewitz, J. Fry, M. Coan, N. Hossain, C. Larson, N. Herold, A. McKerrow, J. N. VanDriel, and J. Wickham. "Completion of the 2001 National Land Cover Database for the conterminous United States", *Photogrammetric Engineering and Remote Sensing*, vol. 73, no. 4, pp. 337–341, April 2007.

[88] I. Hønstvet, I. Encke, D. Hall, and Y. Munro, "AstroSAR-Lite – A Radar System for the Tropics", *XIII Simpósio Brasileiro de Sensoriamento Remoto*, Florianópolis, Brasil, 21-26 April 2007.

[89] R. Horn, "The DLR airborne SAR project E-SAR", *IEEE International Geoscience and Remote Sensing Symposium*, Lincoln, Nebraska, USA 27 – 31 May 1996.

[90] R. Horn, A. Nottensteiner, and R. Scheiber, "F-SAR – DLR's advanced airborne SAR system onboard DO228", *European Conference on Synthetic Aperture Radar*, Friedrichshafen, Germany, 2-5 June 2008.

[91] C.W. Hsu and C.J. Lin, "A simple decomposition method for support vector machines," *Machine Learning*, vol. 46, no. 1, pp. 291–314, January 2002.

[92] G.B. Huang, Q.Y. Zhu, and C.K. Siew, "Extreme learning machine: theory and applications", *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, December 2006.

[93] M. Hutchinson and M.D. Gibbons, "ENVISAT ASAR - Design and Performance with a View to the Future", RTO SET Symposium on Space-Based Observation Technology (RTO MP-61), Samos, Greece, 16-18 October 2000. http://www.dtic.mil/get-tr-doc/pdf?AD=ADP010832

[94] J.R. Huynen, "Phenomenological Theory of Radar Targets," Ph.D. Dissertation, Tech. Univ. Delft, Delft, The Netherlands, 1970.

[95] T. Ince, "Polarimetric SAR Image Classification Using Radial Basis Function Neural Network", *PIERS Online*, vol. 6, no. 5, pp. 470–475, 2010.

[96] Q. Jackson and D.A. Landgrebe, "An adaptive classifier design for high-dimensional data analysis with a limited training data set", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 12, pp. 2664–2679, December 2001,

[97] H. Jia, Y. Murphey, D. Gutchess and T. Chang, "Identifying knowledge domain and incremental new class learning in SVM", *IEEE International Joint Conference on Neural Networks*, Montreal, Canada, 31 July - 4 August 2005.

[98] R.L. Jordan, "The Seasat—A synthetic aperture radar system", IEEE Journal of Oceanic Engineering, vol. 5, no. 2, pp. 154–164, April 1980.

[99] G. Jun and J. Ghosh, "Spatially Adaptive Semi-Supervised Learning with Gaussian Processes for Hyperspectral Data Analysis", *Statistical Analysis and Data M*ining, vol. 4, no. 4, pp. 27–38, August 2011.

[100] G. Jun and J. Ghosh, "Semisupervised Learning of Hyperspectral Data with Unknown Land-Cover Classes", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 1, pp. 273–282, January 2013.

[101] U. Kandaswamy, D.A. Adjeroh, and M. Lee, "Efficient texture analysis of SAR imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 9, pp. 2075–2083, September 2005.

[102] L. Kaufman and P.J. Rousseeuw, *Clustering by means of Medoids in Statistical Data Analysis Based on the L_1–Norm and Related Methods*, edited by Y. Dodge, North-Holland, 1987.

[103] K. Kayabol, A. Voisin, and J. Zerubia, "SAR image classification with non-stationary Multinomial Logistic mixture of amplitude and texture densities", *IEEE International Conference on Image Processing*, Brussels, Belgium, 10-14 September 2011.

[104] K.U. Khan and J. Yang, "Polarimetric Synthetic Aperture Radar Image Classification by a Hybrid Method," *Tsinghua Science and Technology*, vol. 12, no. 1, pp. 97–104, February 2007.

