WARSAW UNIVERSITY OF TECHNOLOGY

INFORMATION AND COMMUNCATIONS TECHNOLOGY

ENGINEERING AND TECHNOLOGY

Ph.D. Thesis

I Made Sandhi Wangiyana, M.Sc.

Deep Learning Methods for Automated Urban Monitoring System using Synthetic Aperture Radar

Supervisor

Professor Piotr Jerzy Samczyński, Ph.D., D.Sc.

WARSAW 2024

Acknowledgements

I express my sincere gratitude to my supervisor Professor Piotr Samczyński, for his guidance, wisdom, and positivity, not just for research work but for my overall academic career. It still amazes me how he managed to find time among his tight daily meetings when I needed his feedback. I would also like to express my gratitude to Bang Zulkieflimansyah for his visionary scholarship program, providing the opportunity for a local West Nusa Tenggara residence like me to study in Europe.

Huge thanks to Karol Abratkiewicz, Muhammad Abbiyu, Joanna Pluto-Kossakowska, Miftah Farid, and Rauzan Sumara, for reviewing the draft and providing thoughtful comments. Many thanks to Pedro Gomez, Marta Malik, Łukasz Maślikowski, Marcin Żywek, Artur Gromek, and other colleagues from Warsaw University of Technology for their warmness, support, and wonderful conversations during my study.

My deepest gratitude goes to my family for their unconditional love and support throughout my life. Special thanks to Aulia Anne for your love, care, and patience. This thesis is simply impossible without them. Finally, many thanks to Intan Puspitaningrum, Megha Asri, Melissa Manurung, and many other Indonesian diasporas in Warszawa and Kraków for always welcoming and bringing the great joy of feeling at home.

Streszczenie

Obszary miejskie pomimo zajmowania niewielkiej części powierzchni Ziemi stanowią centrum ludzkich osad i działalności gospodarczej i mają kluczowe znaczenie dla monitorowania zmian. Zaczynając od szybkiej oceny wpływu klęsk żywiołowych, aż po uchwycenie dynamiki rozwoju miast, dane teledetekcyjne są potrzebne do analizy pokrycia dużych obszarów. Radar z syntetyczną aperturą (ang. *Synthetic Aperture Radar, SAR*) jest jednym z instrumentów teledetekcyjnych, które mogą zapewnić globalne i ciągłe obserwacje Ziemi. Jego zdolność do przenikania przez chmury i niezależność od światła słonecznego jest zaletą w porównaniu z obrazami optycznymi. Jednak jego unikalne właściwości są złożone i trudne do przeanalizowania przez osoby niebędące ekspertami. Fakt ten prowadzi do wykorzystania głębokiego uczenia i sieci neuronowych, które w ciągu ostatniej dekady stworzyły zmianę paradygmatu opracowywania algorytmów opartych na danych w sposób kompleksowy.

Proponowany system monitorowania zmian byłby rozwiązaniem dwuetapowym: wykrywanie zdarzeń na skalę miejską przy globalnym monitorowaniu środowiska, a następnie analiza cech poszczególnych budynków na obrazach o wyższej rozdzielczości.

Niniejsza rozprawa przedstawia badania nad wykonalnością i użytecznością algorytmów głębokiego uczenia do zautomatyzowanej analizy obszarów miejskich z wykorzystaniem obrazów SAR. W pierwszym etapie zaproponowano nienadzorowaną metodę uczenia się przy użyciu lekkiego autoenkodera w celu uzyskania cech wysokiego poziomu do wykrywania dużych zmian w wieloczasowych obrazach SAR. W drugim etapie zaproponowano dwie metody do detekcji zabudowy: ekstrakcję obrysów budynków i klasyfikację pokrycia terenu. Niniejsza praca koncentruje się na maksymalizacji wykorzystania danych o intensywności na zobrazowaniach SAR, które są zarejestrowanymi wartościami rozproszenia mikrofali dostępnymi w teledetekcyjnych systemach SAR. Proponowane algorytmy zostały wytrenowane przy użyciu wystarczającej wielkości zbioru danych uczących w różnych obszarach miejskich. Wyniki wykazały istotną wydajność generalizacji, która jest niezbędna dla systemu monitorowania zmian o globalnym zasięgu.

Słowa kluczowe: Radar z syntetyczną aperturą, Uczenie Głębokie, Detekcja zmian, Segmentacja, Analiza urbanistyczna

Abstract

Despite covering a small portion of the Earth's land surface, urban areas are critical to monitor since they are the center of human settlements and economic activities. From a quick assessment of natural disasters' impact to capturing the dynamics of urban growth, remote sensing data is needed for the analysis of large area coverage. Synthetic Aperture Radar (SAR) is one of the remote sensing instruments that can provide global and continuous observations of the Earth. Its ability to penetrate clouds and not depend on sunlight is an advantage over optical sensors. However, its unique properties are difficult for non-experts to analyze. This fact leads to the exploitation of deep learning and neural networks, which, for the past decade, have created the paradigm shift of developing data-driven algorithms in an end-to-end manner.

The ideal monitoring system would be a two-stage solution: detection of city-wide events with global coverage monitoring, and then analysis of building-unit features in higher-resolution images.

