T.C.

ISTANBUL AYDIN UNIVERSITY

INSTITUTE OF GRADUATE STUDIES

A DEEP LEARNING BASED METHOD FOR CLASSIFICATION OF POLARIMETRIC SAR DATA

MASTER'S THESIS

İkram BEN-WARRAK

Department of Engineering

Electrical and Electronics Engineering Program

MARCH, 2023

T.C.

ISTANBUL AYDIN UNIVERSITY

INSTITUTE OF GRADUATE STUDIES

A DEEP LEARNING BASED METHOD FOR CLASSIFICATION OF POLARIMETRIC SAR DATA

MASTER'S THESIS

İkram BEN-WARRAK

(Y2013.300028)

Department of Engineering

Electrical and Electronics Engineering Program

Thesis Advisor: Assist. Prof. Dr. Necip Gökhan KASAPOĞLU

MARCH, 2023

APPROVEL PAGE

DECLARATION

I hereby declare with respect that the study "A Deep Learning Based Method for Classification of Polarimetric Sar Data", which I submitted as a Master / PhD thesis, is written without any assistance in violation of scientific ethics and traditions in all the processes from the Project phase to the conclusion of the thesis and that the works I have benefited are from those shown in the References. (06/03/2023)

İKRAM BEN -WARRAK

FOREWORD

POLARİMETRİK SAR VERİLERİNİN SINIFLANDIRILMASI İÇİN DERİN ÖĞRENMEYE DAYALI BİR YÖNTEM

ÖZET

Polarimetrik yapay açıklıklı radar (PolSAR) görüntülerinin sınıflandırması önemli bir PolSAR veri uygulamasıdır. Son yıllarda derin öğrenmenin yaygınlaşması, PolSAR görüntü sınıflandırmasında da önemli iyileşmeler sağlamıştır. Bu tezde, polarimetrik yapay açıklıklı radar (SAR) görüntülerinin sınıflandırılması için, bir derin evrişimsel sinir ağını kullanılmaktadır. Karmaşık değerli evrişimsel sinir ağı (CV-CNN), geleneksel evrişimsel sinir ağını (CNN) karmaşık değerli işlemeye yönelik genişletir ve SAR görüntülerinde bulunan genlik ve açı bilgilerini kullanır.

Bu tezde, CV-CNN' nin polarimetrik SAR görüntü sınıflandırma işleminin performans değerlendirilmesi, pikselleri belirli arazi türlerine kategorize etme başarımını içerir. CV-CNN sonuçları SVM gibi bir makine öğrenmesi teknik ve Mahalanobis uzaklığı gibi basit bir istatistiksel tekniği ile karşılaştırmalı olarak, test başarımı, karmaşıklık matrisi kullanılarak verilir.

Anahtar Kelimeler- Makine Öğrenimi, Derin Öğrenme, PolSAR, CV-CN

A DEEP LEARNING BASED METHOD FOR CLASSIFICATION OF POLARIMETRIC SAR DATA

ABSTRACT

Polarimetric synthetic aperture radar (POLSAR) image classification is a crucial POLSAR data application. The widespread use of deep learning in recent years has led to significant improvements in POLSAR image classification. In this thesis, a deep convolutional neural network is used for the classification of polarimetric synthetic aperture radar (SAR) image. The complex-valued convolutional neural network (CV-CNN) extends the traditional convolutional neural network (CNN) to handle complex data and uses the amplitude and angle information found in SAR images.

In this thesis, the performance evaluation of the polarimetric SAR image classification process of CV-CNN includes its performance in categorizing pixels to specific terrain types. The test performance is given using overall accuracy and confusion matrix, comparing CV-CNN results with machine learning technique such as SVM and a simple statistical technique such as Mahalanobis distance.

Keywords- PolSAR, Machine Learning, Deep Learning, CV-CNN**.**

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

A. Introduction

Synthetic aperture radar, or SAR, is a type of active microwave imaging system that can generate detailed images regardless of lighting or weather conditions. A more advanced version of SAR, known as polarimetric SAR or PolSAR, can work in different polarization modes, which allows it to analyze the scattering characteristics of various types of land cover. This makes PolSAR particularly useful for image classification, which has a range of applications across various fields. Synthetic Aperture Radar (SAR) was first developed in the 1950s as a means of creating long-range maps from an aerial or satellite perspective. Since then, it has undergone numerous improvements, including the ability to gather multipolarization data, cover wide swaths of land, and produce multi-frequency radar images. One of the current areas of focus in the field of remote sensing is the use of polarimetry with SAR, which allows for the monitoring of Earth's surface by gathering information about its physical properties. This is achieved using PolSAR (Polarimetric SAR) systems, which are capable of transmitting and receiving electromagnetic waves in both vertical and horizontal polarizations, in contrast to traditional SAR systems that only operate in a single polarization mode. This allows PolSAR to infer the properties of the surface more accurately.

In recent years, the use of fully polarimetric synthetic aperture radar (PolSAR) for microwave remote sensing has become increasingly important, and it has been applied successfully in a variety of contexts. PolSAR image classification, which can be used as a final product for end users or as a foundational step in supporting other applications, has received a great deal of research attention. Over the past three decades, a significant number of algorithms have been created specifically for the purpose of classifying images generated by the polarimetric SAR (PolSAR) technology. The field of polarimetry has a rich history, beginning in the 1700s. George C. Sinclair put forward the idea of using a singular matrix, specifically with regard to

radar cross-section measurements of coherent scattering, in the 1940s. In the 1950s, E.M. Kennaugh made significant contributions to the field. However, there was little advancement in polarimetry until the 1970s, when Huynen's doctoral thesis renewed interest in the field. Despite this renewed interest, the full potential of polarimetry was not realized until the 1980s, when technological advancements in radar devices made it possible to develop polarimetric sensors for both military and civilian applications. Initially, these sensors were primarily used in airborne systems.

Imaging radar has been widely used as a useful tool for remote sensing of the Earth since the successful launch of the SEASAT satellite with a synthetic aperture radar (SAR) system in 1978. SAR is the only practical technique for achieving high spatial resolution with imaging radar and is particularly useful for imaging from space platforms. The method consists of creating a large opening by moving the radar system. Using the AIRSAR system, the National Aeronautics and Space Administration Jet Propulsion Laboratory (NASA-JPL) captured the first polarimetric image in 1985. SAR satellites have also been used to detect ocean phenomena such as currents, surface and internal waves, and sea ice. The ability of imaging SAR systems to operate continuously and collect data in all weather conditions makes them a valuable tool for global Earth monitoring. Image classification, which involves assigning pixels to specific terrain types, is one of the primary applications of PolSAR data. The classification map that results can be used for land cover mapping or as input for further processing. One of the primary benefits of using microwaves for imaging is that they can penetrate clouds, allowing radar to provide its own illumination. This means that SAR can image the Earth at any time of day or night and in most weather conditions. There are now numerous space-borne and airborne SAR (AIRSAR) systems available, which are frequently used as primary remote sensing instruments in conjunction with multispectral radiometers. (G.Mullissa, C. Persello, and V. Tolpekin,2018). The SEASAT satellite was a pioneer in utilizing synthetic aperture radar (SAR) technology for the purpose of observing large areas of oceans and sea ice. It was soon discovered that the technology also had many potential uses for discriminating between different types of terrain and identifying targets.

SEASAT SAR was designed to operate in the L-band frequency range (with a wavelength of 23.5 cm), and it featured a single polarization channel, known as HH (Horizontal transmit and Horizontal receive). Despite the fact that the satellite only

functioned for a brief period of 105 days due to a significant technical malfunction, it effectively showcased the potential of imaging radar technology. This achievement ultimately led to the launch of numerous space-borne SAR systems in the following decades, such as NASA's SIR-A and SIR-B in the early 1980s, the European ERS-1 and ERS-2 in the 1990s, the Japanese JERS-1 in 1992, and the Canadian RADARSAT-1 in 1995. The success of SEASAT SAR also led to an increased focus on developing and researching multipolarization and fully polarimetric imaging radar, which are considered a natural progression from single-polarization SAR technology. In recent decades, various pixel-based classification techniques have been proposed that make use of one of three main types of coherent scattering matrix information: the inherent characteristics of polarimetric scattering mechanisms, statistical properties, or a combination of both polarimetric scattering characteristics and statistical properties. Speckle noise can negatively affect the accuracy of methods that use pixel-wise polarimetric target decomposition parameters. To improve performance, researchers have looked into incorporating semantic information like spatial relations, texture, and shape, resulting in the creation of various region-based methods. A popular method for overcoming the limitations of pixel-based classification is the use of Markov random fields or conditional random fields to account for spatial interactions and segment data into distinct objects. Another approach that some researchers have taken is to use neural network models in conjunction with superpixel segmentation to improve performance. Region-based approaches can enhance classification outcomes, but they have limitations in their capacity to extract basic, manually designed spatial features from the data, and their capability to represent and differentiate may be limited. This is particularly true for high-resolution PolSAR data, for which shallow methods are often ineffective. To effectively classify this type of data, it is necessary to have a deep understanding of the underlying physics and to use a deep learning model that can automatically discover high-level spatial representations from the data.

There are several traditional methods proposed for PolSAR image classification, such as wishart classifiers, target decompositions (TDs), and random fields (RFs). These methods frequently rely on low-level or mid-level characteristics obtained through feature engineering. However, these features may not be sufficient for accurately classifying high-resolution PolSAR data, particularly when dealing with complex scenes or multiple scattering effects. As a result, there has been a recent

3

movement for the use of deep learning algorithms, which can learn high-level features from data and have demonstrated promising results in PolSAR image classification applications. Traditionally, features used for PolSAR image classification have included statistical features, backscattering elements, TD-based features, and features commonly used for other types of images. However, these features are often class-specific and require manual engineering, and the process of selecting and extracting them can be time-consuming and involve a considerable amount of trial and error. Furthermore, some features, like TD-based characteristics, rely substantially on complicated PolSAR data analysis and might be computationally costly. (J. Geng, H. Wang, J. Fan and X. Ma,2018).

Recently, deep learning has become increasingly popular in the field of remote sensing, especially for the classification of synthetic aperture radar (SAR) images. This is because deep learning techniques can automatically find useful features from data, which leads to better classification accuracy and less need for manual feature engineering. Unlike shallow models like support vector machines and random forests that depend on a lot of input features and may not be effective with nonlinear data, deep learning techniques, particularly deep convolutional neural networks, have the ability to learn hierarchical representations of the data, enabling them to better understand the intricate relationships and patterns within the data. This is why deep learning has shown promising results in SAR image classification.

B. Synthetic Aperture Radar (SAR):

Synthetic Aperture Radar (SAR) is a radar system used to generate high-resolution images of the Earth's surface. It works by transmitting a radar pulse and measuring the time delay between the emission and the reflection back from the surface to create a detailed image. SAR has several benefits compared to other imaging systems. It operates 24/7, unaffected by atmospheric conditions such as clouds, and generates images with remarkable resolution, making it ideal for purposes like mapping, disaster response, and analyzing land use.

A SAR system is comprised of three key components: a pulsed microwave transmitter, an antenna for both transmitting and receiving signals, and a receiver unit. This system is mounted on a moving platform and operates in a sidelooking configuration, with the antenna aimed perpendicular to the direction of the platform's movement (azimuth)

and angled towards the ground. The area captured by the antenna's beam in the ground range (x) and azimuth (y) directions is known as the antenna footprint. As the platform moves, the antenna scans the area, creating an image swath, which is the total area covered by the antenna's beam. This allows the system to produce a complete scan of the desired area.
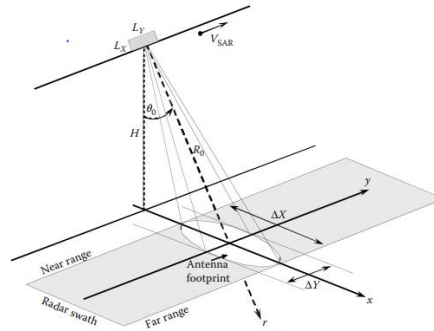
Figure 1. SAR imaging geometry in strip-map mode.

C. Polarimetric Synthetic Aperture Radar (POLSAR):

POLSAR is a cutting-edge variation of Synthetic Aperture Radar (SAR) that offers enhanced capabilities. Unlike standard SAR systems that only operate in a single polarization mode, PolSAR is equipped to work in different polarization modes, providing a more comprehensive understanding of the scattering properties of various land cover types. This capability is made possible by the system's ability to transmit and receive electromagnetic waves in both horizontal and vertical polarizations. This advanced feature enhances the accuracy of surface property determination, making PolSAR particularly useful for tasks such as image classification. In addition to providing information about surface properties, PolSAR can also be utilized to monitor changes in land cover and assess the environmental impact of human activities.

PolSAR is a unique type of radar system that has the ability to measure not only the backscattered energy of objects in a scene but also their polarization state. This makes it distinct from traditional radar systems, which operate using a single, fixed polarization antenna for both transmission and reception. The goal of using PolSAR is to gather information about the nature of land features by analyzing changes in the polarization state of electromagnetic waves reflected from the Earth's surface. There are several methods for interpreting PolSAR data, including scattering matrices, covariance matrices, decomposition techniques, and polarimetric signatures. (Haixia

Bi, Jian Sun , Zongben Xu 2019). These methods allow for the representation of backscatter responses for all possible combinations of transmit and receive polarizations, thereby giving a complete understanding of the polarimetric information obtained from the system.
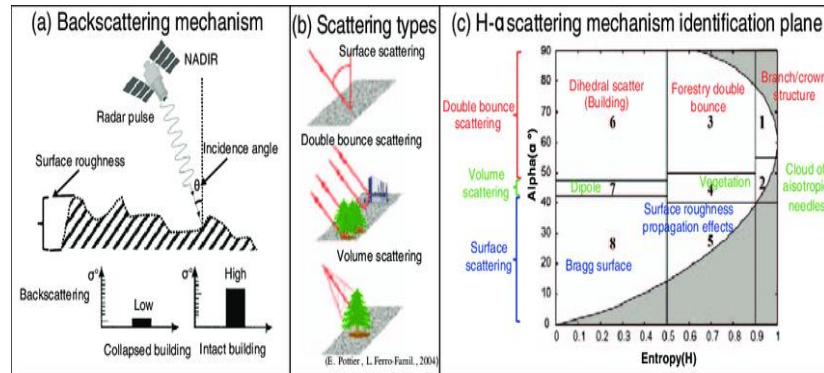
Figure 2. The illustration of polarimetric SAR scattering mechanism, consisting of three parts: (a) representation of backscattering mechanism, (b) depiction of classical polarimetric SAR scattering categories, and (c) visualization of H-alpha scattering mechanism recognition plane.