[105] Y. Kim and J.J. van Zyl, "Comparison of forest estimation techniques using SAR data," *IEEE International Geoscience and Remote Sensing Symposium*, pp. 1395–1397, Sydney, Australia, 9-13 July 2001.

[106] S. Kiranyaz, M. Gabbouj, J. Pulkkinen, T. Ince, and K. Meissner, "Network of Evolutionary Binary Classifiers for Classification and Retrieval in Macroinvertebrate Databases", *IEEE International Conference on Image Processing*, Hongkong, China, 26-29 September 2010.

[107] S. Kiranyaz, T. Ince, A. Yildirim, and M. Gabbouj, "Evolutionary Artificial Neural Networks by Multi-Dimensional Particle Swarm Optimization", *Neural Networks*, vol. 22, no. 10, pp. 1448–1462, December 2009.

[108] S. Kiranyaz, T. Ince, S. Uhlmann, and M. Gabbouj, "Collective Network of Binary Classifier Framework for Polarimetric SAR Image Classification: An Evolutionary Approach", *IEEE Transactions on Systems, Man, and Cybernetics - Part B,* vol. 42, no. 4, pp. 1169–1186, August 2012.

[109] S. Kiranyaz, S. Uhlmann, T. Ince, and M. Gabbouj, "Perceptual Dominant Color Extraction by Multi-Dimensional Particle Swarm Optimization," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, Article 451638, 13 pages, 2009.

[110] A. Kourgli, M. Ouarzeddine, and Y. Oukil, "Land cover identification using polarimetric sar images," *ISPRS TC VII Symposium – 100 Years ISPRS*, vol. XXXVIII, 2010.

[111] U. Kreßel, "Pairwise classification and support vector machines," *Advances in Kernel Methods—Support Vector Learning*, B. Schölkopf, C.J. C. Burges, and A. J. Smola, Eds. Cambridge, MA: MIT Press, pp. 255–268, 1999.

[112] G. Krieger, I. Hajnsek, P. Papathanassiou, N. Younis, and A. Moriera, "Interferometric synthetic aperture radar (SAR) missions employing formation flying", *Proceedings of the IEEE*, vol. 98, no. 5, pp. 816–843, May 2010.

[113] E. Krogager, "A new decomposition of the radar target scattering matrix," *Electronics Letters,* vol. 26, no. 18, pp. 1525–1526, August 1990.

[114] V.A. Krylov, G. Moser, S.B. Serpico, and J. Zerubia, "Supervised High-Resolution Dual-Polarization SAR Image Classification by Finite Mixtures and Copulas", *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 3, pp. 554–566, March 2011.

[115] C. Lardeux and P. Frison, "Classification of tropical vegetation using multifrequency partial SAR polarimetry", *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 1, pp. 133–137, January 2011.

[116] J.S. Lee and E. Pottier, *Polarimetric Radar Imaging: From Basics to Applications*, Brian J. Thompson, CRC Press, 2009.

[117] J.S. Lee, M.R. Grunes, and G. De Grandi, "Polarimetric SAR speckle filtering and its implication for classification", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 5, pp. 2363–2373, September 1999.

[118] J.S. Lee, T.L. Ainsworth, and Y. Wang, "On polarimetric SAR speckle filtering", *IEEE International Geoscience and Remote Sensing Symposium*, Munich, Germany, 22-27 July 2012.

[119] J.S. Lee, M.R. Grunes, and E. Pottier, "Quantitative comparison of classification capability: fully polarimetric versus dual and single-polarization SAR", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 11, pp. 2343–2351, November 2001.

[120] A. Levinshtein, A. Stere, K.N. Kutulakos, D.J. Fleet, S.J. Dickinson, and K. Siddiqi, "TurboPixels: Fast Superpixels Using Geometric Flows", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2290–2297, December 2009.

[121] M.S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: State of the art and challenges", *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 2, no. 1, pp. 1–19, February 2006.