This thesis explores the feasibility of deep learning algorithms for automated urban analysis using SAR. For the first stage, an unsupervised learning method using a lightweight autoencoder was proposed to output high-level features for detecting large-event changes in multitemporal SAR images. For the second stage, two SAR building-unit analyses were proposed: extraction of building footprints and land classification. This work focuses on maximizing the usage of SAR intensity data, which is the projected radar echoes available in all SAR systems. Proposed algorithms were trained using sufficient dataset size in diverse urban scenes. Results have demonstrated reasonable generalization performance, which is necessary for a monitoring system with global coverage.

Keywords: Synthetic Aperture Radar, Deep Learning, Change Detection, Segmentation, Urban Analysis

Contents

1 Introduction 12

1.1 SAR for Disaster Response 12

1.2 SAR for Urban Analysis 13

1.3 SAR and Deep Learning for Monitoring Systems 14

1.4 Research Goals 15

1.5 Contributions 16

1.6 Organization 16

2 Synthetic Aperture Radar 19

2.1 History of SAR 19

2.1.1 Radar Equation 20

2.1.2 Side-Looking Airborne Radar 21

2.2 SAR Image Generation 22

2.2.1 Range Dimension 23

2.2.2 Azimuth Dimension 24

2.3 SAR Acquisition Modes 26

2.3.1 Data Acquisition 26

2.3.2 Polarimetry 27

2.3.3 Interferometry 27

2.4 SAR Image Analysis 28

2.4.1 Surface Roughness 29

2.4.2 Speckle 29

2.4.3 Geometric Distortions 30

3 Deep Learning 32

3.1 The Learning Algorithm 32

3.1.1 The Task 32

3.1.2 The Performance Metric 33

3.1.3 The Learning Experience 33

3.2 Artificial Neural Networks 34

3.2.1 The Neuron 34

3.2.2 A Network of Neurons 35

3.2.3 Gradient Descent 36

3.2.4 Information Theory 37

3.3 Convolutional Neural Network 39

3.3.1 The convolution layer 39

3.3.2 Segmentation Models 40

3.4 Evaluation Metrics 42

4 Deep Learning for Building Unit Damage Assessment using SAR: Progress and Challenges 45

4.1 Introduction 45

4.2 Physical interpretations of building damage in SAR 46
4.2.1 Intensity 46
4.2.2 Coherence 47
4.2.3 Polarimetry 48
4.2.4 Unit of analysis 50

4.3 State of the art 50
4.3.1 Building footprint extraction in SAR 51
4.3.2 Optical-based building damage assessment 51
4.3.3 Building-unit damage assessment in SAR 53

4.4 Advancing open research on building damage assessment 53
4.4.1 Public Dataset 53
4.4.2 Open data providers (SAR data) 54
4.4.3 Damage Assessment (labels) 55

4.5 Discussion 56
4.5.1 Challenges 56
4.5.2 Future directions 56

4.6 Conclusion 58

5 Data Augmentation for Building Footprint Segmentation in SAR Images: An Empirical Study 59

5.1 Introduction 59

5.2 Methods 60
5.2.1 Dataset overview 60
5.2.2 Segmentation Model 62
5.2.3 Training and Evaluation 62
5.2.4 Ablation study 64

5.3 Data Augmentation 64
5.3.1 Reduce Transformation 64
5.3.2 Geometric Transformation 65
5.3.3 Pixel Transformation 67
5.3.4 Speckle Filters 68
5.3.5 Data Augmentation Design and Strategy 69

5.4 Results 70
5.4.1 Ablation Study 70
5.4.2 Main Experiment 74
5.4.3 Test-Time Augmentation 76

5.5 Discussion 77

5.6 Conclusion 78

6 Detecting Large Scale Event from SAR Time Series 79

6.1 Introduction 79
6.2 Dataset 80
6.2.1 Flood Training Dataset 80
6.2.2 General Event Evaluation Dataset 81
6.2.3 Notation 81
6.2.4 Preprocessing 82
6.3 Methodology 83
6.3.1 Autoencoders 83
6.3.2 Training 84
6.3.3 Evaluation 84
6.4 Results 85
6.4.1 Reconstructing Flood Events 85
6.4.2 Predicting on Other Events 86
6.5 Discussion 89
6.6 Conclusion 90
7 SAR Imagery for Urban Density Analysis 91
7.1 Introduction 91
7.2 Dataset 92
7.2.1 Study area and SAR data 92
7.2.2 Labels and urban density definition 94
7.2.3 SAR features 94
7.3 Methodology 98
7.3.1 Workflow 98
7.3.2 Algorithms 99
7.3.3 Evaluation 100
7.4 Results 101
7.4.1 Data Augmentation and Combining Features 103
7.5 Discussion 104
7.5.1 Weak descriptive features from single and dual polarization SAR 104
7.5.2 Trade-off between object size and details 104
7.5.3 Reliability of Urban Atlas 105
7.5.4 Discussion on accuracy 107
7.6 Conclusion 107
8 Conclusion 109
8.1 Research summary 109
8.2 Future work 110
References 111