D. Frequency Bands used for SAR/POLSAR Systems:

Synthetic aperture radar (SAR) systems use radio waves in different frequency bands to create images of the Earth's surface. The frequency band that is used can affect the resolution and penetration ability of the radar.

· L band (1-2 GHz): L band radar has a relatively long wavelength, which allows it to penetrate through vegetation and shallow water. It is often used for mapping and remote sensing applications.

· C band (4-8 GHz): C band radar has a shorter wavelength than L band radar, which allows for higher resolution images. It is commonly used for mapping and monitoring applications, and it can also penetrate through clouds and smoke.

· X band (8-12 GHz): X band radar has a shorter wavelength than C band radar, which allows for even higher resolution images. It is often used for high-resolution imaging applications.

· P band (140-190 MHz): P band radar has a very long wavelength, which allows it to penetrate through dense materials such as soil and rock. It is used for

specialized applications such as subsurface imaging and underground utility detection.

- S band (2-4 GHz): S band radar is commonly used for radar altimeter systems, which measure the altitude of aircraft above the ground. It has a shorter wavelength than L band radar, which allows for higher resolution measurements.

- Ku band (12-18 GHz): Ku band radar is often used for satellite communications and remote sensing applications. It has a shorter wavelength than C band radar, which allows for higher resolution images.

- Ka band (30-50 GHz): Ka band radar has a very short wavelength, which allows for extremely high-resolution images. It is used for high-resolution imaging and radar altimeter systems.

Table 1. Types of SAR bands used and their respective wavelength and frequencies.

Band	Wavelength (cm)	Frequency (GHz)
P	30-100	0.3-1.0
L	15-30	1.0-2.0
S	7.5-15	2.0-4.0
C	3.75-7.5	4.0-8.0
X	2.4-3.75	8.0-12.5
Ku	1.67-2.4	12.5-18.0
K	1.13-1.67	18.0-26.5
Ka	0.75-1.13	26.5-40.0

A radar system operates by sending out a radar pulse and then measuring the reflection that comes back from the surface. By examining the time between the pulse being emitted and the reflection being received, the system can create an image of the surface. The choice of frequency band can impact the radar's capacity for detail and its ability to penetrate.

E. Deep Learning:

Deep learning involves the use of artificial neural networks (ANNs) that have several layers of processing units. These units, referred to as "neurons," are organized in a sequence of interconnected layers. Each layer learns to identify increasingly complex and abstract features as data moves through the network. This hierarchical learning of features enables deep learning algorithms to attain high precision in diverse

tasks like image classification and object detection as they learn to spot patterns and features that might not be easily noticeable to humans. In the area of remote sensing, deep learning has been utilized for various purposes, such as object detection, land cover classification, and image analysis. Deep learning, particularly through the use of convolutional neural networks, has demonstrated strong capabilities in various fields, including natural language processing and image processing. It has the potential to be applied to other areas as well. Deep learning algorithms, such as CNNs, are able to learn more abstract feature representations compared to traditional hand-engineered filters, leading to improved generalization performance in image classification tasks. The successful application of deep learning techniques in image classification has been driven by the availability of big data, advanced algorithms, and improved computing power. These same factors are present in the field of PolSAR image classification, making it a promising area for the use of deep learning techniques to improve classification results.

PolSAR is a type of radar system that is designed to measure the amount of energy that is reflected from objects in a scene, along with their polarization state. Unlike traditional radar systems, which use a single polarization antenna, PolSAR can operate in different polarization modes, allowing it to gather more comprehensive information about the features of a scene. It is widely used in remote sensing applications for tasks such as land cover classification, object detection, and image interpretation. PolSAR is a radar system used for remote sensing purposes that can measure the energy of objects in a scene that is reflected back to the radar as well as their polarization state. Unlike traditional radar systems that only use one fixed polarization, PolSAR systems can operate in different polarization modes. The different polarization states of electromagnetic waves can reveal information about the characteristics of the land features being imaged. To understand the data gathered by PolSAR, various techniques such as scattering matrices, covariance matrices, and polarimetric signatures are utilized. Recent advances in the field of PolSAR image classification have been made possible by the use of deep learning techniques, which have yielded promising results.

Unsupervised methods, on the other hand, do not rely on labeled data and instead attempt to cluster similar data points together based on the features of the data. These techniques are useful in situations where labeled data is scarce or expensive to

obtain. Additionally, semi-supervised methods make use of a limited amount of labeled data and additional unlabeled data to learn the relationship between the input and output variables. With the development of deep learning techniques, a growing number of studies have applied these approaches to PolSAR image classification tasks, resulting in improved accuracy and more robust performance compared to traditional methods. Unsupervised methods in PolSAR image classification attempt to categorize pixels based on polarimetric statistical principles without the use of any labeled information. These methods generally employ polarimetric target decomposition theories such as Cloude-Pottier decomposition, Freeman decomposition, and four-component decomposition to produce an initial categorization. Further refinement of the results is then performed using iterative algorithms that are based on different polarimetric statistical laws like the complex Wishart distribution, K-distribution, and U-distribution. Although unsupervised methods are convenient and quick to execute, their accuracy is typically lower compared to supervised methods, making them more suitable for preliminary analysis than final categorization.

The objective is to estimate the mapping function in such a way that new input data can be utilized to predict the output variables. Labeling the data for a machine learning model to learn from can be a challenging and time-consuming process. In supervised learning, a dataset with previously known correct answers is used to train the algorithm to make predictions. The algorithm predicts the outcome and is corrected by the known answers, continuing this process until its accuracy reaches an acceptable level. This type of learning is called "supervised," as the training dataset serves as a guide for the learning process. Popular supervised methods for PolSAR image classification include Bayesian classification, neural networks, support vector machines (SVM), and k-nearest neighbor, due to their high classification accuracy (CA).
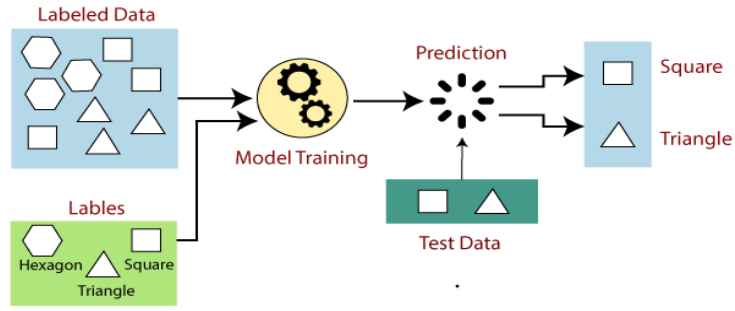
Figure 3. Supervised Learning

F. Literature Reviews:

Deep learning algorithms are a type of machine learning that is based on artificial neural networks. They have become increasingly popular in recent years due to their ability to process large amounts of data and extract meaningful features from it. This makes them particularly well-suited for classification tasks, such as identifying objects in images, recognizing speech, and analyzing text. Unlike traditional machine learning algorithms that rely on manual feature engineering, deep learning algorithms are capable of automatically learning hierarchical representations of data, which can often lead to better performance. There are several types of deep learning algorithms, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), and long short-term memory (LSTM) networks. These algorithms can be trained end-to-end, meaning they can learn directly from the raw data without the need for manual feature engineering. This makes the training process more efficient and effective.

In terms of evaluating the performance of deep learning classification models, researchers often use a variety of metrics, including accuracy, precision, recall, and the F1 score. These metrics are used to assess the model's performance on a test dataset and can help researchers compare different models and identify areas for improvement. Overall, the field of deep learning classification is still evolving, and there is a wealth of opportunity for further research and innovation. The literature on the topic continues to grow rapidly, with many different approaches and methods being proposed.

A new approach for PolSAR image classification was presented in a research study. The method employs an advanced deep Q-network (DQN) technique. It generates a large amount of useful data by collaborating with a reinforcement learning agent and employing an -greedy technique. The first step in the process is to preprocess

the PolSAR data to minimize the effect of speckle noise and extract multiple features. Then, the extracted features and the corresponding training images are input into a deep reinforcement learning model tailored specifically for PolSAR image classification. The experiment results showed that the proposed method had superior classification performance compared to conventional supervised classification techniques like convolutional neural networks, random forests, and linear support vector machines with L2 losses. (Chen-Fu Chien, Yun-Siang Lin & Sheng-Kai Lin,2020).

Another research study proposed a way to enhance the accuracy of classifying polarimetric synthetic aperture radar (PolSAR) images using deep learning techniques. The approach involves pre-treating the images by reducing their dimensionality, breaking them down and rebuilding them using wavelet techniques, applying anisotropic filtering, and combining the images into gr $ayscale. The processed images are then fed into a DeepLabV3+ deep learning architecture for training. The results showed that this method resulted in a 2.3%–3.0% increase in the Kappa coefficient, which measures classification accuracy. This improvement highlights that the proposed method effectively improves the classification accuracy of PolSAR images through image preprocessing without the need for additional stacking. (Mohsen Jafari, Yasser Maghsoudi,2015).

A method has been proposed that uses a unique deep learning network to automatically extract features from raw input data and include spatial information between pixels in PolSAR images in the input data. The deep structured feature network (DSFN) requires fewer parameters to be adjusted during pre-training and fine-tuning compared to traditional deep networks. The proposed method was tested using real PolSAR data and was found to be efficient with a higher classification accuracy compared to conventional deep networks. (Yongqing Wang, Yanbo Luo, Hao Li, and Qiushi Chen,2022).

G. Research Objective and Motivation:

The purpose of this study is to find the best method for classifying land based on polarimetric characteristics using PolSARpro software and L-band AIRSAR imagery from Flevoland, which has a resolution of 1024 x 750 pixels. The study used a pre-existing set of 15 categories as a reference. The results showed that deep learning

methods, including CNNs, were successful in interpreting PolSAR data and generally performed better than traditional classifiers. However, the performance of deep learning networks is greatly influenced by the number of labeled samples available. The findings of the study provided insight into the most effective method for PolSAR image classification.

The study aimed to compare the accuracy of PolSAR data classification between traditional classifiers and deep learning methods. Specifically, the study used a support vector machine (SVM) classifier and a Complex-Valued convolutional neural network (CV-CNN) model. The goal of the study is to determine the most effective method for land classification using a dataset that consists of 15 categories. The sample data used in the study was L-band AIRSAR imagery from Flevoland.

II. POLSAR FEATURE EXTRACTION

A. Introduction

Feature extraction is a technique used to reduce the number of resources required to analyze large sets of data. The process involves transforming the original dataset into a set of illustrative features that briefly and precisely represent the original information. This is done to make the data more manageable and reduce the computational resources required for analysis. The goal of feature extraction is to remove unnecessary data and minimize the memory and processing power needed to analyze the data.

The analysis of complex datasets can be a challenging task due to the high number of variables involved. This can result in overfitting, which is when the classification algorithm is too tailored to the training samples, affecting its ability to generalize to new samples. To address this, feature extraction is used to reduce the number of variables and make the data more manageable for analysis. This process can also help preserve sensor-specific information in the case of data from multiple sensors. Properly extracted features can provide a physical interpretation of the target being observed. Another related step, feature selection, involves choosing a subset of features to identify the most valuable information for a specific application. (Si-Wei Chen,2022). When it comes to analyzing PolSAR data, the process of feature extraction involves analyzing the intensity, polarimetric, and texture information of the data. This can be achieved through different techniques, including statistical analysis, geometric methods, and the extraction of textural features. The objective is to identify the most significant features that can be utilized to represent the information in a more concise and meaningful manner.

PolSAR data contains complex values that are challenging to use directly in machine learning algorithms. Therefore, feature extraction is an essential step in the analysis of PolSAR data. Support Vector Machine (SVM) and Convolutional Neural Network (CNN) are two popular algorithms used for PolSAR data classification, and they differ significantly in their approaches to feature extraction. SVM requires explicit feature extraction from the complex valued PolSAR data. Three common methods for feature extraction are the Pauli, Cloude-Pottier, and Huynen

decompositions. These methods transform the complex data into real-valued features that can be used as input to the SVM algorithm. The feature extraction process is typically the same for all applications of SVM to PolSAR data. In contrast, CNNs can learn relevant features for classification directly from the PolSAR data without explicit feature extraction. The PolSAR data is used as input to the network, and the network is trained to learn features that are relevant for classification. The CNN network typically consists of convolutional layers that learn spatial features and fully connected layers that perform classification. CNNs are better suited for larger datasets and are better at learning complex features. In summary, SVM requires explicit feature extraction from the PolSAR data, while CNNs can learn features directly from the data. SVM is generally more suited for smaller datasets, while CNNs can handle larger datasets and are better at learning complex features. (S. Chen and C. Tao,2018).

B. Scattering Matrix

The scattering matrix is a crucial tool for understanding the scattering characteristics of an object in polarimetric radar data. It is a 2x2 matrix that summarizes the polarimetric information of the target being analyzed and is made up of complex numbers that represent the amplitude and phase of the radar signals returned to the sensor with different polarizations.

For PolSAR data, the scattering matrix is defined as:

$$S = \begin{bmatrix} S_{hh} & S_{hv} \\ S_{vh} & S_{vv} \end{bmatrix} \qquad (2.1)$$

Where S_{hh}, S_{hv}, S_{vh}, and S_{vv} are the elements of the matrix and represent the backscattered radar signals for horizontal-horizontal (HH), horizontal-vertical (HV), vertical-horizontal (VH), and vertical-vertical (VV) polarizations, respectively. The elements of the scattering matrix can be calculated from the radar signals using the following equations:

$$S_{hh} = \left(\frac{1}{N}\right) \sum_{i=1}^{i=N} (|hh_i|^2) \qquad (2.2)$$

$$S_{hv} = \left(\frac{1}{N}\right) \sum_{i=1}^{i=N} (|hv_i|^2) \qquad (2.3)$$

$$S_{vh} = \left(\frac{1}{N}\right) \sum_{i=1}^{i=N} (|vh_i|^2) \qquad (2.4)$$

$$S_{vv} = \left(\frac{1}{N}\right)\Sigma_{i=1}^{i=N}(|vv_i|^2) \qquad (2.5)$$

Where N is the number of pixels in the image, hh_i, hv_i, vh_i, and vv_i are the backscattered radar signals for the i-th pixel.