[122] M. Li and Z. Zhou, "SETRED: Self-training with editing", *Advances in Knowledge Discovery and Data Mining: Lecture Notes in Computer Science*, vol. 3518, pp. 611–621, 2005.

[123] M. Li and Z. Zhou, "Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples", *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, vol. 37, no. 6, pp. 1088–1098, June 2007.

[124] S. Liew, S. Kam, T. Tuong, P. Chen, V. Minh, and H. Lim, "Landcover classification over the Mekong river delta using ERS and RADARSAT SAR images", *IEEE International Geoscience and Remote Sensing Symposium*, Singapore, 3-8 August 1997.

[125] Y. Linde, A. Buzo, and R. Gray, "An Algorithm for Vector Quantizer Design", *IEEE Transactions on Communications*, vol. 28, no. 1, pp. 84–95, January 1980.

[126] X. Liu, Y. Li, W. Gao, and L. Xiao "Double Polarization SAR Image Classification based on Object-Oriented Technology", *Journal of Geographic Information System*, vol. 2, no. 2, pp. 113–119, April 2010.

[127] H. Liu, H. Motoda, R. Setiono, and Z. Zhao, "Feature Selection: An Ever Evolving Frontier in Data Mining", *Journal of Machine Learning Research - 4th Workshop on Feature Selection in Data Mining*, Hyderabad, India, 21 June 2010.

[128] A. Lonnqvist, Y. Rauste, M. Molinier, and T. Hame, "Polarimetric SAR data in land cover mapping in boreal zone," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 10, pp. 3652–3662, October 2010.

[129] Y. Lou, "Review of the NASA/JPL airborne synthetic aperture radar system, *IEEE International Geoscience and Remote Sensing Symposium*, Toronto, Canada, June 24-28, 2002.

[130] D. Lu and Q. Weng, "A survey of image classification methods and techniques for improving classification performance", *International Journal of Remote Sensing*, vol. 28, no. 5, pp. 823–870. January 2007.

[131] H. Luo, L. Tong, X. Li, Y. Chen, X. Liu, M. Li, and Y. Zhang, "Polarimetric SAR image classification based on polarimetric decompostition and neural networks theory", *Proccedings of SPIE 6788*, 2007.

[132] J.B. MacQueen, "Some methods for classification and analysis of multivariate observations", *Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, University of California Press, Berkeley, California, USA, 1967.

[133] P. Mallapragada, R. Jin, A.K. Jain, and Y. Liu, "SemiBoost: boosting for semi-supervised learning", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 11, pp. 2000–2014, November 2009.

[134] B.S. Manjunath, J.R. Ohm, V.V. Vasudevan, and A. Yamada, "Color and Texture Descriptors", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, pp. 703–715, June 2001.

[135] B.S. Manjunath, P. Wu, S. Newsam, and H. Shin, "A texture descriptor for browsing and similarity retrieval", *Journal of Signal Processing: Image Communication*, vol. 16, no. 1-2, pp. 33–43, September 2000.

[136] D. Massonnet and J.-C. Souyris, *Imaging with Synthetic Aperture Radar*, CRC Press, 2008.

[137] U. Maulik, and D. Chakraborty, "A self-trained ensemble with semisupervised SVM: An application to pixel classification of remote sensing imagery", *Pattern Recognition*, vol. 44, no. 3, pp. 615–623, March 2011.

[138] G.J. McLachlan and K.E. Basford, *Mixture Models: Inference and Applications to Clustering*, Marcel Dekker, New York, 1988

[139] R. Min and X. Yang, "Application of AdaBoost in polarimetric SAR image classification," *IEEE Radar Conference,* no. 60772143, 2009.

[140] P.A. Mitchell, "Hyperspectral digital imagery collection experiment (HYDICE)", *Proccedings of SPIE 2587, Geographic Information Systems, Photogrammetry, and Geological/Geophysical Remote Sensing*, 70, November 1995.