List of Acronyms

AI	**A**rtificial **I**ntelligence
AOI	**A**rea **o**f **I**nterest
AUPRC	**A**rea **U**nder the **P**recision **R**ecall **C**urve
BDA	**B**ulding **D**amage **A**ssessment
BFE	**B**uilding **F**ootprint **E**xtraction
CD	**C**hange **D**etection
CEMS	**C**opernicus **E**mergency **M**anagement **S**ervice
CNN	**C**onvolutional **N**eural **N**etwork
DA	**D**ata **A**ugmentation
DL	**D**eep **L**earning
ESA	**E**uropean **S**pace **A**gency
FN	**F**alse **N**egative
FP	**F**alse **P**ositive
FPN	**F**eature **P**yramid **N**etwork
FPR	**F**alse **P**ositive **R**ate
GLCM	**G**ray **L**evel **C**o-occurrence **M**atrix
GMAP	**G**amma **M**aximum **A** **P**osteriori
GPU	**G**raphics **P**rocessing **U**nit
GRD	**G**round **R**ange **D**etected
HH	**H**orizontal transmit **H**orizontal receive
HR	**H**igh **R**esolution
HV	**H**orizontal transmit **V**ertical receive
InSAR	**I**nterferometric **S**ynthetic **A**perture **R**adar
IoU	**I**ntersection **o**ver **U**nion
LULC	**L**and **U**se **L**and **C**over
mFPR	**m**ean **F**alse **P**ositive **R**ate
mIoU	**m**ean **I**ntersection **o**ver **U**nion
ML	**M**achine **L**earning
mPrec	**m**ean **Prec**ision
mRec	**m**ean **Rec**all
OSM	**O**pen**S**treet**M**ap
PolSAR	**P**olarimetric **S**ynthetic **A**perture **R**adar
RCS	**R**adar **C**ross **S**ection
ReLU	**R**ectified **L**inear **U**nit
RGB	**R**ed, **G**reen, **B**lue
SAR	**S**ynthetic **A**perture **R**adar
SLAR	**S**ide-**L**ooking **A**irborne **R**adars
SLC	**S**ingle **L**ook **C**omplex
SNAP	**S**entinel **A**pplication **P**latform
SSL	**S**emi-**S**upervised **L**earning
TN	**T**rue **N**egative
TP	**T**rue **P**ositive

TTA	**T**est **T**ime **A**ugmentation
UA	**U**rban **A**tlas
VH	**V**ertical transmit **H**orizontal receive
VHR	**V**ery **H**igh **R**esolution
VV	**V**ertical transmit **V**ertical receive

List of symbols and operators

c	Speed of light
t	Time
P_R	Power received
P_S	Power sent
G	Antenna gain
λ	Wavelength
σ	Radar Cross Section (RCS)
σ_0	Backscatter coefficient
$\lvert \cdot \rvert$	Absolute value operator
E_S	Scattered energy
E_i	Intercepted energy
A	Area
H	Altitude of flying vehicle
V	Velocity of flying vehicle
T_p	Pulse duration
ρ_R	Range resolution
ρ_G	Ground resolution
ρ_A	Azimuth resolution
θ_i	Incidence angle
R	Distance
R_0	Distance to the center of radar footprint
L	Antenna length
W	Antenna width
W_s	Swath width
f_r	Pulse Repetition Frequency
f_s	Sampling rate
θ_H	Beamwidth in azimuth axes
θ_V	Beamwidth in range axes
X	A random variable
x	Values of a random variable X
y	Target vector (ground truth)
$\hat{y}$	Target vector (predictions)
z	Embedding vector
f_a	Activation function
w	Weights/parameter vector of a neural network
b	Bias of a neuron
$\mathcal{L}$	Loss or error function
α	Learning rate
$\mathcal{H}(P)$	Entropy of a probability distribution P
D_{KL}	Kullback-Leibler divergence
C_E	Cross-entropy
e	Euler's number
$\mathbb{E}[x]$	Expected value of x

γ	Coherence
S	Scattering matrix
T	Coherency matrix
X_t	Sample image at time t
$x_t^{r,c}$	Tile from X_t at row r and column c
$\hat{x}$	Reconstruction of x
$DI_{\tau,t}$	Difference image between images of time τ and time t
$dt_{\tau,t}^{r,c}$	Difference tile from $DI_{\tau,t}$ at row r and column c
$\mathcal{E}$	Encoder
$\mathcal{D}$	Decoder
S_C	Cosine similarity
$\mathcal{T}$	Threshold
c	A label class
$\not{p}$	A SAR feature

1 Introduction

Synthetic Aperture Radar (SAR) has been widely used in Earth observation for many decades. It provides images regardless of weather and independent of sunlight for various remote sensing applications, from climate change research, environmental monitoring, planetary exploration, and disaster analysis. This dissertation uses SAR as the main source of data for automated analysis that were applied to disaster response and urban monitoring.

1.1 SAR for Disaster Response

On February 6, 2023, a catastrophic magnitude 7.8 earthquake hit the borders of southeast Turkey and northwest Syria, which was followed by a magnitude 7.5 earthquake nine hours later [1]. In Turkey alone, at least 270 thousand buildings were collapsed, giving an estimate of 210 million tons of rubble [2]. The event resulted in the death of over 50 thousand people which most were casualties from the direct hit of collapsed residential apartments or buried under the rubble.

The two events exceeded expectations not only in magnitude but also in terms of the damage they caused. According to studies [3], the wide rupture length was caused by the earthquake doublet rupturing multiple segments of the East Anatolian Fault Zone in one go and causing a larger slip. The main quake rupture extended over 300 km while the aftershock resulted in a shorter rupture of about 100 km but larger land displacements of up to 7-8 m. Moreover, the two large earthquakes occurred in neighboring fault zones, resulting in widespread damage in an area of about 350,000 km^2 [4]. Turkey's president declared a three-month state of emergency in 10 provinces.