C. Covariance and Coherency Matrix

The C_3 matrix (also known as the volume scattering covariance matrix) is a 3x3 matrix that describes the second-order statistical properties of the backscattered radar signals for a specific scattering mechanism. The elements of the matrix are calculated as the covariance between the different channels of the radar signals. The C_3 matrix is defined as:

$$C_3 = \begin{bmatrix} C_{3hh} & C_{3hv} & C_{3vh} \\ C_{3vh} & C_{3vv} & C_{3vv} \\ C_{3hh} & C_{3hv} & C_{3vh} \end{bmatrix} \qquad (2.6)$$

Where C_{3hh}, C_{3hv}, and C_{3vv} are the elements of the matrix and represent the covariance between the backscattered radar signals for horizontal-horizontal (HH), horizontal-vertical (HV), and vertical-vertical (VV) polarizations, respectively.

The elements of the C_3 matrix can be calculated using the following equations:

$$C_{3hh} = \left(\frac{1}{N}\right)\Sigma_{i=1}^{i=N}[hh_i - mu_{3hh}](hh_i - mu_{3hh})^T \qquad (2.7)$$

$$C_{3vh} = \left(\frac{1}{N}\right)\Sigma_{i=1}^{i=N}[vh_i - mu_{3vh}](vh_i - mu_{3vh})^T \qquad (2.8)$$

$$C_{3hv} = \left(\frac{1}{N}\right)\Sigma_{i=1}^{i=N}[hv_i - mu_{3hv}](hv_i - mu_{3hv})^T \qquad (2.9)$$

$$C_{3vv} = \left(\frac{1}{N}\right)\Sigma_{i=1}^{i=N}[vv_i - mu_{3vv}](vv_i - mu_{3vv})^T \qquad (2.10)$$

The coherency matrix is a 3x3 matrix that describes the coherence between the different channels of the radar signals. The elements of the coherency matrix are calculated as the normalized cross-covariance between the different channels of the radar signals. The coherency matrix is defined as:

$$T = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix} \qquad (2.11)$$

Where T_{11}, T_{12}, T_{13}, T_{22}, T_{23} and T_{33} are the elements of the matrix and represent the coherency between the backscattered radar signals for different polarizations. The elements of the coherency matrix can be calculated using the following equations:

$$T_{11} = (C_{hh} - |mu_{hh}|^2)/(|E_{hh}|^2) \qquad (2.12)$$

$$T_{12} = (C_{hv} - Re\{mu_{hv} * mu_{hh}^*\})/(|E_{hh}| * |E_{hv}|) \qquad (2.13)$$

$$T_{13} = (C_{vh} - Re\{mu_{vh} * mu_{hh}^*\})/(|E_{hh}| * |E_{hv}|) \qquad (2.14)$$

$$T_{21} = (C_{hv} - Re\{mu_{hv} * mu_{vv}^*\})/(|E_{hh}| * |E_{vv}|) \qquad (2.15)$$

$$T_{22} = (C_{vv} - |mu_{vv}|^2)/(|E_{vv}|^2) \qquad (2.16)$$

$$T_{23} = (C_{vh} - Re\{mu_{vh} * mu_{vv}^*\})/(|E_{vv}| * |E_{hv}|) \qquad (2.17)$$

$$T_{31} = (C_{vh} - Re\{mu_{vh} * mu_{hv}^*\})/(|E_{hh}| * |E_{vv}|) \qquad (2.18)$$

$$T_{32} = (C_{vh} - Re\{mu_{vh} * mu_{hv}^*\})/(|E_{vv}| * |E_{hv}|) \qquad (2.19)$$

$$T_{33} = (C_{vv} - |mu_{vh}|^2)/(|E_{hv}|^2) \qquad (2.20)$$

Where C_{hh}, C_{hv}, and C_{vv} are the elements of the covariance matrix, mu_{hh}, mu_{hv} and mu_{vv} are the mean of the respective channel, E_{hh}, E_{hv} and E_{vv} are the mean of the respective channel.

D. Feature Extraction

Feature extraction is a technique used to condense multiple data groups into a more manageable size, allowing for greater control over the data. This process is necessary as working with large, unfiltered data sets can consume a significant amount of computational resources. By implementing feature extraction, we can reduce the number of resources required for data processing while still maintaining important information. Table 2 provides an overview of some of the most used features in this process, along with a brief definition of each feature. (M.AL-Bayati,2020).

Table 2. Some of the most used features with definition

Feature	Description	Definition
Original features	Scattering matrix	[S]
	Covariance matrix	$[C_3]$
	Coherency matrix	$[T_3]$
Decomposition feature	Pauli decomposition	$\alpha = \left\|\dfrac{S_{hh} + S_{vv}}{\sqrt{2}}\right\|$ $\beta = \left\|\dfrac{S_{hh} - S_{vv}}{\sqrt{2}}\right\|$ $\gamma = \left\|\dfrac{S_{hv} + S_{vv}}{\sqrt{2}}\right\|$
	Huynen decomposition	T_{11}, T_{22}, T_{33}
	Krogager decomposition	K_H, K_S, K_D
	Entropy\Anisotropy\Alpha	H, A, α
	Barnes	T_{11}, T_{22}, T_{33}
	Freeman Durden	P_S, P_D, P_V

III. DATA CLASSIFICATION

A. Target Decomposition

Target decomposition is a method utilized in machine learning to break down a complicated target variable into multiple simpler parts. This approach is beneficial when the target variable is composed of several underlying factors or features, and the objective is to comprehend and forecast the connections between these factors and the target variable. By breaking down the target variable into simpler components, it becomes easier to pinpoint the primary factors that influence the target variable and to construct more precise predictive models.

In target decomposition, there are two methods to decompose a target variable:

- o Coherent target decomposition is a method of breaking down the target variable into components that are directly related to the original target variable and retain the majority of its information. These components are usually easy to interpret and have a clear physical meaning.

- o Incoherent target decomposition is a technique where the complex target variable is broken down into multiple components that are not directly related to the original target variable and do not retain much of its information. These components are often difficult to interpret and lack a clear physical meaning.

Coherent target decomposition is typically chosen when it is important to understand and interpret the underlying factors that influence the target variable, whereas incoherent target decomposition is mainly used when the main objective is to extract features that can be used for prediction.

1. Pauli Decomposition

The Pauli decomposition is a technique used to decompose a complex-valued PolSAR image into three separate images, each representing a different polarization state. The method is based on the Pauli matrices, which are a set of three 2x2 matrices that serve as a basis for the space of 2x2 complex matrices. The Pauli decomposition helps to separate the different polarization states in a PolSAR image, providing more information about the scattering mechanism and the nature of the target.

The decomposition is performed by taking the inner product of the complex valued PolSAR image with each of the Pauli matrices, resulting in three images, each of which represents a different polarization state. The scattering S matrix is expressed as the complex sum of the Pauli matrices, where each basis matrix is connected with an elementary scattering process. (Jong, Pottier.p:214).

$$S = \begin{bmatrix} S_{hh} & S_{hv} \\ S_{vh} & S_{vv} \end{bmatrix} = \frac{a}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{b}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} + \frac{c}{\sqrt{2}} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} + \frac{d}{\sqrt{2}} \begin{bmatrix} 0 & -j \\ j & 0 \end{bmatrix} \qquad (3.1)$$

where a, b, c, and d are all complex and are given by:

$$a = \frac{S_{hh}+S_{vv}}{\sqrt{2}} \qquad b = \frac{S_{hh}-S_{vv}}{\sqrt{2}} \qquad c = \frac{S_{hv}+S_{vh}}{\sqrt{2}} \qquad d = j\frac{S_{hv}-S_{vh}}{\sqrt{2}} \qquad (3.2)$$

The application of the Pauli decomposition to deterministic targets is a coherent composition of four scattering mechanisms: the first is single scattering from a plane surface (single or odd-bounce scattering), the second and third are diplane scattering (double or even-bounce scattering) from corners with relative orientations of 0 and 45, respectively, and the final element is all the antisymmetric components of the scattering S matrix. These interpretations are based on an examination of the properties of the Pauli matrices when their wave polarization changes.

In the monostatic case, where $S_{hv} = S_{vh}$, the Pauli matrix basis can be reduced to the first three matrices leading to d =0. It follows the Span value given by:

$$Span = |S_{hh}|^2 + 2|S_{hv}|^2 + |S_{vv}|^2 = |a|^2 + |b|^2 + |c|^2 \qquad (3.3)$$

By decomposing the complex-valued PolSAR image into these three different images, the Pauli decomposition allows for the extraction of the polarimetric information of the scene. The co-polarized (HH or VV) and cross-polarized (HV or VH) images are used to extract the polarimetric information of the scene. (Si-Wei Chen 2022). The total power image is used as a reference. After the decomposition, these images can be used as input to various classification algorithms, such as Random Forest, Support Vector Machine, and Neural Networks, to classify the image based on the polarization information. It's important to note that the Pauli decomposition is a pre-processing step for PolSAR image classification and is not a classification algorithm by itself.

B. Unsupervised Classification

Unsupervised classification algorithms categorize images without any previous knowledge of the surface, obviating the need for ground truth maps or training samples. This classification is performed automatically by locating clusters based on a particular parameter. There are three benefits to using this technique. To begin with, as previously noted, prior knowledge of the target to be classified is not necessary for the initial isolation of the pixels of the image. Unsupervised approaches also reduce human errors because the analyst is not needed to make repeated decisions throughout the categorization process. Supervised classification may miss certain known categories, while unsupervised classification takes them into account. However, unsupervised classification also has its drawbacks, such as limited control by the expert over the categories selected by the classification method and a lack of correlation between the selected information categories and the natural groups of spectral categories. (Umut, Ö. 2020).

The study will apply various unsupervised techniques such as H/alpha/A decomposition, Freeman and Durdan, and Huynen decomposition, which will be analyzed and evaluated using MATLAB.

2. H/Alpha/A Decomposition Theorem

H/alpha/A decomposition is a technique used to decompose a PolSAR image into three components: the HH, HV, and VV channels. These channels are also known as the H-alpha-A decomposition.

The H-alpha-A decomposition assumes that the scattering mechanisms of a target can be modeled as a combination of three types of scattering: isotropic, volumetric, and double-bounce. The H-alpha-A decomposition uses the coherency matrix of the PolSAR data to separate these three scattering mechanisms into three channels. The HH channel is related to the isotropic scattering, the HV channel is related to the volume scattering, and the VV channel is related to the double-bounce scattering. T_3 Coherency matrix can be written as:

$$T_3 = U_3 \Sigma U_3^{-1} \qquad (3.4)$$

Where $\sum$ 3 x 3 is diagonal matrix with non-negative reel elements, contain the T3 eigenvalues.

$$\Sigma = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \qquad (3.5)$$

While $U_3 = [U_1, U_2, U_3]$ is a 3x3 eigenvectors matrix of the T3 matrix. The eigenvector of the averaged coherency matrix can be written us:

$$U = [\cos \alpha e^{j\phi} \quad \sin \alpha \cos \beta e^{j(\delta+\phi)} \quad \sin \alpha \sin \beta e^{j(\gamma+\phi)}]^T \qquad (3.6)$$

After finding the eigenvalues and the eigenvectors of T_3, it is now possible to calculate the entropy and alpha. H/alpha/A decomposition provides scattering information of the surface, and it is often used for the classification of PolSAR images.

a. Polarimetric scattering Entropy (H)

Entropy is a measure of the impurity or disorder of a system. In the context of image classification, entropy is often used as a parameter to measure the homogeneity of a segmented image. In information theory, the entropy of a random variable X is defined as:

$$H = -\sum_{k=1}^{N} P_k \log_N (P_k) \qquad (3.8)$$

Where N is the polarimetric dimension, and P_k the pseudo-probabilities determined from the eigenvalues like:

$$P_k = \frac{\lambda_k}{\sum_{i=1}^{N} \lambda_i} \qquad (3.9)$$

In image classification, the entropy of a segmented image serves as a measure of its homogeneity. If the entropy is high, it means the segments are diverse and not homogeneous, whereas a low entropy value indicates that the segments are homogeneous. The entropy is used to assess the effectiveness of unsupervised classification algorithms, with low entropy and high homogeneity being desirable outcomes. It is also used to judge the quality of the segmentation process. If the entropy is high, it suggests that the segmentation is poor, whereas a low entropy value indicates good segmentation. (S. Chen and C. Tao,2018).

b. Polarimetric scattering parameter α

In the context of radar remote sensing, the polarimetric scattering parameter α is a dimensionless parameter that is used to describe the scattering properties of a

target. The scattering angle is the third relevant parameter; this represents the type of scattering mechanism in play. The angle is defined as:

$$\alpha = \sum_{i=1}^{N} P_i \, \alpha_i \qquad (3.10)$$

When α approaches π/4, this represents the volume scattering component; when α is π/2, this represents the dihedral scattering component; and when α is 0, this represents the odd scattering component. (Zhou, Wang, 2016). α can be used to classify different types of scatterers and surfaces such as vegetation, soil, water, man-made structures, and urban areas. For example, vegetation typically has a low value of α, while urban areas have a high value of α. It is used to extract different scattering information from a target, which is important for the classification of radar images, and it is usually combined with other parameters like Entropy and other polarimetric parameters to get a more accurate classification.
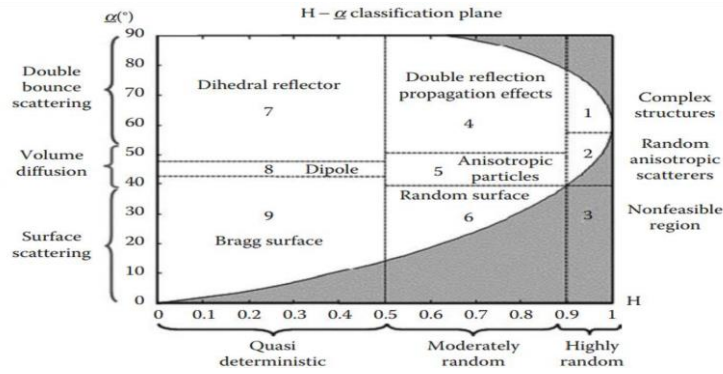
Figure 4. 2-dimensional H/ α

c. Polarimetric scattering parameter A

The polarimetric scattering parameter A is another parameter used in the context of radar remote sensing to describe the scattering properties of a target. It is defined as the ratio of the co-polarization power to the total power. The polarimetric scattering parameter A can be defined as:

$$A = \frac{\lambda_2 - \lambda_3}{\lambda_2 + \lambda_3} \qquad (3.11)$$

With $\lambda_1 > \lambda_2 > \lambda_3 > 0$.