[141] W.M. Moon, G. Staples, D.J. Kim, S.E. Park, and K.A Park, "RADARSAT-2 and coastal applications: Surface wind, waterline, and intertidal flat roughness", *Proceedings of the IEEE*, vol. 98, no. 5, pp. 800–815, May 2010.

[142] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, "A Tutorial on Synthetic Aperture Radar", *IEEE Geoscience and Remote Sensing Magazine*, vol. 1, no. 1, pp. 6–43, March 2013.

[143] G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: A review", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 3, pp. 247–259, May 2011.

[144] M. Muhlbaier, A. Topalis, and R. Polikar, "Learn++.NC: Combining Ensemble of Classifiers Combined with Dynamically Weighted Consult-and-Vote for Efficient Incremental Learning of New Classes", *IEEE Transactions on Neural Networks,* vol. 20, no. 1, pp. 152–168, January 2009.

[145] A. Müller, A. Hausold, and P. Strobl, "HySens - DAIS / ROSIS Imaging Spectrometers at DLR", *8th Int. Symp. on Remote Sensing*, Toulouse, France, 2001.

[146] J. Muñoz-Marí, F. Bovolo, L. Gomez-Chova, L. Bruzzone, and G. Camps-Valls, "Semisupervised one-class support vector machines for classification of remote sensing data", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 8, pp. 3188–3197, August 2010

[147] NASA AIRSAR website, http://airsar.jpl.nasa.gov/

[148] Y. Nemoto, H. Nishino, M. Ono, H. Mizutamari, K. Nishikawa, and K. Tanaka, "Japanese Earth Resources Satellite-1 synthetic aperture radar", *Proceedings of the IEEE*, vol. 79, no. 6, pp. 800–809, June 1991.

[149] E. Nezry, *Land Applications of Radar Remote Sensing*, (*Chapter 1: Adaptive Speckle Filtering in Radar Imagery)*, InTech, Dr. Damien Closson (Ed.), 2014.

[150] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, July 2002.

[151] K. Okamoto, S. Ozawa, and S. Abe, "A Fast Incremental Learning Algorithm of RBF Networks with Long-Term Memory", *IEEE International Joint Conference on Neural Networks*, Portland, Oregon, USA, 20-24 July 2003.

[152] C. Oliver, S. Quegan, *Understanding Synthetic Aperture Radar Images*, SciTech Publ., 2004.

[153] K. Ouchi, "Recent Trend and Advance of Synthetic Aperture Radar with Selected Topics", *Remote Sensing*, vol. 5, no. 2, pp. 716–807, February 2013.

[154] MPEG-7 Part 8 (ISO/IEC TR 15938-8) "Extraction and use of MPEG-7 descriptions"

[155] M. Partio, B. Cramariuc, and M. Gabbouj, "An Ordinal Co-occurrence Matrix Framework for Texture Retrieval", *EURASIP Journal on Image and Video Processing*, vol. 2007, Article ID 17358, 15 pages, 2007.

[156] D.R. Paudyal, A., Eiumnoh, and J. Aschbacher, "Textural information in SAR images for land-cover applications", *IEEE International Geoscience and Remote Sensing Symposium*, Firenze, Italy, 10-14 July 1995

[157] J.S. Pearlman, P.S. Barry, C.C. Segal, J. Shepanski, D. Beiso, and S.L. Carman, "Hyperion, a space-based imaging spectrometer," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 6, pp. 1160–1173, June 2003.

[158] R. Polikar, "Ensemble based systems in decision making", *IEEE Circuits and Systems Magazine*, vol. 6, no. 3, pp. 21–45, March 2006.

[159] R. Polikar, L. Udpa, S. Udpa, and V. Honavar, "Learn++: An incremental learning algorithm for supervised neural networks", *IEEE Transactions on System Part C: Man and Cybernetics*, vol. 31, no. 4, pp. 497–508, November 2001.

[160] R.G. Pontius and M. Millones, "Death to Kappa: birth of quantity disagreement and allocation disagreement for accuracy assessment", *International Journal of Remote Sensing*, vol. 32, no. 15, pp. 4407–4429, August 2011.