Figure 1.1 The town of Islahiye in Gaziantep province. Optical images from Maxar [5] a) before the earthquake (December 27, 2022), b) after the earthquake (February 7, 2023), and c) post-earthquake SAR image from Capella Space (February 9, 2023) [6].

Early emergency response focuses on removing rubble and rescuing buried civilians. It is necessary to locate the worst hit areas, detect collapsed buildings, and determine rescue routes. However, the challenge is in analyzing such a large area in the shortest amount of time possible, as less time meant more rescued lives. Additionally, extreme cold weather that hit the regions hampered rescue operations. Damaged roads and rubble also made it difficult to find survivors and get crucial aid into affected areas. Several airports have also been closed after being damaged by the earthquakes. That is why remote sensing plays a critical role in disaster management. The earliest aerial or satellite images can be used to coordinate humanitarian aid. Several commercial space companies such as Maxar [5], Capella Space [6], Umbra [7], and ICEYE [8] have released Very High Resolution (VHR) satellite data at 50 cm/pixel to aid in emergency response. An example of this humanitarian aid data is in Figure 1.1. These remote sensing data were used by disaster relief groups such as Copernicus Emergency Management Service (CEMS) and Humanitarian OpenStreetMap, which actively assessed the situation, manually identifying damaged buildings from optical images. However, optical sensors depend on good weather conditions for optimal analysis. Such situations are not usual post-disaster events, such as Cyclone Freddy that struck Mozambique and Madagascar in the same month and year [9] which was closely monitored using SAR [10].

SAR sensors have strong advantages over optical sensors for monitoring purposes and for rapid analysis. Using active instruments that can penetrate through clouds, data can be obtained independently from daylight and weather. This means the data can be used almost immediately after an airborne or spaceborne instrument passes.

1.2 SAR for Urban Analysis

As stated by Fedra [11], urban management addresses the problems that are spatially distributed as well as dynamically changing. One of the primary use cases of remote sensing in urban monitoring is the creation and updating of urban maps. High-resolution airborne or spaceborne images can capture the current layout of buildings and roads, providing essential information required by urban planners, engineers, and policymakers to make informed decisions. These maps are crucial for understanding the spatial distribution of infrastructure and can help identify areas where new developments are feasible. SAR's ability to penetrate canopies enables the detection of hidden structures, providing a more comprehensive view of the urban environment.

Urban growth is another important use case. As cities expand, it is crucial to monitor how new developments affect existing infrastructure and the environment from the effects of

increased greenhouse gas emissions. Built-up structures induce strong backscatter in radar images and thus can be distinguished from natural objects [12]. This information helps urban planners to manage growth sustainably, ensuring that new developments are integrated smoothly into the existing urban fabric.

1.3 SAR and Deep Learning for Monitoring Systems

Historically, radar images have been developed with the main usage in military applications. Interpretation requires experts with knowledge of statistics, information theory, and signal processing. Currently, SAR has gained more usage in civil applications. Sentinel-1 constellation by the European Space Agency (ESA) has played a significant role in most SAR research because of the global coverage and free-of-charge usage [13]. Over the past years, better SAR technology has emerged. Commercial space companies providing SAR data have moved towards microsatellite constellations, increasing the flexibility for high temporal coverage, and different imaging modes to cover wider areas or capture finer details.

This abundance of data along with the popularity of artificial intelligence gave birth to a plethora of data-driven algorithms using Deep Learning (DL) and neural networks. The unobstructed continuous observation capabilities provided by SAR sensors and the automated end-to-end analysis from deep learning are a promising combination for a monitoring system.

Despite the promise, there are limitations regarding this solution. New DL models and architectures are typically benchmarked on natural RGB (Red, Green, Blue) images. This brings the challenge of bridging the domain gap between natural everyday images to overhead remote sensing images, and finally to side-looking SAR images. DL scales with more quality data. Despite the rise in space companies or startups, it is still difficult to obtain large amounts of relevant data for training, due to:

- The trade-off between spatial resolution and swath width (coverage area). This means there is no single approach to collecting SAR data for the multitude of applications it is used for. In target recognition, the highest resolution possible is required for a higher chance of identifying key features of the target. In flood detection, ice monitoring, and earthquake deformation analysis, a wider swath is favorable to delineate the large area impact. SAR providers usually have different imaging modes to manipulate this trade-off.
- The use of microsatellites is meant to give customers flexible control of imaging mode, such as the area of interest, frequency of observation, coverage, etc. However, this means that historical images of areas hit by disasters are rarely available since it's difficult to anticipate

and pre-emptively plan a monitoring mission to specific areas. Therefore, only post-event images are usually available.

- Furthermore, unlike in optical images where images from different sensors can be "harmonized" to obtain consistent radiometric values [14], the ground response from SAR sensors is very sensitive to the imaging mode and the instrument, such as the look angle, direction, and frequency band. This adds the challenge of learning hidden patterns from the data distribution when data from different providers are combined.

With these limitations in mind, a feasible monitoring system will require a two-stage detection. The first stage is for monitoring and detecting large-scale events in global coverage. The second stage is performing building-unit analysis from highly detailed images. This dissertation aims to verify the feasibility of this monitoring system with the purpose of disaster mitigation and urban analysis.

1.4 Research Goals

The theses of this dissertation are as follows:

- It is possible to improve the classification of building footprints using data augmentation for a limited set of SAR images.
- It is possible to detect large event changes from multitemporal SAR images using an autoencoder that was trained in an unsupervised way.
- It is possible to do urban Land Use Land Cover (LULC) classification on a single polarization SAR image.