The value of A ranges from 0 to 1, where a value of 0 indicates completely cross-polarized scattering and a value of 1 indicates completely co-polarized scattering. A is related to the degree of depolarization of the target, and it can be used to classify

different types of scatterers and surfaces such as vegetation, soil, water, man-made structures, and urban areas. For example, vegetation typically has a low value of A, while urban areas have a high value of A. It is used to extract different scattering information from a target, which is important for the classification of radar images, and it is usually combined with other parameters like entropy and other polarimetric parameters to get a more accurate classification.

3. Freeman-Durden three-component decomposition

The Freeman-Durden three-component decomposition is frequently employed in the classification of PolSAR pictures to obtain information about the scattering capabilities of the various targets in the image. By decomposing the image into the isotropic, volume, and surface scattering components, it is possible to identify different types of land cover and vegetation. For example, the isotropic scattering component can be used to identify areas of water, the volume scattering component can be used to identify areas of vegetation, and the surface scattering component can be used to identify areas of urban or man-made structures. Once the image has been decomposed into its three scattering components, various classification techniques can be applied to the resulting channels to classify the image into different land cover classes. Commonly used techniques include clustering algorithms like k-means, decision trees, and support vector machines. (G. Mullissa,2019).

The decomposition of the coherency matrix into three scattering components using P_d, P_s, and P_v is done differently. The decomposition is based on the following equation:

$$S = P_{ds\_iso} + P_{ss\_vol} + P_v * S_{sur} \qquad (3.12)$$

Where S is the coherency matrix, P_{ds_iso} is the backscattering power of the isotropic component, P_{ss_vol} is the backscattering power of the volume component, Pv is the backscattering power of the surface component, S_{iso}, S_{vol}, and S_{sur} are the scattering matrices of the three scattering components.

> - $S_{iso} = [1,0,1]^T$ is the isotropic scattering matrix.

> - $S_{vol} = [\cos(2\varphi), \sin(2\varphi), 0]^T$ is the volume scattering matrix.

> - $S_{sur} = [\cos(2\theta), 0, \sin(2\theta)]^T$ is the surface scattering matrix.

Where φ is the phase angle and θ is the scattering angle.

P_d, P_s, and P_v can be calculated from the coherency matrix elements using the following equations:

$$P_d = \frac{(\sigma_{hh} + \sigma_{vv})}{2} \qquad (3.13)$$

$$P_s = \sigma_{hh} - P_d \qquad (3.14)$$

$$P_v = \frac{(\sigma_{vv} + \sigma_d)}{2} \qquad (3.15)$$

In general, P_S, P_d and P_v are used to generate RGB image bleu, red and green respectively. The coherency matrix is a 3x3 matrix that describes the statistical properties of the radar backscatter. It is calculated from the radar data using the following elements:

- $\sigma_{hh} = < (S_{hh})^2 >$ is the average of the square of the backscattered signal in the horizontal-horizontal (HH) polarization.

- $\sigma_{hv} = < S_{hh} * S_{hv} >$ is the average of the product of the backscattered signal in the horizontal-horizontal (HH) polarization with the backscattered signal in the horizontal-vertical (HV) polarization.

- $\sigma_{vv} = < (S_{vv})^2 >$ is the average of the square of the backscattered signal in the vertical-vertical (VV) polarization.

Where $<>$ denotes the averaging operation, S_{hh} is the complex backscattered signal in the HH polarization and S_{hv} is the complex backscattered signal in the HV polarization.

4. Huynen Decomposition

The Huynen Decomposition can also be used as an unsupervised classification. method for PolSAR data. In this approach, the decomposition is applied to the entire image, and the scattering components are used as features for clustering. The Huynen decomposition can be applied to the coherency matrix (S) of each pixel in a PolSAR image using the following steps:

- Calculate the degree of polarization A:

$$A = (\sigma_{hh} + \sigma_{vv})/(\sigma_{hh} + \sigma_{hv} + \sigma_{vv}) \qquad (3.16)$$

Where σ_{hh}, σ_{hv} and σ_{vv} are the elements of the coherency matrix S.

- Calculate the isotropic scattering matrix S_{iso}:

$$S_{iso} = [1,0,1]^T \qquad (3.17)$$

- Calculate the anisotropic scattering matrix S_{aniso} :

$$S_{aniso} = \sqrt{(1-A^2)}[\cos(2\varphi), \sin(2\varphi), \cos(2\theta)]^T \qquad (3.18)$$

Where φ and θ are the scattering phase angle and scattering angle respectively.

- Calculate the isotropic and anisotropic scattering components:

$$P_{iso} = A\sigma_{hh} + A\sigma_{vv} \qquad (3.19)$$

$$P_{aniso} = \sqrt{(1-A^2)}(\sigma_{hh}\cos(2\varphi) + \sigma_{hv}\sin(2\varphi) + \sigma_{vv}\cos(2\theta)) \qquad (3.20)$$

Assign the pixel to a cluster based on the values of P_{iso} and P_{aniso}.

Note that there are different ways to extract the scattering phase angle (φ) and scattering angle (θ) from the coherency matrix, and some methods may use different equations than the ones presented here. Also, the clustering step can be applied using different clustering algorithms, and it's usually done using software such as Matlab or ENVI.

IV. METHODS USED FOR POLSAR DATA CLASSIFICATION

A. Supervised Classification

Supervised classification is a method of classifying PolSAR data by referencing labeled instances. This method employs prior information about the surface types that appear in the image to develop a predictor to recognize and categorize the various surfaces in the image. The process of supervised classification typically involves the following steps:

- o Collect a set of labeled examples that are representative of the different surface types present in the image. These examples are used to train the classifier.

- o Extract features from the PolSAR data that are relevant to the classification task. These features may include backscattering power, scattering angles, and polarimetric parameters.

- o Train a classifier, such as a random forest or support vector machine, using the labeled examples and the extracted features.

- o Apply the trained classifier to the entire image to classify each pixel into one of the different surface types.

The process of categorizing PolSAR data through supervision involves the utilization of various methodologies. These methodologies are used to categorize image pixels based on previously labeled examples. Some frequently employed techniques include the Mahalanobis distance, minimum distance classifier, K-nearest neighbors, maximum probability classification, and linear and radial support vector machines. The result of this categorization process is a thematic map, which assigns a specific class label to each pixel. The success of the classification process is dependent on the quality and accuracy of the chosen training data; therefore, it's crucial to have an ample supply of high-quality training samples to establish the discriminatory rules between classes. (Guo, Wang, Gao, Shi, Zhang, Hou,2015).

B. Support Vector Machine (SVM)

Support Vector Machines (SVMs) are a type of machine learning algorithm designed to minimize structural errors and enhance the generalization capability of machine learning. They aim to minimize errors in predictions that occur due to the use of limited data during training, thereby producing accurate classification results even with limited labeled samples. The effectiveness of SVMs has led to the development of various other algorithms based on this technique, such as the system proposed by Maghsoudi et al., which incorporates feature selection using a non-parametric evaluation function and SVM. Additionally, SVM has been combined with radial basis kernel functions and stochastic distance methods to improve the reliability of region-based categorization.

Support Vector Machines (SVMs) are a widely used machine learning algorithm for supervised learning tasks, including both classification and regression. The objective of SVMs is to find the optimal boundary, also known as a hyperplane, that separates the data into different classes with maximum efficiency. This boundary is selected so as to maximize the margin, which is the distance between the boundary and the closest data points from each class. The essence of SVMs is to find a boundary that effectively divides the data into different classes. One of the significant advantages of SVMs is their ability to handle non-linearly separable data, meaning data that cannot be divided by a straight line. (W. Liu, J. Yang, P. Li, et al, 2018).
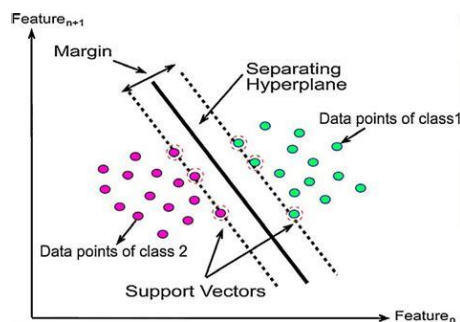
Figure 5. Linear support vector machine

SVMs can be adapted to handle non-linear boundaries through the use of the kernel trick. This technique transforms the data into a new, higher-dimensional space where it becomes linearly separable. In industry, SVMs are widely applied in areas such as image classification, text classification, and bioinformatics, due to their high precision and capability of processing high-dimensional data. However, the choice of kernel

function and selection of parameters can have a significant impact on the performance of SVMs, and large data sets can consume a considerable amount of computational resources.

The term "feature space" in the context of support vector machines (SVMs) refers to the dimensional space in which the data points are depicted for analysis. In a classic SVM setup, the feature space is the same as the original input space. Conversely, in a kernel SVM, the data is transformed into a higher-dimensional space, which becomes the feature space, before being used in the categorization process. The kernel function, represented as K (x, y), plays a crucial role in projecting the data points into the higher-dimensional feature space used in kernel SVM. (C. Lardeux, P.Louis Frison,2009). The kernel function must satisfy the Mercer condition, which ensures that the inner product between the transformed data points can be computed. This is critical in SVM, as the classification boundary is determined based on the inner product of the data points in the feature space. Three common types of kernel functions that can be used in SVM are:

- Sigmoid: $K(x_i, x) = \tan(\gamma x_j^T . x_i + p)$ (4.1)

- Polynomial: $K(x_i, x) = \tan((\gamma x_j^T . x_i + p)^r$ with p>0 (4.2)

- Radial basis function: $K(x_i, x) = exp^{-\frac{(x_j - x_i)}{2\sigma^2}}$ (4.3)

Each of these kernel functions is defined by its own parameters, like p, that must be set before training.

The application of a kernel function in SVM enables the classification to be performed by finding the hyperplane in a high-dimensional feature space instead of the original input space. This approach can be particularly advantageous when the data cannot be separated linearly in the original space. The kernel function transforms the input data into a new feature space where the data points can be split using a linear boundary.

C. Mahalanobis Distance

The Mahalanobis distance is a numerical representation of the difference between a point and a statistical distribution. It calculates the distance from a given point to the mean of the dataset, taking into consideration the covariance of the data.

The formula for the Mahalanobis distance between a point x and a distribution with a mean vector μ and covariance matrix S is:

$$D(x) = \sqrt{(x - \mu)^T \, S^{-1}(x - \mu)} \qquad (4.4)$$

Where x is a column vector representing the point, μ is the mean vector of the distribution, S is the covariance matrix and T denotes transpose.

The Mahalanobis distance is a way of measuring the distance between a point and a distribution. It takes into account the covariance of the data and allows you to compute the distance while taking into account the correlation between variables. The Mahalanobis distance can be used in various fields, such as machine learning, image processing, and pattern recognition. It is particularly useful when the data has a non-uniform distribution or when the variables are correlated. In multivariate statistical process control, it is used to identify patterns in multivariate data that are not easily noticeable in univariate data. In the application of Synthetic Aperture Radar (SAR) data, the Mahalanobis distance is used to determine the error in matching two classes (M1 and M2) with the input feature vector (v). The minimum-error classifier finds the error between the feature vector and the mean of each class and selects the one with the minimum error. as illustrated in figure 6.
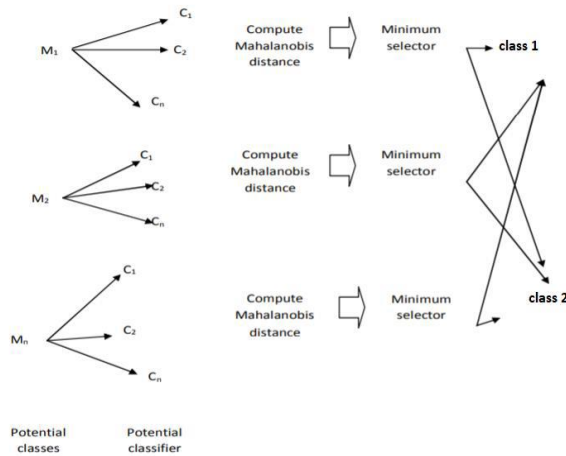
Figure 6. Multiple classes using the Mahalanobis distance.

D. Convolutional Neural Network (CNN)

A convolutional neural network (CNN) can be utilized for the task of classifying images obtained from polarimetric synthetic aperture radar (PolSAR)

systems. This type of deep learning model is specifically designed to analyze and recognize patterns in image data using a mathematical operation called convolution. The CNN typically consists of multiple layers, including:

- ✓ Convolutional layers: A convolution operation is performed on the input data using these layers, typically using a set of learnable filters. The convolution operation extracts features from the input data, and the filters are adjusted during training to optimize the extraction of relevant features. The convolutional layer applies the following equation:

$$Output = Convolution\ (Input, Weights) + Bias$$

- ✓ Pooling layers: These layers simplify the data by aggregating values in small areas of the input data, either by selecting the largest or computing the mean of these values. Pooling layers play a role in reducing processing requirements and making the data insensitive to small changes in position. The pooling layer applies the following equation:

$$Output = Pooling\_function(Input)$$

- ✓ Fully connected layers: These layers are a type of neural network layer that makes a connection between each neuron in the current layer and every neuron in the next layer, much like a traditional multi-layer perceptron (MLP). The purpose of these layers is to make final decisions based on the features that have been extracted through the convolutional and pooling layers. The fully connected layer applies the following equation:

$$Output = w_x + b$$

- ✓ Activation function: These are used to introduce non-linearity in the output of each layer. Commonly used activation functions are ReLU, sigmoid, tanh, and softmax. The activation function applies the following equation:

$$Output = Activation\ (Input)$$

- ✓ Loss function: The purpose of the loss function is to evaluate the performance of the model and determine the error between the

predicted values and the actual values. The loss function quantifies this error and guides the optimization process in adjusting the model's parameters. The most frequently used loss functions are mean squared error and cross-entropy.