[161] E. Pottier, J.S. Lee, and L. Ferro-Famil, "Advanced concepts in polarimetry – part 1 (Polarimetric Target Description, Speckle Filtering and Decomposition Theorems)", Technical report, DTIC Document, February 2005. http://handle.dtic.mil/100.2/ADA437482

[162] Z. Qi, A. Yeh, X. Li, and Z. Lin, "A novel algorithm for land use and land cover classification using RADARSAT-2 polarimetric SAR data", *Remote Sensing of Environment*, vol. 118, pp. 21–39, March 2012.

[163] R.K. Raney, A.P. Luscombe, E.J. Langham, and S. Ahmed, "RADARSAT", *Proceedings of the IEEE*, vol. 79, no. 6, pp. 839–849, June 1991.

[164] F. Ratle, G. Camps-Valls, and J. Weston, "Semisupervised neural networks for efficient hyperspectral image classification", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 5, pp. 2271–2282, May 2010.

[165] J.A. Richards, *Remote Sensing with Imaging Radar* (Signals and Communication Technology), Springer-Verlag Berlin Heidelberg, 2009.

[166] G.H. Rosenfield and K. Fitzpatrick-Lins, "A coefficient of agreement as a measure of thematic classification accuracy", *Photogrammetric Engineering and Remote Sensing*, vol. 52, no. 2, pp. 223–227, February 1986.

[167] A. Rosenqvist, M. Shimada, N. Ito, and M. Watanabe, "ALOS PALSAR: A pathfinder mission for global-scale monitoring of the environment", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 11. pp. 3307–3316, November 2007.

[168] C. Sammut and G. Webb, *Encyclopedia of Machine Learning*, pp 515–518, Springer, 2010.

[169] A. Sato, Y. Yamaguchi, and G. Singh, "Four-Component Scattering Power Decomposition With Extended Volume Scattering Model," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 2, pp. 166–170, March 2012.

[170] A. Schistad Solberg and A. K. Jain, "Texture fusion and feature selection applied to SAR imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 35, no. 2, pp. 475–479, March 1997.

[171] M. Seeger, Learning with labeled and unlabeled data, EPFL-REPORT-161327, 2001.

[172] B.M. Shahshahani and D.A. Landgrebe, "The effect of unlabeled samples in reducing the small sample size problem and mitigating the Hughes phenomenon", *IEEE Transactions on Geoscience and Remote Sensing*, vol 32, no. 5, pp. 1087–1095, September 1994.

[173] K. Shanmugan, V. Narayanan, V. Frost, J. Stiles, and J. Holtzman, "Textural Features for Radar Image Analysis", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 19, no. 3, pp. 153–156, July 1981.

[174] M. Shimada, "Advance Land-Observation Satellite (ALOS) and its follow-on Satellite, ALOS-2", 4th International Workshop on Science and Applications of SAR Polarimetry and Polarimetric Interferometry, Frascati, Italy, 26–30 January 2009.

[175] M. Shimoni, D. Borghys, R. Heremans, C. Perneel, and M. Acheroy, "Fusion of PolSAR and PolInSAR data for land cover classification," *International Journal of Applied Earth Observation and Geoinformation*, vol. 11, no. 3, pp. 169–180, June 2009.

[176] M. Simard, S.S. Saatchi, and G. De Grandi, "The Use of Decision Tree and Multiscale Texture for Classification of JERS-1 SAR Data over Tropical Forest," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 38, no. 5, pp. 2310–2321, September 2000.

[177] H. Skriver, "Crop Classification by Multitemporal C-and L-Band Single-and Dual-Polarization and Fully Polarimetric SAR", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 6, pp. 2138–2149, June 2012.

[178] H. Skriver, F. Mattia, G. Satalino, A. Balenzano, V.R.N Pauwels, N.E.C Verhoest, and M. Davidson, "Crop classification using short-revisit multitemporal SAR data,"*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 4, no. 2, pp. 423–431, June 2011.