To solve these theses, the goal of deep learning, which is generalization, needs to be considered. Generalization is the ability to predict sufficiently well in a broad range of problems in various study areas. Considering the limitations mentioned in the previous section, this research focuses on algorithms applied to the SAR intensity image which is the most commonly available SAR data. Therefore, the phase information used in interferometric analysis, or polarimetric decompositions used with polarimetric data are not considered. Research was conducted with the following aims:

- Evaluation and benchmarking of the performance of state-of-the-art neural network architectures used in Computer Vision research on SAR data.
- Development and validation of pre-processing methods for fitting large remote sensing data to reasonably sized neural networks.

- Experimentation of various data augmentation strategies specifically for radar images.
- Experimentation of algorithms on various urban landscapes and acquisition modes.

1.5 Contributions

Addressing the study directions mentioned above, several contributions were made in progressing the field of DL for SAR which were published in a journal and various conference proceedings:

- Experimental verification of data augmentation methods and strategy specific to SAR images. It was shown that geometrical transformations were more effective than pixel transformation, except for quarter rotation and vertical flip, which lowered performance due to extreme displacement of the shadow and layover patterns from large infrastructures [15], [16].
- Experimental verification on the use of explainable methods for analyzing neural network's decision making in SAR object classification [17].
- Development of a general large-event detector using a lightweight autoencoder trained on unlabeled multitemporal SAR images [18], [19].
- Review of the state-of-the-art research of DL-based building-unit damage assessment from SAR images.
- Open-sourced the codes used in all experimental chapters in this dissertation, from the collection of remote sensing data, preprocessing, training pipeline, to benchmark results. The codes can be found at https://github.com/sandhi-artha/dissertation.

1.6 Organization

The dissertation is organized as follows:

Chapter 2

The main data source of this work is SAR images in urban scenes. Unlike optical images, SAR is not intuitive. It has unique properties present in urban landscapes such as layover, foreshortening, and shadows. This chapter summarizes basic SAR theory and image interpretation that will be leveraged in analysis for subsequent chapters.

Chapter 3

The main method used in this work is Deep Learning (DL). This chapter explains the main building blocks of DL, which is the Artificial Neural Network (ANN), and summarizes how

learning is performed. The Convolutional Neural Network (CNN) and the autoencoder as the primary architecture used in this study are also discussed.

Chapter 4

Building Damage Assessment (BDA) is important to plan optimal emergency responses after a disaster. This chapter aims to summarize state-of-the-art research of deep learning-based methods for building-unit damage assessment using SAR. Approaches for quantifying building damage in different SAR features were reviewed. The challenges of building-unit analysis and ways to advance open research were discussed. The review concludes with key findings and opportunities for future research.

Chapter 5

Building information is a valuable resource in disaster management. The task of building footprint extraction using SAR images still falls behind the optical images mainly from the limited amount of data and the unique geometric and radiometric features of building objects. In this chapter, data augmentation was proposed as the solution to the limited SAR datasets and to improve robustness from SAR specific features. Experiments were conducted on various transformation methods and their impact on the segmentation performance. The study provides insights on selecting augmentation methods that improve detection from radar imagery.

Chapter 6

Urban analysis from remote sensing images generally requires very high resolution (VHR) data to identify various sized man-made objects such as buildings and roads. In SAR, high spatial resolution comes at the cost of smaller coverage, not to mention the massive storage required. In this chapter, the autoencoder was proposed to detect large event changes from Sentinel-1 multitemporal SAR data. It was trained in an unsupervised manner to learn representations from crops of SAR images. The distance between representations of pre and post-images in the latent space was used to identify areas that have encountered significant change.

Chapter 7

The potential of SAR data to provide up-to-date information can be used in urban density analysis. Built-up structures can be distinguished through scattering mechanisms which can be visible in Polarimetric SAR data. However, Polarimetric modes are not common in satellite operations. This chapter compares the use of single polarization X-band and dual polarization C-band SAR data for LULC in urban areas. The Urban Atlas dataset was used as a reference.

The performance of unsupervised clustering and supervised deep segmentation methods were analyzed along with a discussion of their trade-offs.

Chapter 8

Finally, the main outcomes of this dissertation are summarized in this chapter. The chapter ends with outlines for future work.

2 Synthetic Aperture Radar

The word radar is an abbreviation for radio detection and ranging, suggesting its main purpose is to measure distance. A radar is an instrument that emits electromagnetic waves and receives the returned signals (echoes) from any objects (targets) that bounce off along the propagation path. By measuring the time delay between transmitting and receiving the reflected signal the distance between the sensor and the object can be calculated. Modern radars are a more sophisticated transducer/computer system that is also used to track, identify, image, and classify targets [20].

When using radar to develop an image with two dimensions, some geometric constraints must be enforced to prevent ambiguities in distinguishing objects. Mainly, if the radar instrument is pointed at the nadir (the direction pointing below the airborne/spaceborne vehicle), points located near the vertical will have the same range from the radar. This also includes points at the same range, but opposite direction. Therefore, for imaging radars, the instrument must be side-looking so that the ground distance of a point can be sorted as a function of its distance from the radar.