✓ Optimizer: The optimizer updates the parameters of the network to reduce the value of the loss function. Popular optimization algorithms used in neural networks include Stochastic Gradient Descent (SGD), Adam, and RMSprop. These algorithms determine the direction in which the weights and biases should be adjusted in order to minimize the difference between the predicted output and the actual output.

The CNN is trained using a large dataset of labeled PolSAR images; during the training process, the weights of the filters in the convolutional layers are adjusted to reduce the loss. After the training process, the CNN can be used to classify new PolSAR images by applying the trained filters to extract features and making decisions based on the final output of the network. (Wang,He, Liu, et al.2018).
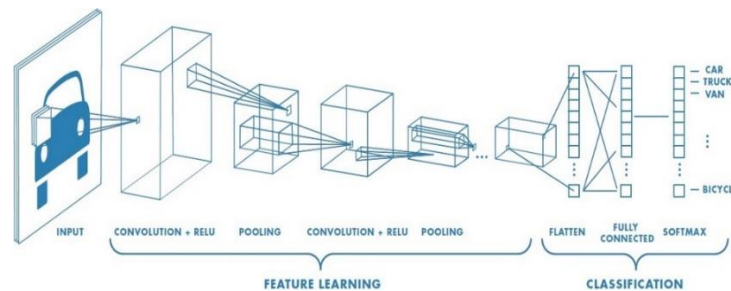
Figure 7. Understanding of Convolutional Neural Network (CNN)

The ability of convolutional neural networks (CNNs) to routinely extract hierarchical features and get end-to-end classification is a key factor in their success in classifying images obtained from polarimetric synthetic aperture radar (PolSAR) systems. This is because CNNs can extract features at different levels of abstraction and perform the classification task all in one step, as opposed to traditional methods that rely on manual feature extraction. This makes CNNs more efficient and effective at image classification tasks, especially when it comes to PolSAR images. (Ronny Hänsch, O.H.2017). It's valuable to consider that adding more convolutional layers to a CNN can enhance its accuracy and ability to generalize, but also necessitates a larger training dataset for convergence to occur. In the context of PolSAR image

classification, shallow CNN models, with fewer than five convolutional layers, may outperform deep CNN models even though they have lower accuracy on average. This is due to the limited amount of data available in PolSAR benchmarks.

1. Complex-Valued Convolutional Neural Network (CV-CNN)

A CV-CNN can be utilized to identify and categorize objects based on the polarimetric characteristics of the radar signals they reflect, as seen in PolSAR data. This data provides information about the polarization state of objects and can be useful in separating various types of terrain, such as urban regions, water bodies, and forest areas. A CV-CNN can be trained to recognize patterns in the PolSAR data and then classify new data based on those patterns.

A CV-CNN for PolSAR data classification works by training the network on a set of labeled PolSAR data. The labeled data is used to teach the network to recognize patterns in the data that are associated with different classes of land cover. The training process involves inputting the PolSAR data into the network, which is composed of multiple layers of artificial neurons. Each layer applies a set of mathematical operations to the data called convolution and pooling, which extract features from the data that are relevant to the classification task. (Cao, Wu, Zhang et al, 2019).

The features extracted by the convolutional layers are then fed into fully connected layers, where they are processed and combined to determine the final classification of the input data. During training, the parameters of both the convolutional and fully connected layers are adjusted to optimize the accuracy of the network's decisions. After training, this network can be used to classify new, unlabeled PolSAR data by providing it as input and using the learned patterns to make predictions about its class. To measure the performance of the network, complex-valued is used to compare the network's predictions with the actual class labels of the data.

The performance of a CV CNN can be determined through cross-validation by dividing the PolSAR dataset into training and testing portions. The network learns patterns from the training set, and its ability to classify unseen data is evaluated on the testing set. This process is repeated several times with various partition combinations to obtain a more precise evaluation of the network's overall performance. (Zhang, Wang, Xu, Ya-Qiu Jin,2017).
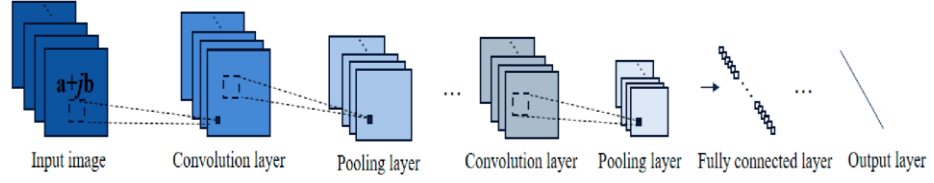
Figure 8. Framework of the CV-CNN model.

❖ Convolution operation:

$$O(I + 1)m = \varepsilon(I + 1)m = \sum p = 1P\varepsilon(I)p * \omega(I + 1)p +$$
$$\psi(I + 1)m = \sum p = 1P\Psi(I)p * \omega(I + 1)p \qquad (4.5)$$

The equation represents how the output feature map of a convolutional neural network (CNN) is calculated for a specific layer (l+1) and a specific position (m). This is done by taking the convolution of the input feature map of the previous layer (l) at position k with a set of filter banks (w) for the current layer (l+1) at position p and kernel K. The result of this convolution operation is then summed up over all possible positions (k) and kernel sizes (K) to produce a single value. This value is then added to a bias term (b) for the current layer (l+1) at position p. The real and imaginary parts (ξ and ψ) of the complex-valued domain are considered in the calculation. The result is a weighted sum of the previous layer's output feature maps (ψ(l)p).

❖ Nonlinear activation function: f(·) is the nonlinear activation function, usually represented as the sigmoid function.

❖ Pooling layer: Max pooling and average pooling are two popular methods for reducing the spatial dimensions of the input feature maps in a pooling layer. The goal of pooling is to reduce computational cost and improve the network's invariance to small translations of the input data by simplifying the spatial structure of the feature maps and merging similar features into a single representation.

❖ Fully connected layer: In a fully connected layer, every neuron in the current layer is connected to the input features from the previous layer. The output of this layer is calculated using a combination of the weight

matrix (W(L)), bias term (b(L)), and activation function (σ(·)). The activation function is typically represented by the softmax function for classification tasks, and the final output (O(L)) is the result of applying this activation function to the sum of the weighted input features and bias term. (Xie, Ma, Zhao, Liu, 2020).

2. Configuration of (CV-CNN)

The design of the CV-CNN model is shown in Figure 9 and is comprised of several key components, including input and output layers as well as two convolutional layers, one pooling layer, and one fully connected layer. The input layer has a size of 12x12x6, which means that each patch of input data is 12x12 in size and has 6 channels. The convolutional and pooling layers reduce the size of the feature maps, but if the size of the input data is smaller than 12x12, zero padding is used to maintain the desired depth for the network. The fully connected layer connects every neuron in one layer to every neuron in the next layer.

The input layer has a size of 12x12x6, meaning each local patch has dimensions of 12x12 and 6 channels. The first convolutional layer consists of six filters with a size of 3x3x6 and a stride of 1. This results in six feature maps of size 10x10. The average pooling layer, with a pooling size of 2x2 and a stride of 1, is then applied, reducing the feature map size to 5x5. The second convolutional layer has a filter size of 3x3x6x12 and creates 12 feature maps with a size of 3x3. (Zhang, Wang, Xu, Ya-Qiu Jin,2017).

These 3-D feature maps are then reshaped into a 1-D vector with 108 neurons to serve as the fully connected layer. Finally, the output layer with c neurons is applied, where c is the number of classification classes. In the Flevoland dataset, c is equal to 15.
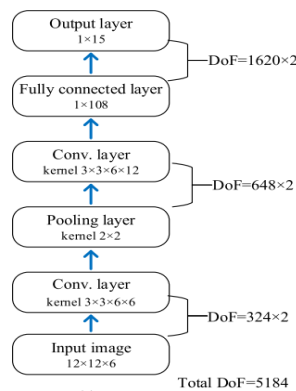
Figure 9. Overall architecture of (a) CV-CNN

E. Machine and Deep Learning techniques for POLSAR classifications

Traditional machine learning techniques like Support Vector Machaines (SVM) and Mahalanobis distance can be used for classification of land cover using POLSAR data. Here are some advantages and disadvantages of using these techniques compared to deep learning techniques like Convolutional Neural Networks (CNN) and Complex-Valued CNN (CV-CNN):

❖ Advantages of using SVM and Mahalanobis distance for POLSAR data classification:

1. Computationally efficient: SVM and Mahalanobis distance are computationally efficient compared to deep learning techniques like CNN and CV-CNN. This is particularly advantageous when dealing with large datasets, which can take a long time to train using deep learning methods.

2. Interpretable: SVM and Mahalanobis distance are simple models that can be easily interpreted. The parameters and decision boundaries of these models can be easily visualized, which can help in understanding the classification results.

3. Robust to noise: SVM and Mahalanobis distance are robust to noise and outliers in the data. This is particularly advantageous for POLSAR data, which can be affected by speckle noise.

❖ Disadvantages of using SVM and Mahalanobis distance for POLSAR data classification:

1. Limited capacity: SVM and Mahalanobis distance have limited capacity to learn complex features from the data. This can result in lower classification accuracy compared to deep learning techniques.

2. Hand-crafted features: SVM and Mahalanobis distance require hand-crafted features, which can be time-consuming and require domain expertise. This is in contrast to deep learning techniques, which can automatically learn features from the data.

3. Sensitivity to hyperparameters: SVM and Mahalanobis distance are sensitive to the choice of hyperparameters, such as the kernel function and distance metric. This can make the model selection process challenging and require cross-validation.

❖ Advantages of using CNN and CV-CNN for POLSAR data classification:

1. Automatic feature learning: CNN and CV-CNN can automatically learn features from the data, which can reduce the need for hand-crafted features and domain expertise.

2. High capacity: CNN and CV-CNN have high capacity to learn complex features from the data, which can result in higher classification accuracy compared to traditional machine learning techniques.

3. End-to-end learning: CNN and CV-CNN can learn end-to-end mappings from the input data to the output labels, which can result in better generalization and transferability of the learned features.

❖ Disadvantages of using CNN and CV-CNN for POLSAR data classification:

1. Computationally intensive: CNN and CV-CNN are computationally intensive and require high-end hardware, such as GPUs, to train and run.

2. Black-box models: CNN and CV-CNN are complex models that can be difficult to interpret. The learned features and decision boundaries are not easily visualized, which can make it challenging to understand the classification results.

3. Sensitivity to hyperparameters: CNN and CV-CNN are sensitive to the choice of hyperparameters, such as the number of layers, filter sizes, and learning rates. This can make the model selection process challenging and require cross-validation.

The choice of classification method for POLSAR data depends on several factors, such as the complexity of the data, the size of the dataset, and the available computational resources. In general, deep learning techniques like CNN and CV-CNN have been shown to achieve higher classification accuracy compared to traditional machine learning techniques like SVM and Mahalanobis distance. (Rui, Hai, Zongjie,

Zongyong.2019). This is because deep learning methods can automatically learn complex features from the data, which can capture the underlying patterns in the data more effectively.

However, deep learning techniques are computationally intensive and require high-end hardware, such as GPUs, to train and run. In addition, deep learning models are more complex and harder to interpret compared to traditional machine learning models. Therefore, the choice between deep learning and traditional machine learning methods for POLSAR classification depends on the trade-off between accuracy and computational efficiency as well as the interpretability of the model. If high accuracy is the primary goal and computational resources are available, then deep learning techniques like CNN and CV-CNN may be preferred. However, if interpretability and simplicity are important and computational resources are limited, then traditional machine learning techniques like SVM and Mahalanobis distance may be more appropriate.

In the experiment chapter, we will examine how each of these algorithms performs for the Flevoland dataset with 15 categories. We will evaluate which algorithm gives the highest overall accuracy and how it performs for each different category with varying numbers of samples.

V. DESIGN EXPERIMENTS

A. AIRSAR Flevoland Dataset

The proposed algorithm will be tested using the NASA/JPL AirSAR L-band PolSAR dataset. This dataset was acquired in 1989 from Flevoland in the Netherlands and is composed of a 4-look Pauli-RGB image (Figure 10) that depicts 15 different land cover categories, such as water, barley, peas, stem beans, beet, forest, bare soil, grasses, rapeseed, lucerne, wheat A, wheat B, buildings, potato, and wheat C.

The image is represented using the Pauli matrix representation, which differentiates between different crops. In addition, a ground truth map (Figure 11) that uses color coding to indicate the different categories is also provided. L-band polarimetric SAR technology is particularly useful for identifying and classifying different crops in agricultural fields.

Figure 10. Flevoland Pauli-RGB image

Figure 11. Flevoland ground truth map

The AirSAR L-band PolSAR image used in this study was acquired by NASA/JPL and has a resolution of 750x1024 pixels. However, it is crucial to carefully choose the training sets from the ground truth maps, as these maps may not have enough detail to accurately evaluate the classification performance. The chosen pixels from the training sets are then used for mahalanobis distance and svm supervised classification procedures. The classification accuracy is assessed by selecting enough pixels from each training set as a reference class-map, to generate statistically reliable results.

Table 3. Ground truth classes with their samples number on Flevoland data

Number of classes	Classes	Ground Truth
1	Water	30732
2	Forest	17872
3	Lucerne	10992
4	Grass	10198
5	Rapeseed	21293
6	Beet	14628
7	Potatoes	19459
8	Peas	11548
9	Stem beans	8565
10	Bare soil	6020
11	Wheat A	18494
12	Wheat B	11480
13	Wheat C	23621
14	Barely	7901
15	Building	798

B. Supervised Mahalanobis Distance

This is a process of preparing the data for a supervised machine learning algorithm. In this study, the data is the Flevoland ground truth. The first step is to apply the Mahalanobis distance classification to select different samples from the ground truth. Then, different sets of testing samples are selected for each class. After that, the ML distance in MATLAB is applied to find the training dataset with the highest accuracy. The process results in 192247 testing samples and 21354 training samples, with an accuracy result of 63.80% and 63.78%, respectively, after applying ML distance for the L-band. The results suggest a high proportion of bare soil and the images can differentiate between classes, as indicated by the blue color in the image indicating surface scattering.
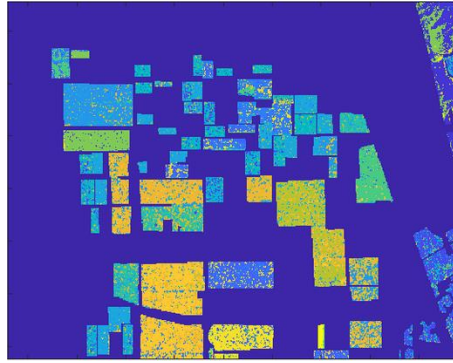
Figure 12. Results obtained by applying Supervised Mahalanobis Distance to the Flevoland data set

1. Confusion Matrix

The values in the confusion matrix are used to calculate various performance metrics such as accuracy, precision, recall, and F1 score, which help to understand the effectiveness of the classifier. The true positives (TP) are the number of instances that are correctly classified as positive. False positives (FP) are instances that are incorrectly classified as positive when they are actually negative. True negatives (TN) are instances that are correctly classified as negatives, while false negatives (FN) are instances that are incorrectly classified as negatives when they are actually positives. True Positives: Positive instances that were correctly classified.