[179] M. Story and R.G. Congahon, "Accuracy assessment: a user's perspective", *Photogrammetric Engineering and Remote Sensing*, vol. 52, no. 3, pp. 397–399, March 1986.

[180] X. Su, C. He, Q. Feng, X. Deng, and H. Sun, "A Supervised Classification Method Based on Conditional Random Fields With Multiscale Region Connection Calculus Model for SAR Image," *IEEE Geoscience and Remote Sensing Letters*, vol. 8, no. 3, pp. 497–501, May 2011.

[181] M. Swain and D. Ballard, "Color indexing", *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, November 1991.

[182] Mitchell, T. *Machine Learning*, McGraw Hill., p.2, 1997.

[183] A. Torre and P. Capece, "COSMO-SkyMed: The Advanced SAR Instrument", *5th International Conference on Recent Advances in Space Technologies*, Rome, Italy, 9–11 June 2011.

[184] R. Touzi, W M Boerner, J S Lee, and E. Lueneburg, "A review of polarimetry in the context of synthetic aperture radar: concepts and information extraction", *Canadian Journal of Remote Sensing*, vol. 30, no. 3, pp. 380–407, June 2004.

[185] R. Touzi, "Target scattering decomposition in terms of roll invariant target parameters", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 1, pp. 73–84, January 2007.

[186] P. Treitz, P. Howarth, and E. Soulis, "Textural processing of multi-polarization SAR for agricultural crop classification", *IEEE International Geoscience and Remote Sensing Symposium*, Lincoln, Nebraska, USA, 27- 31 May 1996.

[187] D. Tuia and G. Camps-Valls, "Semisupervised Remote Sensing Image Classification with Cluster Kernels" *IEEE Geoscience and Remote Sensing Letters*, vol. 6, no. 2, pp. 224–228, February 2009.

[188] D. Tuia and G. Camps-Valls, "Urban image classification with semisupervised multiscale cluster kernels", *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 4, no. 1, pp. 65–74, January 2011.

[189] D. Tuia, M. Volpi, L. Copa, M. Kanevski, and J. Munoz-Mari, "A Survey of Active Learning Algorithms for Supervised Remote Sensing Image Classification", *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 3, pp. 606–617, June 2011.

[190] V. Turkar, R. Deo, Y.S. Rao, S. Mohan, and A. Das, "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*, vol. 5, no. 3, pp. 936–941, March 2012.

[191] D. Turner and I.H. Woodhouse, "An Icon-Based Synoptic Visualization of Fully Polarimetric Radar Data", *Remote Sensing,* vol. 4, no. 3, pp. 648–660, March 2012.

[192] Y. Tzeng, "A fuzzy neural network to SAR image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 36, no. 1, pp. 301–307, January 1998.

[193] U.S. Geological Survey Images, [Online], http://www.terraserver.com/

[194] S. Uhlmann and S. Kiranyaz, "Evaluation of Classifiers for Polarimetric SAR Classification", *IEEE International Geoscience and Remote Sensing Symposium*, Melbourne, Australia, 21-26 July 2013.

[195] S. Uhlmann and S. Kiranyaz, "Classification of dual- and single polarized SAR Images by Incorporating Visual Features", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 90, pp. 10–22, April 2014.

[196] S. Uhlmann and S. Kiranyaz, "Integrating Color Features in Polarimetric SAR Image Classification", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, no. 4, pp. 2197–2216, April 2014.

[197] S. Uhlmann, S. Kiranyaz, T. Ince, and Moncef Gabbouj, "Dynamic and Data-Driven Classification of Polarimetric SAR Images", *SPIE Remote Sensing: Image and Signal Processing in Remote Sensing*, Prague, Czech Republic, 19-22 September 2011.

[198] S. Uhlmann, S. Kiranyaz, T. Ince, and M. Gabbouj, "SAR Imagery Classification in Extended Feature Space by Collective Network of Binary Classifiers", *European Signal Processing Conference*, Barcelona, Spain, 28 August-2 September 2011.