The range R can be calculated by measuring the time t it takes for the transmitted pulse to travel to the target and return, considering that electromagnetic wave propagates at the speed of light c, then

$$R = \frac{ct}{2}. \tag{2.1}$$

In this chapter, a short introduction is laid out on how an image is formed in SAR, types of acquisition modes, and how to interpret a SAR image, which will be important in subsequent chapters.

2.1 History of SAR

The development of SAR systems was motivated by the need for an all-weather aerial remote surveillance device. Radar becomes the sensible choice, due to its ability to penetrate fogs and clouds, without the need of visible light. However, to obtain sufficient resolution, the antenna would need to be the size of a football field, making it impossible to carry on a reconnaissance aircraft. In the early 1950s, Carl Wiley discovered the use of Doppler frequency analysis could improve the image resolution of a side-looking radar. It led to the development of the SAR technique, which mainly focuses on the signal processing in the azimuth dimension

to obtain higher resolution by synthesizing an aperture that is longer than the actual physical antenna.

The National Oceanic and Atmospheric Administration (NOAA) and engineers from the Jet Propulsion Lab (JPL) explored the use of SAR for oceanic observations. SAR's wavelength is sensitive to small surface changes which makes it suitable for monitoring surface wave patterns and currents. This led to the launch of Seasat in 1978, which marked the first use of SAR in civilian applications. Prior to Seasat, earth observations were performed via Landsat optical cameras. Seasat stopped operating later in the same year due to a short circuit in its power system. In 1991, the launch of ERS-1 by the European Space Agency was the first in a series of orbital SAR satellites aimed at providing long-term earth observation data. This was later followed by JERS-1, ERS-2, Radarsat and Envisat [21].

2.1.1 Radar Equation

The interaction between the incident wave and a target is expressed by the radar equation. The power received by the radar antenna P_R can be expressed as

$$P_R = \frac{P_S \cdot G^2 \cdot \lambda^2}{(4\pi)^3 \cdot R^4} \cdot \sigma, \tag{2.2}$$

where P_R depends on the power of the sender P_S, the antenna gain G, the wavelength λ, the distance between the antenna and the target R, and the radar cross section (RCS) σ

During the interaction of the transmitted electromagnetic wave with a potential target, part of the energy carried by the incident wave is absorbed by the target while the rest is reradiated as a new electromagnetic wave and modulated with the properties of the target [22]. To characterize the target property in terms of power exchange, the concept of RCS σ was introduced in [23] for point targets, described as the ratio of the energy scattered back from the target $\overrightarrow{E_s}$ and the energy intercepted by the target $\overrightarrow{E_i}$ and expressed as an area in m^2

$$\sigma = 4\pi \cdot \frac{\left|\overrightarrow{E_s}\right|^2}{\left|\overrightarrow{E_i}\right|^2}. \tag{2.3}$$

For extended or distributed targets, which occupies space larger than the radar footprint, the concept of backscattering coefficient σ^0 is defined as the RCS per unit area A which makes it have no dimension

$$\sigma^0 = \frac{\sigma}{A}. \tag{2.4}$$

2.1.2 Side-Looking Airborne Radar

The early technology for imaging radars is the Side-Looking Airborne Radar (SLAR) mounted on a plane at height H. To form a 2D image, the echoes received from the ground are sorted by their arrival time in both range and azimuth direction. The radar resolution describes the ability of a radar to distinguish nearby targets. In the range dimension, objects at different ranges can be distinguished if they are separated farther than half the transmitted pulse length T_p. Therefore, the range resolution ρ_R of SLAR is

$$\rho_R = cT_p/2. \tag{2.5}$$

However, this is the slanted range dimension, which is perpendicular to the pulse direction. To measure objects on the ground surface, the ground range resolution ρ_G is

$$\rho_G = \rho_R / \sin\theta_i\,. \tag{2.6}$$

It means ρ_G is not constant as it is a function of the look angle or the incidence angle θ_i which varies from the near range (start of the swath closest to the radar) to the far range, as shown in Figure 2.1.

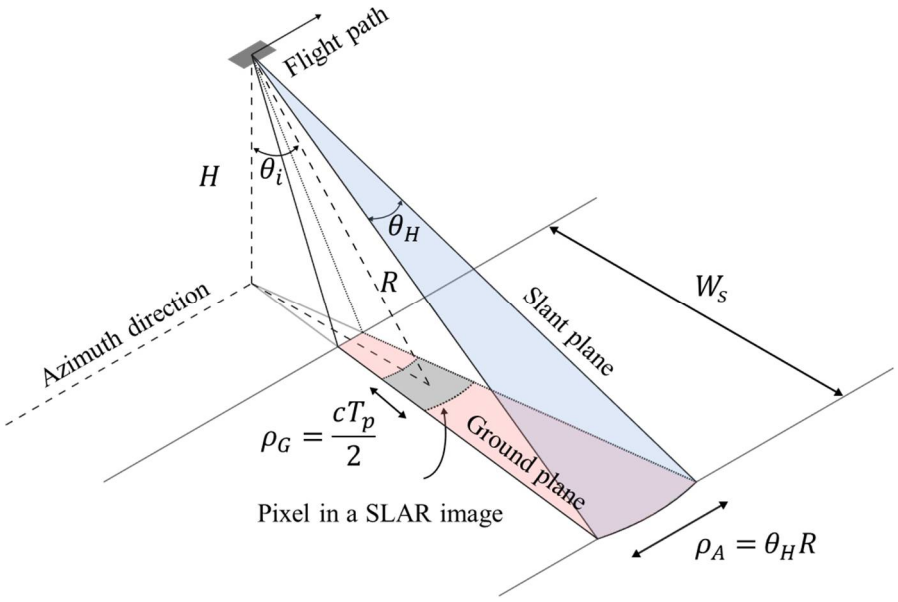


Figure 2.1 Geometry of a SLAR system.