- ❖ False Positives: Incorrectly classified positive instances.

- ❖ True Negatives: Correctly classified negative instances.

- ❖ False Negatives: Incorrectly classified negative instances.

Accuracy is defined as the total number of correct predictions divided by the total number of predictions made by the model. Precision is the ratio of true positive predictions to the total number of positive predictions made by the model. Recall, also known as sensitivity or the "true positive rate," is the ratio of true positive predictions to the total number of actual positive instances in the data. The F1-score is the harmonic mean of precision and recall and provides a single metric that balances both metrics. Specificity is the ratio of true negative predictions to the total number of actual negative instances in the data. These metrics provide valuable information about the performance of a classification model and help identify areas for improvement.

Figure 13. Confusion Matrix for Binary Datasets.

The four parameters calculated to compare the performance of each model are:

- Accuracy: Represents the amount of tests accurately classified by the classifier.

$$\text{Accuracy} = \frac{(\text{True Positives} + \text{True Negatives})}{(\text{True Positives} + \text{True Negatives} + \text{False Positives} + \text{False Negatives})}$$

- Precision: Refers to the accuracy of the classifier, specifically the rate at which the samples are correctly identified as positive among all the samples classified as positive in the test set.

$$\text{Precision} = \frac{\text{True Positives}}{(\text{True Positives} + \text{False Positives})}$$

- Recall: Reflects the integrity of the model, also known as the detection rate or sensitivity of the model. It measures the successful labeling of positive samples in the test set.

$$\text{Recall} = \frac{\text{True Positives}}{(\text{True Positives} + \text{False Negatives})}$$

- F1-Score: Refers to the harmonic mean of Precision and Recall, which provides a balanced evaluation of both the false positive and false negative rates.

$$\text{F1}_{\text{Score}} = 2 * \frac{(\text{Precision} * \text{Recall})}{(\text{Precision} + \text{Recall})}$$

2. **Mahalanobis Training Confusion Matrix**

To create a confusion matrix, it is necessary to combine both the actual class labels and the predictions made by the classifier. The actual class labels are placed in columns, while the predictions made by the classifier are placed in rows. This forms a matrix with correct classifications along the diagonal from the top left to the bottom right. The number of samples used for training the model is 21354, with correct

classifications represented by the values along the diagonal of the matrix. The accuracy of the model is measured by comparing the actual class labels to the predictions made by the model, which is found to be 63.79%.

Table 4. Mahalanobis Training Confusion matrix for flevolend.

Mahalanobis Training Confusion matrix for flevolent

		1 Water	2 Forest	3 Lucerne	4 Grass	5 Rapeseed	6 Beet	7 Potatoes	8 Peas	9 Stem-Beans	10 Bare-Soil	11 Wheat-A	12 Wheat-B	13 Wheat-C	14 Borely	15 Building
1	Water	1850	5	15	9	42	2	1	74	2	862	5	15	39	152	0
2	Forest	0	1148	16	11	0	88	275	12	104	0	101	0	16	16	0
3	Lucerne	0	0	659	213	5	6	0	1	2	0	0	2	33	178	0
4	Grass	0	5	146	586	14	60	4	11	14	4	7	5	48	115	0
5	Rapeseed	1	2	1	7	1267	150	0	133	4	4	95	344	117	4	0
6	Beet	1	16	6	62	54	1095	48	93	16	3	42	0	15	11	0
7	Potatoes	0	285	7	6	0	285	1029	188	38	0	92	0	10	5	0
8	Peas	0	8	0	1	96	88	16	703	1	0	81	144	13	3	0
9	Stem-Beans	0	23	28	69	0	43	11	19	619	0	5	0	38	1	0
10	Bare-Soil	25	0	1	8	0	0	0	0	0	549	0	0	4	15	0
11	Wheat-A	0	1	2	5	61	69	0	62	6	0	1031	426	185	1	0
12	Wheat-B	0	0	1	6	143	24	0	15	0	0	53	759	147	0	0
13	Wheat-C	0	0	40	120	50	21	0	2	0	3	62	345	1650	69	0
14	Borely	1	0	62	61	6	0	0	0	1	21	0	0	13	625	0
15	Building	0	1	0	2	0	0	1	0	19	0	0	0	2	3	51

The Kappa coefficient is also used to assess the agreement between the actual class labels and the predictions made by the model, with the result of the Kappa value being 0.62, where a value of 1 indicates a perfect agreement and a value of 0 indicates no agreement.

Table 5. Trained samples and the accuracy of the resulted prediction compared with the true data.

	Classes	Training Samples	Percentage
1	Water	3073	60.20
2	Forest	1787	64.24
3	Lucerne	1099	59.96
4	Grass	1019	57.51
5	Rapeseed	2129	59.51
6	Beet	1462	74.90
7	Potatoes	1945	52.90
8	Peas	1154	60.92
9	Stem beans	856	72.31
10	Bare soil	602	91.20
11	Wheat A	1849	55.76
12	Wheat B	1148	66.11
13	Wheat C	2362	69.86
14	Barely	790	79.11
15	Building	79	64.56

3. Mahalanobis Testing Confusion Matrix

A confusion matrix is used during the testing phase to assess the accuracy of the machine learning model. The matrix compares the actual classifications with the predictions made by the model. The true classifications are listed in columns, while the predictions are in rows, and the correct classifications are shown along the diagonal. In the testing phase, 192,247 values were used, and an overall accuracy of 63.80% was achieved by comparing the true data with the test results.

An average accuracy of 66.26% was also obtained. The Kappa value measures the agreement between the actual classifications and the predictions made by the model. In this case, the Kappa value was approximately 0.62, which indicates a relatively good agreement between the actual and predicted classifications.

Table 6 Mahalanobis Testing Confusion matrix for Flevoland

		1 Water	2 Forest	3 Lucerne	4 Grass	5 Rapeseed	6 Beet	7 Potatoes	8 Peas	9 Stem-Beans	10 Bare-Soil	11 Wheat-A	12 Wheat-B	13 Wheat-C	14 Borely	15 Building
1	Water	16703	34	197	56	351	17	12	568	37	7832	37	162	297	1356	0
2	Forest	2	10242	117	135	5	839	2596	153	910	2	783	5	117	179	0
3	Lucerne	0	1	5957	2058	47	37	0	8	17	1	3	21	370	1373	0
4	Grass	1	32	1451	5277	114	463	74	137	129	86	35	36	360	984	0
5	Rapeseed	19	28	6	52	11449	1350	0	1127	37	64	906	2982	1058	87	0
6	Beet	0	166	62	465	518	####	396	630	159	15	306	11	123	90	0
7	Potatoes	0	2567	55	102	34	2669	9157	1565	265	0	952	17	114	17	0
8	Peas	2	48	0	8	867	1004	198	6092	10	0	870	1202	84	9	0
9	Stem-Beans	0	140	291	578	10	336	125	158	5661	0	56	0	348	6	0
10	Bare- Soil	257	0	16	63	20	1	0	0	0	4985	0	4	17	55	0
11	Wheat-A	0	17	33	50	631	683	10	755	19	0	9190	3769	1483	5	0
12	Wheat-B	1	0	7	87	1163	202	0	165	2	4	518	6992	1187	4	0
13	Wheat-C	3	9	397	1107	505	187	2	10	4	23	640	3121	14692	559	0
14	Borely	10	0	663	528	17	2	0	0	17	167	0	0	169	5538	0
15	Building	0	20	4	11	2	4	3	3	144	0	5	1	19	10	493

Table 7 . Tested samples and the accuracy of the resulted prediction compared with the true data.

	Classes	Testing Samples	Percentage
1	Water	27659	60.39
2	Forest	16085	63.67
3	Lucerne	9893	60.21
4	Grass	9179	57.49
5	Rapeseed	19164	59.74
6	Beet	13166	77.66
7	Potatoes	17514	52.28
8	Peas	10394	58.61
9	Stem beans	7709	73.43
10	Bare soil	5418	92.01
11	Wheat A	16645	55.21
12	Wheat B	10332	67.67
13	Wheat C	21259	69.11
14	Barely	7111	77.88
15	Building	719	68.57

C. Support Vector Machine Classification

To evaluate the performance of the SVM classifier on the Flevoland dataset, the model was tested using a portion of the data. The training portion consisted of 21354 samples, which were selected from the actual image after removing 10% of the data. The testing portion, on the other hand, consisted of 192247 samples that were not used in the training process. The results showed that the SVM classifier had an accuracy of 79.06 % on the training data and 77.96 % on the testing data after being applied to the L-band.

The results of the predictions were represented in a confusion matrix for both the training and testing data, and the resulting image of the predictions was visualized using the imagesc function. Using the same imagesc function, the ground truth image was also read and visualized.
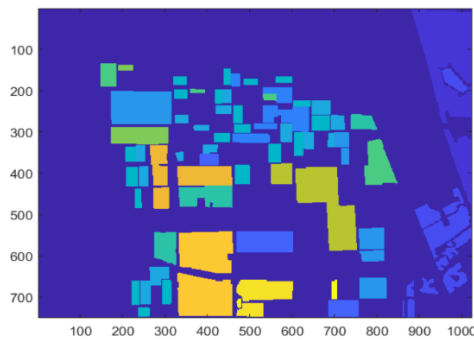
Figure 14. Ground Truth for Flevoland data set

1. SVM Training Confusion Matrix

To build the confusion matrix, it's necessary to gather both the actual classifications and the classifier's outputs in a single chart (Table 8). In this chart, the actual classifications are arranged in columns and the classifier's predictions in rows. There are 21354 values utilized for training, with the diagonal values from upper-left to lower-right in the confusion matrix indicating the values that have been accurately classified. The comparison between the actual data and the results of the trained data results in an overall accuracy of 79.06 percent, with an average accuracy of 75.51%. The Kappa coefficient, which measures the correlation between classifications and actual values, has a value of 0.77 in the training data.

Table 8 illustrates the actual classification of each category in the Flevoland dataset along with the number of training samples used. The accuracy of the SVM

classifier's predictions is assessed by comparing these results to the actual classifications.

Table 8. SVM Training Confusion matrix for Flevoland with Trained samples and the accuracy.

		Water	Forest	Lucerne	Grass	Rapeseed	Beet	Potatoes	Peas	Stem-Beans	Bare-Soil	Wheat-A	Wheat-B	Wheat-C	Borely	Building	Σ	%
1	Water	16703	34	197	56	351	17	12	568	37	7832	37	162	297	1356	0	27659	60.39
2	Forest	2	10242	117	135	5	839	2596	153	910	2	783	5	117	179	0	16085	63.67
3	Lucerne	0	1	5957	2058	47	37	0	8	17	1	3	21	370	1373	0	9893	60.21
4	Grass	1	32	1451	5277	114	463	74	137	129	86	35	36	360	984	0	9179	57.49
5	Rapeseed	19	28	6	52	11448	1350	0	1127	37	64	906	2982	1058	87	0	19164	59.74
6	Beet	0	166	62	465	518	####	396	630	159	15	306	11	123	90	0	13166	77.66
7	Potatoes	0	2567	55	102	34	2669	9157	1565	265	0	952	17	114	17	0	17514	52.28
8	Peas	2	48	0	8	867	1004	198	6092	10	0	870	1202	84	9	0	10394	58.61
9	Stem-Beans	0	140	291	578	10	336	125	158	5661	0	56	0	348	6	0	7709	73.43
10	Bare-Soil	257	0	16	63	20	1	0	0	0	4985	0	4	17	55	0	5418	92.01
11	Wheat-A	0	17	33	50	631	683	10	755	19	0	9190	3769	1483	5	0	16645	55.21
12	Wheat-B	1	0	7	87	1163	202	0	165	2	4	518	6992	1187	4	0	10332	67.67
13	Wheat-C	3	9	397	1107	505	187	2	10	4	23	640	3121	14692	559	0	21259	69.11
14	Borely	10	0	663	528	17	2	0	0	17	167	0	0	169	5538	0	7111	77.88
15	Building	0	20	4	11	2	4	3	3	144	0	5	1	19	10	493	719	68.57

2. SVM Testing Confusion Matrix

The performance of the SVM classifier on the Flevoland dataset was evaluated through the use of Confusion Matrices for both the training and testing data. The testing matrix compared the actual classifications with the classifier's predictions, with a total of 192,247 samples analyzed. The accuracy of the predictions was found to be 77.96%, with an average accuracy of 74.71%. These findings indicate that the Kappa value for the testing data is approximately 0.76.

The results presented in Table 9 depict the actual classifications and the number of tested samples in the Flevoland data, indicating the accuracy of the classifier's predictions when compared to the actual data.