[199] S. Uhlmann, S. Kiranyaz, T. Ince, and M. Gabbouj, "Polarimetric SAR Images Classification using Collective Network of Binary Classifiers", *Joint Urban Remote Sensing Event,* Munich, Germany, 11-13 April 2011.

[200] S. Uhlmann, S. Kiranyaz, and M. Gabbouj, "Polarimetric SAR Classification using Visual Color Features Extracted over Pseudo Color Images", *IEEE International Geoscience and Remote Sensing Symposium*, Melbourne, Australia, 21-26 July 2013.

[201] S. Uhlmann, S. Kiranyaz, M. Gabbouj, and Turker Ince, "Incremental Evolution of Collective Network of Binary Classifiers for Polarimetric SAR Image Classification", *IEEE International Conference on Image Processing*, Brussels, Belgium, 11-14 September 2011.

[202] S. Uhlmann, S. Kiranyaz, and M. Gabbouj, "Semi-Supervised Learning in Ill-posed SAR Classification", *Remote Sensing*, vol. 6, no. 6, pp. 4801–4830, June 2014.

[203] F.T. Ulaby, R. K. Moore, A. K. Fung, *Microwave remote sensing active and passive*, Volume I-III, Artech House, 1986.

[204] F.T. Ulaby, F. Kouyate, B. Brisco, and T.H.L. Williams, "Textural information in SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 24, no. 2, pp. 235–245, March 1986.

[205] F.T. Ulaby, R.K. Moore, A.K Fung, *Microwave Remote Sensing: Active and Passive*, Addison-Wesley: Reading, MA, USA, Volume I, pp. 2–3, 1986.

[206] S. Uratsuka, M. Satake, T. Kobayashi, T. Umehara, A. Nadai, H. Maeno, H. Masuko, and M. Shimada M., "High-resolution dual-bands interferometric and polarimetric airborne SAR (Pi-SAR) and its applications", *IEEE International Geoscience and Remote Sensing Symposium*, Toronto, Canada, June 24-28, 2002.

[207] J.J. van Zyl, "Application of Cloude's target decomposition theorem to polarimetric imaging radar data," *Proccedings of SPIE*, vol. 1748, pp. 184–212, July 1992.

[208] J.J. van Zyl, "Imaging radar polarization signatures: Theory and observation", *Radio Science*, vol. 22, no. 4, pp. 529–543, July-August 1987.

[209] L.J.P. van der Maaten, E.O. Postma, and H.J. van den Herik, "Dimensionality Reduction: A Comparative Review", Tilburg University Technical Report, TiCC-TR 2009-005, 2009.

[210] J.E. Vogelmann, S.M. Howard, L. Yang, C.R. Larson, B.K. Wylie, and N. Van Driel. "Completion of the 1990s National Land Cover Data Set for the Conterminous United States from Landsat Thematic Mapper Data and Ancillary Data Sources", *Photogrammetric Engineering and Remote Sensing*, vol. 67, pp. 650–662, June 2001.

[211] A. Voisin, V. Krylov, G. Moser, S.B. Serpico, and J. Zerubia, "Classification of Very High Resolution SAR Images of Urban Areas Using Copulas and Texture in a Hierarchical Markov Random Field Model", *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 1, pp. 96–100, January 2013.

[212] P. Wang, W. Hong, and F. Cao, "Investigation of the scattering intensity parameters and the hue-saturation based color space models used in the Cloude-Pottier decomposition method", *IET International Radar Conference*, Guilin, China, 20-22 April 2009.

[213] C. Wen, Y. Zhang, and K. Deng, "Urban Area Classification In High Resolution SAR Based on Texture Features", *International Conference on Geo-spatial Solutions for Emergency Management and the 50th Anniversary of the Chinese Academy of Surveying and Mapping*, Beijing, China, 14-16 September 2009.