The resolution in the azimuth direction ρ_A is defined by the width of the antenna footprint in the azimuth direction, which is

$$\rho_A = \lambda R/L, \tag{2.7}$$

where R is the slant distance between the radar and the footprint on the ground, which again varies over the swath width. This consequently imposes limitations on the SLAR system. First,

ρ_A being not constant across different range values, and second, the dependency on distance means high-altitude vehicles are impractical due to the poorer resolution. Alternatively, the antenna length L could be increased but would still require an impractical size to compensate for the large distance. The synthetic aperture radar would later solve this issue.

2.2 SAR Image Generation

A SAR system carried by an airplane or satellite is shown in Figure 2.2. The radar antenna moves with the vehicle along a flight path (azimuth direction) and is oriented parallel to the flight direction and looking sideways to the ground (range direction). The antenna, typically a phased array, has a dimension of length L and width W and moves with the vehicle along its flight path above the earth with height H at velocity V. On the ground, the surface area where the radar pulse is reflected is called the footprint. The whole surface area covered by consecutive radar pulses is called the swath. The radar transmits short pulses with duration T_p, which is repeated with pulse repetition interval (PRI) similar to the period of the signal $T = 1/f_r$, where f_r is the pulse repetition frequency.

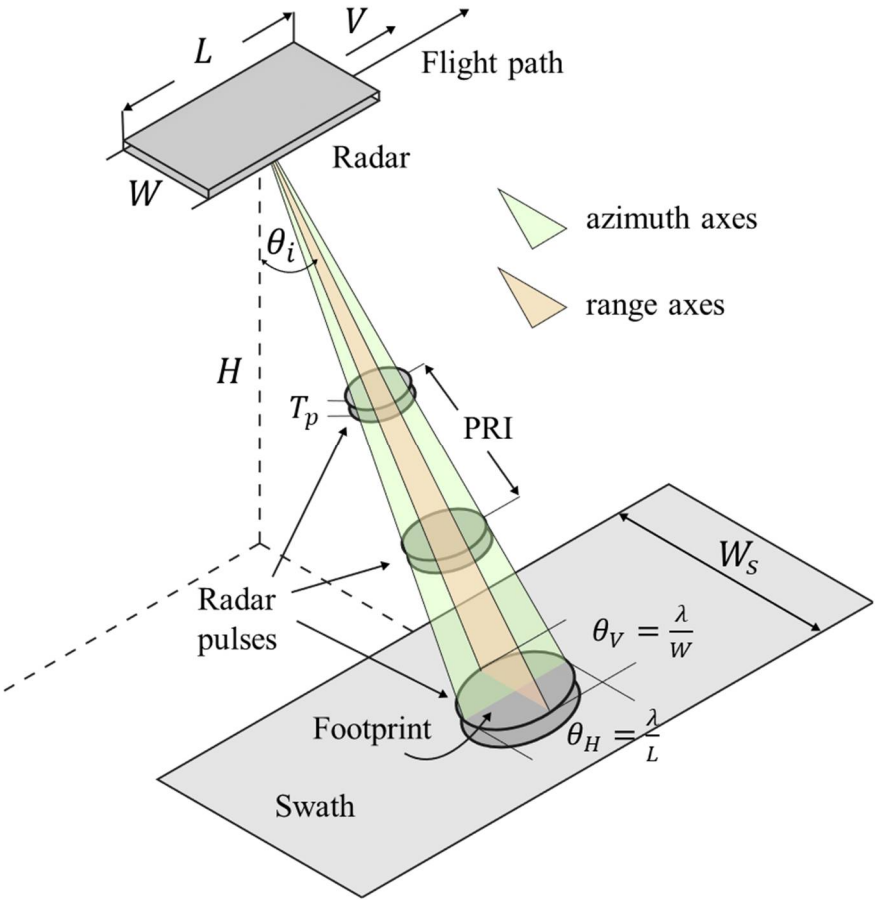


Figure 2.2 SAR imaging geometry.

Once the transmitted pulse is emitted by the radar, a sufficient length of time must elapse to allow any echo signals to return and be detected before the next pulse may be transmitted. Therefore, the rate at which the pulses may be transmitted is determined by the longest range at which targets are expected. Otherwise, the echo might be measured after the next pulse was emitted, resulting in a range ambiguity [23].

The beamwidth in azimuth axes is $\theta_H = \lambda/L$, while the beamwidth in the range axes is $\theta_V = \lambda/W$. The pulse is directed at some angle off the nadir (the direction pointing below the vehicle) called the look angle or the incident angle θ_i. The distance from the radar to the center of the radar footprint is measured by R_0.

The radar pulses illuminate the ground target numerous times, with the time duration of illumination ΔT depends on the beamwidth of the radar antenna and the speed of the flying vehicle, described as

$$\Delta T = \frac{R_0 \theta_H}{V}. \tag{2.8}$$

The radar image is formed by processing the 2D raw data collected by the target echo returned from every radar pulse, transmitted at the rate of f_r. The 2D radar image data are represented in complex numbers and normally can be processed separately by processing range data first, followed by azimuth data.