Table 9 . SVM Testing Confusion Matrix for Flevoland and the accuracy of the resulted prediction compared with the true data.

|		Water	Forest	Lucerne	Grass	Rapeseed	Beet	Potatoes	Peas	Stem-Beans	Bare-Soil	Wheat-A	Wheat-B	Wheat-C	Borely	Building	Σ	%
1	Water	26141	33	135	52	45	19	19	11	12	437	16	16	154	569	0	27659	94.51
2	Forest	15	12898	120	29	4	368	2094	7	455	0	39	2	25	29	0	16085	80.19
3	Lucerne	10	1	8479	562	69	9	0	0	5	2	5	36	493	222	0	9893	85.71
4	Grass	8	61	1745	4904	284	296	82	60	205	38	55	44	810	582	5	9179	53.43
5	Rapeseed	60	8	11	67	15103	305	0	244	22	6	855	1124	1348	0	0	19164	78.81
6	Beet	1	528	42	78	496	####	628	418	379	6	306	7	80	22	11	13166	77.20
7	Potatoes	0	3191	45	31	28	740	12894	234	221	0	97	0	24	9	0	17514	73.62
8	Peas	30	35	2	11	554	833	235	7881	27	0	722	25	37	2	0	10394	75.82
9	Stem-Beans	0	468	179	54	26	369	221	102	6042	0	114	6	128	0	0	7709	78.38
10	Bare-Soil	1464	0	15	66	17	0	0	0	0	3746	0	6	42	62	0	5418	69.14
11	Wheat-A	5	63	23	28	1406	618	7	309	157	0	12634	1055	340	0	0	16645	75.90
12	Wheat-B	7	1	1	84	2807	49	0	3	22	0	717	5386	1255	0	0	10332	52.13
13	Wheat-C	18	16	637	486	998	73	2	0	7	9	239	567	18083	124	0	21259	85.06
14	Borely	44	0	905	586	35	0	0	0	11	29	0	0	479	5022	0	7111	70.62
15	Building	3	35	6	6	0	2	11	2	119	0	5	0	18	8	504	719	70.10
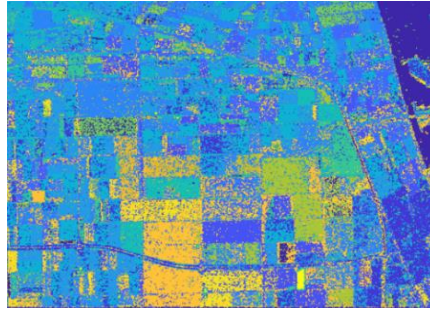
Figure 15. Result-image of the predictions made by the SVM model.

This is a representation of the results of the sample classification shown in figure 15. The "result_image" variable holds the class labels assigned to each sample by the SVM classifier. The "imagesc" function is used to display the "result_image" as a grayscale image, where each pixel's shade corresponds to a specific class label. The "imagesc" function scales the range of "result_image" so that the minimum value corresponds to the first color in the chosen colormap, and the maximum value corresponds to the last color in the colormap.
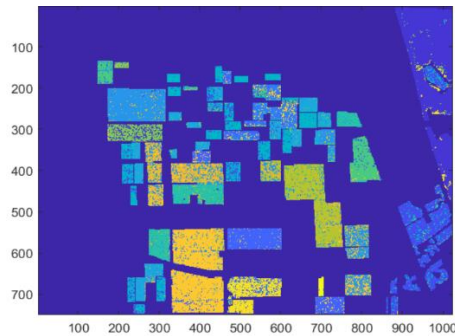
Figure 16 . Result-image of SVM classifier after applying mask.

D. Complex-Valued Convolutional Neural Network Classification

The study was conducted on the Flevoland dataset, a subset of a complete POLSAR image captured by the NASA/Jet Propulsion Laboratory AIRSAR platform in 1989. The Flevoland dataset, which represents an agricultural area in the Netherlands, is a frequently used benchmark for POLSAR data classification research. Figure 17(a) depicts an RGB image created using the intensities with the Pauli decomposition, which has a resolution of 1024x750 pixels. The ground truth, shown in Figure 17(b), identifies 15 different classes, including stembeans, peas, forest, lucerne, three types of wheat, beet, potatoes, bare soil, grass, rapeseed, barley, water, and some buildings. The legend for the ground truth is displayed in Figure 17(c).

47

For small regions like buildings, a zero-padding strategy is employed. For instance, a 12x12 window centered at a building pixel near the edge will have its pixels outside the region filled with zeros. (Zhang, Wang, Xu,and Ya-Qiu, 2017).

(a)   (b)

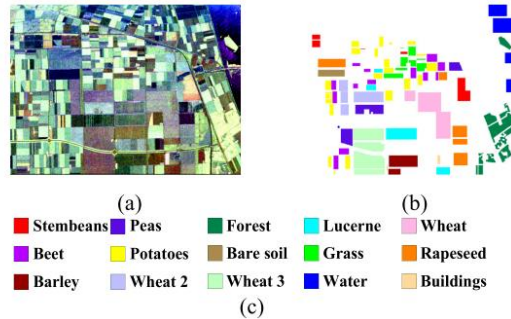|	Stembeans		Peas		Forest		Lucerne		Wheat
	Beet		Potatoes		Bare soil		Grass		Rapeseed
	Barley		Wheat 2		Wheat 3		Water		Buildings

(c)

Figure 17. Flevoland data set. (a) Pauli RGB of POLSAR. (b) Ground truth of (a). (c) Legend of the ground truth

.

A sensitivity analysis was performed to assess the impact of the sampling rate and determine the optimal rate. This was achieved by varying the sampling rate and the results are displayed in Figure 18.
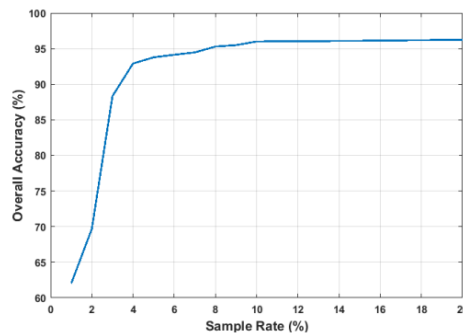
Figure 18. Flevoland OA given a different sample rate.

As depicted in the figure, the overall accuracy (OA) is around 62% when the sampling rate is set at 1%, however, it rapidly increases to 93% as the rate approaches 4%. This highlights the usefulness and performance of the CV-CNN. The accuracy remains stable at 96% when the rate exceeds 10%. Therefore, a sampling rate of 10% is deemed sufficient for the Flevoland dataset. During the experiment, the samples were split into two portions, with 9% utilized for training and 1% for validation. All pixels that have a corresponding ground-truth map were used as testing data.

As can be observed, the accuracy of classification improves as the training sample rate increases, but eventually reaches a level of saturation. Our experience shows that this

saturation level is specific to the dataset, task, and network architecture. If the distribution of the same class is homogeneous and there is a significant difference between classes, even a low training sample rate can yield good results.

For classification problems with a small number of classes, a lower sample rate is usually sufficient. However, a more complex network architecture requires more training data. The topic of sampling rate in SAR image classification has been explored in several studies. Hou et al. selected approximately 5% of the training samples per class, Guo et al. utilized a 10% training sample rate, and Jiao and Fang employed a 5% training sample rate. Based on these findings, we recommend a general range of 5% to 10% for the sample rate in real-world applications. This comparison appears that this method has good performance in classifying pixels, especially for the lucerne, water, and bare soil classes. The results are in good agreement with the ground truth, as can be seen from the comparison between Figures 17(b) and 19(b).
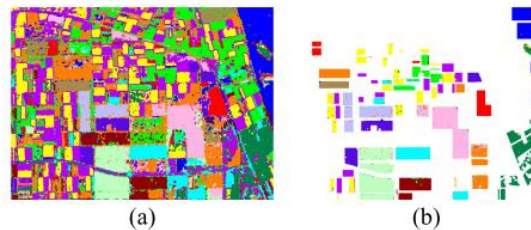
Figure 19. Classification results of the proposed algorithm on the first data set. (a) Result of whole map classification. (b) Result overlaid with the ground-truth map.

Table 10 displays the results of the CV-CNN's classification accuracy on each individual class in the ground truth. The CV-CNN's overall accuracy (OA) is recorded at 96.2%. Table 11 presents the comparison between the classes of the CV-CNN model, in the form of a confusion matrix. By transitioning the CNN from real to complex, the CV-CNN achieved an improvement of 1.2 in terms of the classification error rate.

Table 10 . *Classification accuracy of the whole ground truth (%).*

Class	CV-CNN
Stem beans	98.8
Peas	98.7
Forest	96.8
Lucerne	98.1
Wheat	95.0
Beet	97.6
Potatoes	96.7
Bare soil	98.8
Grass	90.0
Rapeseed	92.0
Barley	94.5
Wheat2	94.2
Wheat3	96.6
Water	99.4
Buildings	83.2
Overall Accuracy	96.2
Overall Error	3.8
OA with Postprocessing	97.7

Table 11. Confusion Matrix of the whole ground truth (%) for CV-CNN.

		1 Water	2 Forest	3 Lucerne	4 Grass	5 Rapeseed	6 Beet	7 Potatoes	8 Peas	9 Stem-Beans	10 Bare-Soil	11 Wheat-A	12 Wheat-B	13 Wheat-C	14 Borely	15 Building
1	Water	99.4	0	0	0	0	0	0	0	0	0.2	0	0	0	0.4	0
2	Forest	0	96.8	0	0	0	0.2	2.8	0	0.2	0	0	0	0	0	0
3	Lucerne	0	0	98.1	1.2	0	0.3	0	0	0.1	0	0	0	0	0.3	0
4	Grass	0	0	7.4	90	2	0.6	0	0	0	0	0	0	0	0	0
5	Rapeseed	0	0	0	0.3	92	0.2	0	0.2	0	0	3.6	3	0.7	0	0
6	Beet	0	0	0	0.2	0	97.6	1.6	0.6	0	0	0	0	0	0	0
7	Potatoes	0	1.6	0	0	0	1.7	96.7	0	0	0	0	0	0	0	0
8	Peas	0	0	0	0	0	0.4	0.1	98.7	0	0	0.7	0	0	0	0
9	Stem-Beans	0	0.3	0.1	0.2	0	0.4	0.2	0	98.8	0	0	0	0	0	0
10	Bare-Soil	0.9	0	0	0	0.3	0	0	0	0	98.8	0	0	0	0	0
11	Wheat-A	0	0	0	0	1.8	0	0	0.2	0	0	95	2.7	0.3	0	0
12	Wheat-B	0	0	0	0.2	3.3	0	0	0	0	0	1.8	94.2	0.5	0	0
13	Wheat-C	0	0	0.4	0.8	1	0	0	0	0	0	0.9	0.3	96.6	0	0
14	Borely	0	0	0.5	3.9	0	0.9	0	0	0.1	0.1	0	0	0	94.5	0
15	Building	0	0	0	0	0	0.4	0	0	16.4	0	0	0	0	0	83.2

In conclusion, the CV-CNN method demonstrated superior performance in the classification of the Flevoland dataset compared to other methods. The utilization of phase information, which is a significant aspect of SAR images, played a significant role in achieving this improved accuracy. The results showed an overall accuracy of 96.2% with the majority of classes having an accuracy higher than 95%. The accuracy of buildings was relatively lower due to a limited number of training samples. The analysis of the histogram of different species also indicated that phase information can be an effective feature for the classification of POLSAR images.

VI. CONCLUSION

Table 12 presents the performance of the three different classification methodes, namely Mahalanobis distance, Support Vector Machine (SVM), and Complex-Valued Convolutional Neural Network (CV-CNN), in classifying various land cover categories of Flevoland. Each row of the table represents a different land cover category, and the columns show the percentage of correctly classified samples for each method. For example, for the Water class, the Mahalanobis distance method correctly classified 60.39% of the samples, while SVM and CV-CNN achieved higher accuracies of 94.51% and 99.4%, respectively. Similarly, for the Forest class, Mahalanobis distance, SVM, and CV-CNN achieved accuracies of 63.67%, 80.19%, and 96.8%, respectively.

Table 12. Classification accuracy for whole data with three different classification methods.

CLASSES	MAHALANOBIS DISTANCE	SVM	CV-CNN
Water	60.39	94.51	99.4
Forest	63.67	80.19	96.8
Lucerne	60.21	85.71	98.1
Grass	57.49	53.43	90.0
Rapeseed	59.74	78.81	92.0
Beet	77.66	77.20	97.6
Potatoes	52.28	73.62	96.7
Peas	58.61	75.82	98.7
Stem beans	73.43	78.38	98.8
Bare soil	92.01	69.14	98.8
Wheat A	55.21	75.90	95.0
Wheat B	67.67	52.13	94.2
Wheat C	69.11	85.06	96.6
Barely	77.88	70.62	94.5
Building	68.57	70.10	83.2
OVERALL ACCURACY	63.80%	77.96%	96.2%

The overall accuracy, which represents the percentage of correctly classified samples across all classes, is presented in the last row of the table. The Mahalanobis distance and SVM methods achieved overall accuracies of 63.80% and 77.96%, respectively, while the CV-CNN method achieved the highest overall accuracy of 96.2%.

The aim of the study was to assess the performance of different classification algorithms on the Flevoland dataset, which is a subset of a full polarimetric SAR image. This dataset is particularly challenging due to the diversity of the landscapes it depicts. The results of the algorithms were compared to other commonly used methods in the field, Mahalanobis distance, Support Vector Machine (SVM), and Complex-Valued Convolutional Neural Network (CV-CNN).

The study concludes that the CV-CNN method is superior to traditional methods for classifying SAR images. This superiority is attributed to the effective utilization of phase information in the CV-CNN, which is a crucial characteristic of SAR images. The results of the study demonstrate the potential of CNNs for processing SAR images, thereby paving the way for further research in this area. The study's findings show that significant improvements in the accuracy of CV-CNN can be achieved, thereby extending the range of potential applications for CV-CNN in SAR image interpretation.

REFERENCES

BOOKS :

AMARUDIN, R. FERDIANA AND WIDYAWAN (2020) "**A Systematic Literature Review of Intrusion Detection System for Network Security: Research Trends, Datasets and Methods**" 2020 4th International Conference on Informatics and Computational Sciences (ICICoS), pp. 1-6, doi: 10.1109/ICICoS51170.2020.9299068.

AMROLLAHI M., HADAYEGHPARAST S., KARIMIPOUR H., DERAKHSHAN F., SRIVASTAVA G. (2020) **Enhancing Network Security Via Machine Learning: Opportunities and Challenges**. In: Choo KK., Dehghantanha A. (eds) Handbook of Big Data Privacy. Springer, Cham. https://doi.org/10.1007/978-3-030-38557-6_8.

BRUCE , J. S. (2020). **Automatic Target Recognition** (Vol. TT120). SPIE PRESS BOOK.

G.MULLISSA, C. PERSELLO, and V. TOLPEKINl, "**Fully convolutional networks for multi-temporal SAR image classification**," in Proc. IEEE Int. Geosci. Remote Sens. Symp., 2018, pp. 6635–6638.