[214] C.A. Wiley, "Synthetic aperture radars—A paradigm for technology evolution", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 21, no. 3, pp. 440–443, May 1985.

[215] D.H. Wolpert, "Stacked generalization", *Neural Networks*, vol. 5, no. 2, pp. 241–259, March 1992.

[216] Y. Yamaguchi, T. Moriyama, M. Ishido, and H. Yamada, "Four-component scattering model for polarimetric SAR image decomposition", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 8, pp. 1699–1706, August 2005.

[217] Y. Yamaguchi, A. Sato, W.M. Boerner, R. Sato, and H. Yamada, "Four-component scattering power decomposition with rotation of coherency matrix," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 6, pp. 2251–2258, June 2011.

[218] W. Yang, D. Dai, J. Wu, and C. He, "Weakly Supervised Polarimetric SAR Image Classification With Multi-Modal Markov Aspect Model," *ISPRS TC VII Symposium 100 Years ISPRS, Vienna, Austria, vol. XXXVIII,* 5-7 July 2010.

[219] R. Yang, J. Yang, and Q. Qin, "The application of H-α-Wishart on land cover classification and analysis", *Proccedings of SPIE 7285*, 2008.

[220] W. Yang, T. Zou, D. Dai, and Y. Shuai, "Supervised land-cover classification of TerraSAR-X imagery over urban areas using extremely randomized clustering forests," *Urban Remote Sensing Event,* Shanghai, China, 20-22 May 2009.

[221] H. Yin, Y. Cao, and H. Sun, "Combining pyramid representation and AdaBoost for urban scene classification using high-resolution synthetic aperture radar images", *IET Radar, Sonar and Navigation*, vol. 5, no. 1, p. 58–64, January 2011.

[222] J.J. Yin, J. Yang, and Y. Yamaguchi, "A new method for polarimetric SAR image classification", *2nd Asian-Pacific Conference on Synthetic Aperture Radar*, Xian, China, 26-30 October 2009.

[223] P. Yu, A.K. Qin, and D.A. Clausi, "Polarimetric SAR image segmentation using region growing with edge penalty", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 4, pp. 1302–1317, April 2012.

[224] H.A. Zebker and J.J. van Zyl, "Imaging radar polarimetry: A review", *Proceedings of the IEEE*, vol. 79, no. 6, pp. 1583–1605, June 1991.

[225] Y. Zhang and L. Wu, "A new classifier for polarimetric SAR images," *Progress In Electromagnetics Research*, vol. 94, pp. 83–104, 2009.

[226] L. Zhang, B. Zou, J. Zhang, and Y. Zhang, "Classification of Polarimetric SAR Image Based on Support Vector Machine Using Multiple-Component Scattering Model and Texture Features," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, pp. 1–10, 2010.

[227] Y.A. Zheng, J.S. Song, W.M. Zhou, and R.H. Wang, "False Color Fusion for Multi-band SAR Images Based on Contourlet Transform", *Acta Automatica Sinica*, vol. 33, no. 4, pp. 337-341, April 2007.

[228] X. Zhou and C. Zhang, "A perceptive uniform pseudo-color-coding method of SAR images", *International Conference on Radar,* Shanghai, China, 16-19 October 2006.

[229] Z. Zhou and M. Li, "Tri-training: exploiting unlabeled data using three classifiers", *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 11, pp. 1529–1541, November 2005.

[230] Z. Zhou, "When semi-supervised learning meets ensemble learning", *8th International Workshop on Multiple Classifier Systems*, pp. 529–538, 2009.

[231] X. Zhu, "Semi-supervised learning literature survey", Technical Report 1530, Department of Computer Sciences, University of Wisconsin at Madison, Madison, WI (2005) http://www.cs.wisc.edu/~jerryzhu/pub/ssl_survey.pdf.

[232] T. Zou, W. Yang, D. Dai, and H. Sun, "Polarimetric SAR Image Classification Using Multifeatures Combination and Extremely Randomized Clustering Forests," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, pp. 1–10, 2010.