HE, M. TU, D. XIONG ET AL ., "**Nonlinear Manifold Learning Integrated with Fully Convolutional Networks for Polsar Image Classification**," Remote Sensing, vol. 12, no. 4, p. 655, 2020.

J. GENG, H. WANG, J. FAN and X. MA, "**SAR Image Classification via Deep Recurrent Encoding Neural Networks**," in IEEE Transactions on Geoscience and Remote Sensing, vol. 56, no. 4, pp. 2255-2269, April 2018

LEE & POTTIER - **Polarimetric Radar Imaging**.

LI DENG and DONG YU. **DEEP LEARNING: METHODS AND APPLICATIONS** Microsoft Research One Microsoft Way Redmond, WA 98052.

ARTICLES :

LEKUN ZHU, XIAOSHUANG MA, PENGHAIh WU and JIANGONG XU. **Multiple Classifiers Based Semi-Supervised Polarimetric SAR Image Classification Method.**

LUCIANA O. PEREIRA, LUIZ F. A. FURTADO , EVLYN M. **Multifrequency and Full-Polarimetric SAR** Assessment for Estimating Above Ground Biomass and Leaf Area Index in the Amazon Várzea Wetlands L. M. Novo 2 ID , Sidnei J. S. Sant'Anna 2 , Veraldo Liesenberg 4 ID and Thiago S. F. Silva.

JOURNALS :

ALISON B. **Deep Learning with GPU Technology for Image & Feature Recognition.** Lowndes Submitted in accordance with the requirements for the degree of Bachelor of Science in Artificial Intelligence 2012/2015.

BAMBRICK N., (2020). **Support Vector Machines**. [Online] Available at: https://www.kdnuggets.com/2016/07/support-vector-machinessimpleexplanation.html

B. HOU, H. D. KOU, and L. C. JIAO, "**Classification of polarimetric SAR images using multilayer autoencoders and superpixels**," IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 9, no. 7, pp. 3072–3081, Jul. 2016.

CHEN YANG, BIAO HOU, **CNN-Based Polarimetric Decomposition Feature Selection for PolSAR Image Classification.** Member, IEEE, Bo Ren , Member, IEEE, Yue Hu, and Licheng Jiao, Fellow, IEEE 2019.

HAIXIA BI, JIAN SUN, **A Graph-Based Semisupervised Deep Learning Model for PolSAR Image Classification** Member, IEEE, and Zongben Xu.2019

HONGWEI DONG , LAMEI ZHANG , **Attention-Based Polarimetric Feature Selection Convolutional Network for PolSAR Image Classification.** Senior Member, IEEE, Da Lu , Member, IEEE, and Bin Zou , Senior Member, IEEE 2022.

JURGEN SCHMIDHUBER The Swiss AI Lab IDSIA, **Deep learning in Neural Networks: An Overview.** Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, University of Lugano & SUPSI, Galleria 2, 6928 Manno-Lugano, Switzerland 2015.

MOHSEN JAFARI, YASSER MAGHSSOUDI, **A New Method for Land Cover Characterization and Classification of Polarimetric SAR Data Using Polarimetric Signatures.** Member, IEEE, and Mohammad Javad Valadan Zoej,2015.

QIAN WU, ZAIDOU WEN, **A Statistical-Spatial Feature Learning Network for PolSAR Image Classification.** Member, IEEE, Yongqing Wang , Yanbo Luo, Hao Li, and Qiushi Chen,2022.

ZHIMIAN ZHANG, HAIPENG WANG, "**Complex-Valued Convolutional Neural Network and Its Application in Polarimetric SAR Image Classification**", Senior Member, IEEE, Feng Xu, Senior Member, IEEE, and Ya-Qiu Jin, Fellow, IEEE. VOL. 55, NO. 12, DECEMBER 2017.

ELECTRONIC JOURNALS:

BAMBRICK N., (2020). **Support Vector Machines**. [Online] Available at:

https://www.kdnuggets.com/2016/07/support-vector-machines-simpleexplanation.html

R.-E. FAN, K.-W. CHANG, C.-J. HSIEH, X.-R. WANG, AND C.-J. LIN. **LIBLINEAR: A Library for Large Linear Classification**, Journal of Machine Learning Research 9(2008), 1871-1874. Sage Journals.[Online] Available at: https://journals.sagepub.com/doi/10.1177/000812561986492

URL-1 "**Regularization in Machine Learning - Javatpoint**", www.javatpoint.com, 2022. [Online]. Available: https://www.javatpoint.com/regularization-in-machine-learning, [Accessed: 10- May- 2022].

URL-2 "**Regression Analysis in Machine learning - Javatpoint**", www.javatpoint.com, 2022. [Online]. Available:

https://www.javatpoint.com/regression-analysis-in-machine-learning .
[Accessed: 01- Jun- 2022]

PAPERS:

ADUGUNA G. MULISSA, **PolSARNet: A Deep Fully Convolutional Network for Polarimetric SAR Image Classification** IEEE, Claudio Persello , Senior Member, IEEE, and Alfred Stein 2019.

CEDRIC LARDEUX, PIERRE-LOUIS FRISON, CELINE TISON, JEAN-CLAUDE Souyris, Benoît Stoll, Bénédicte Fruneau, and Jean-Paul Rudant. **Support Vector Machine for Multifrequency SAR Polarimetric Data Classification.** 2009.

DAIFENG XIAO, ZHIRUI WANG , **Terrain Segmentation in Polarimetric SAR Images Using Dual-Attention Fusion Network.** Youming Wu , Xin Gao, and Xian Sun.

F. LIU, L. JIAO, and X. TANG, "**Task-Oriented Gan For Polsar Image Classification and Clustering**," IEEE transactions on neural networks and learning systems, vol. 30, no. 9, pp. 2707–2719, 2019.

F. ZHAO, M. TIAN, W. XIE ET AL., "**A New Parallel Dual-Channel Fully Convolutional Network Via Semi-Supervised FCM for Polsar Image Classification**," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 13, pp. 4493–4505, 2020.

GAO, J.; DENG, B.; QIN, Y.; WANG, H.; LI, X. **Enhanced Radar Imaging Using a Complex-Valued Convolutional Neural Network**. IEEE Geosci. Remote Sens. Lett. 2019, 16, 35–39.

H. BI, J. SUN, and Z. XU, "**A Graph-Based Semisupervised Deep Learning Model for Polsar Image Classification**," IEEE Trans. Geosci. Remote Sens., vol. 57, no. 4, pp. 2116–2132, 2018.

H. DONG, L. ZHANG, and B. ZOU, "**Polsar Image Classification with Lightweight 3d Convolutional Networks**," Remote Sensing, vol. 12, no. 3, p. 396, 2020.

J. KLICPERA, S. WEIENBERGER, and S. GUNNEMANN, "**Diffusion Improves Graph Learning**," in Proc. Neural Information Processing Systems (NeurIPS), Vancouver, Canada, Dec. 2019

J. R. BERGADO, C. PERSELLO, and A. STEIN, "**Recurrent Multi-Resolution Convolutional Networks for VHR Image Classification**," IEEE Trans. Geosci. Remote Sens., vol. 56, no. 11, pp. 6361–6374, Nov. 2018.

LAMEI ZHANG, **Fully Polarimetric SAR Image Classification via Sparse Representation and Polarimetric Features.** IEEE, Liangjie Sun, Bin Zou, Member, IEEE, and Wooil M. Moon, Life Fellow, IEEE 2022.

L. CHEN, G. PAPANDREOU, I. KOKKINOS, K. MURPHY and A. L. YUILLEY, "**DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs**," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 4, pp. 834-848, 1 April 2018.

LEE, J.S.; POTTIER, E. **Polarimetric Radar Imaging: From Basics to Applications**; CRC Press: Boca Raton, FL, USA, 2009.

LICHENG JIAO, SENIOR Member, IEEE, and FANG LIU. **Wishart Deep Stacking Network for Fast POLSAR Image Classification**.

LIU, W.; QIN, R.; SU, F. **Weakly supervised classification of time-series of very high-resolution remote sensing images by transfer learning**. Remote Sens. Lett. 2019, 10, 689–698.

LI, Y.; ZHANG, Y.; ZHU, Z. **Learning Deep Networks under Noisy Labels for Remote Sensing Image Scene Classification**. In Proceedings of the 2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 3025–3028.

L. JIAO and F. LIU, "**Wishart Deep Stacking Network for Fast POLSAR Image Classification**," IEEE Trans. Image Process, vol. 25, no. 7, pp. 3273–3286, Jul. 2016.

LV, W.J.; WANG, X.F. **Overview of Hyperspectral Image Classification**. J. Sens. 2020, 2020, 4817234.

L. WANG, X. Xu, H. DONG ET AL., "**Exploring Convolutional lstm for Polsar Image Classification**," in Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS). IEEE, 2018, pp. 8452–8455.

MIRCEA SERBAN PAVEL, HANNES SCHULZ, and SVEN BEHNKE **Object Class Segmentation of RGB-D Video using Recurrent Convolutional Neural Networks** Universit¨at Bonn Computer Science Institute VI Friedrich-Ebert-Allee 144 53113 Bonn pavel@cs.uni-bonn.de, schulzh@ais.uni-bonn.de, behnke@cs.uni-bonn.de

M.Sc. THESIS Eng. Maha Mohammed Attieya AL-BAYATİ **POLARIMETRIC SAR DATA INTERPRETATION** Department of Electrical and Electronics Engineering Electrical and Electronics Engineering Program 2020.

NEGRI, R.G.; FRERY, A.C.; SILVA, W.B.; MENDES, T.S.G.; DUTRA, L.V. **Region-Based Classification of PolSAR Data using Radial Basis Kernel Functions with Stochastic Distances**. Int. J. Digit. Earth 2019, 12, 699–719.

RONNY HANSCH, O.H. **Classification of Polarimetric SAR Data by Complex Valued Neural Networks**; ISPRS Hannover Workshop-highresolution Earth Im.2017.

RUI, M., HAI, L., ZONGIIE, C., & ZONGYONG CUI. (2019, March 21). **A Gradually Distilled CNN for SAR Target Recognition.** IEEE ACCESS, 7. doi:10.1109

S. CHEN and C. TAO, "**Polsar Image Classification using Polarimetric Feature-Driven Deep Convolutional Neural Network**," IEEE Geoscience and Remote Sensing Letters, vol. 15, no. 4, pp. 627–631, 2018.

SHIJIE REN, FENG ZHOU, **Semi-Supervised Classification for PolSAR Data with Multi-scale Evolving Weighted Graph Convolutional Network.** IEEE

SI-WEI CHEN, **PolSAR Image Classification Using Polarimetric-Feature-Driven Deep Convolutional Neural Network.** IEEE, and Chen-Song Tao.2022.

SUN, F.; FANG, F.; WANG, R.; WAN, B.; WU, X. **An Impartial Semi-Supervised Learning Strategy for Imbalanced Classification on VHR Images**. Sensors 2020, 20, 6699.

UMUT, Ö. **Automatic Target Recognition (ATR) from SAR Imaginary by Using Machine Learning Techniques**. European Journal of Science and Technology, 165-169. doi: 10.31590/ejosat.802811(2020, September).

WANG YANG1,2, LU JIAGUIO2 & WU XIANLIANG1 1. **New algorithm of target classification in polarimetric SAR** Key Lab of Intelligent Computing & Signal Processing of Ministry of Education, Anhui Univ., Heifei 230039, P. R. China; 2. The 38th Research Inst., China Electronic Technology Corporation, Hefei 230031, P. R. China (Received November 4, 2006)

W. LIU, J. YANG, P. LI, ET AL. "**A Novel Object-Based Supervised Classification Method with Active Learning and Random Forest for PolSAR Imagery**," in Remote Sensing, vol. 10, no. 7, pp. 1092, 2018.

XIE, W.; MA, G.; ZHAO, F.; LIU, H.; ZHANG, L. **PoISAR Image Classification via a Novel Semi-Supervised Recurrent Complex-Valued Convolution Neural Network**. Neurocomputing 2020, 388, 255–268.

Y. CAO, Y. WU, P. ZHANG et al., "**Pixel-Wise Polsar Image Classification Via a Novel Complex-Valued Deep Fully Convolutional Network**," Remote Sensing, vol. 11, no. 22, p. 2653, 2019

Y. GUO, S. WANG, C. GAO, D. SHI, D. ZHANG, and B. HOU, "**Wishart RBM Based DBN for Polarimetric Synthetic Radar Data Classification**," in Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS), Jul. 2015, pp. 1841–1844

YU ZHOU, HAIPENG WANG, **Polarimetric SAR Image Classification Using Deep Convolutional Neural Networks.** IEEE, Feng Xu, Senior Member, IEEE, and Ya-Qiu Jin, Fellow, IEEE 2016.

ZHONG-QIU ZHAO, **Object Detection with Deep Learning: A Review** IEEE, Peng Zheng, Shou-tao Xu, and Xindong Wu, Fellow, IEEE

RESUME

IKRAM BEN- WARRAK

Electrical and Electronics

Engineer

As a highly motivated and ambitious graduate student specializing in Electric and Electronic Engineering, I am dedicated to driving technological advancements in the field. With a solid foundation in theoretical principles and hands-on experience, I possess a strong passion for innovation and problem-solving. Throughout my academic journey, I have consistently demonstrated exceptional technical proficiency, adaptability, and a collaborative approach to projects. My goal is to leverage my knowledge and skills to contribute to the development of cutting-edge electrical systems and sustainable solutions that address global challenges. Committed to continuous learning and staying abreast of emerging technologies, I am seeking opportunities to apply my expertise in a dynamic and forward-thinking organization.

Education

September 2020 - May 2022

MSc. Electrical and Electronics Engineering

Thesis in Machine Learning and Deep Learning Techniques.

ISTANBUL AYDIN University - Turkey

- CGPA: 4.00

September 2015 - June 2018

Bachelor in Mechanics and Energy.

Hassan II University Casablanca, Morocco.

September 2014 - June 2015

SCIENTIFIC BACCALAUREATE Option Physics. Casablanca, Morocco

Expertise

- Identity and Access Management Principles.
- Electrical wiring and planning.
- Machine Learning and Artificial Intelligence

Skills

- MATLAB
- Microsoft Office
- Problem-solver
- Team Player
- Self Motivated
- Punctual
- Time management

Languages

- Arabic
- English
- French
- Turkish