2.2.1 Range Dimension

Like in SLAR systems, objects or scatterers at different ranges are distinguished based on the time of arrival of their echo. The reception period is limited, thus, determining the minimum and maximum range of the swath width W_s. As with (2.5), a narrower pulse means a better resolving capability between two-point targets. However, over great distances, the energy carried by narrow pulses is reduced, lowering the sensitivity of the radar for weaker targets. The solution is to transmit a pulse with the frequency swept linearly with time, referred to as a chirp.

The advantage of transmitting the chirp waveform is that, on reception, it can be compared against a replica of itself using the operation of correlation, the result of which is a compressed pulse with its center located very precisely in time. Figure 2.3 shows the correlation outcome is a narrow function, which will be used to achieve the range resolution. Using the compressed pulse which has the half power width of $1/B$, the slant range resolution will then be

$$\rho_R = \frac{c}{2B}. \tag{2.9}$$

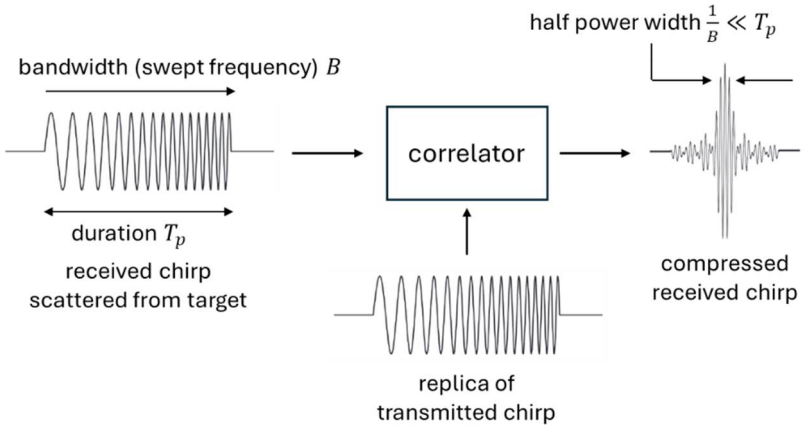


Figure 2.3 Range compression.

2.2.2 Azimuth Dimension

If improvement of ρ_R is achieved by using pulse compression technique, the improvement of ρ_A takes advantage of the synthetic aperture concept. Figure 2.4 shows the SAR imaging system with M ground targets located along the azimuth dimension or along-track direction represented by the u-axis. Each target has an RCS σ_m, where $m \in \{0,1,2,\dots,M-1\}$. The radar illuminates the target area with the beamwidth θ_H. Let $f_0(u)$ be an ideal target function in the azimuth domain, which identifies a group of M targets located along u-axis

$$f_0(u) = \sum_{m=0}^{M-1} \sigma_m \delta(u - u_m). \tag{2.10}$$

The azimuth processing is based on the phase history of returned signals from the targets, which are located along the u-axis and illuminated under the radar beam. Consider a single target σ, which is under the center beam of the radar (highlighted in green) and therefore has the shortest range R from the radar. The $L_S = R\theta_H$ is then the synthetic aperture length which is equivalent to the length of the along-track footprint [24].

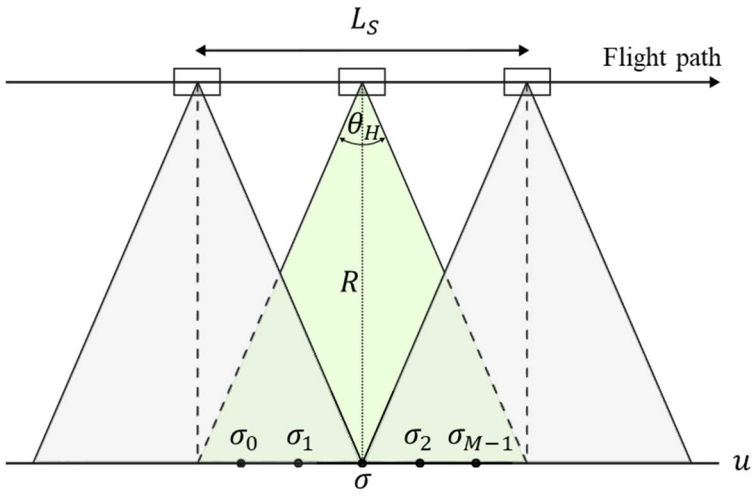


Figure 2.4 A simplified view of the radar pulses sliced in the azimuth dimension.

Since the SAR platform is continuously moving, the echoes returning from objects in the front part of the beam are Doppler shifted to higher frequencies, while echoes from the rear part of the beam are shifted to lower frequencies. This enables the antenna footprint to be divided into bins of equal Doppler shifts (which consequently of equal range). The frequency shift of a moving target is given by

$$f_D = \frac{V_{rel}}{\lambda}, \tag{2.11}$$

with V_{rel} being the relative motion between the source and detector. The frequency history of a signal from a target that has moved through the beam has an approximately linear shift in frequency, where it appears to sweep through a range of frequencies (bandwidth) from high to low. This is known as azimuth compression. The system must therefore be able to transmit consistent (coherent) pulses and accumulate the echo information over successive pulses to allow the synthesis of a virtual antenna that is much longer than the physical antenna.

In this azimuth compression, the ability of the instrument to differentiate signals in time is the inverse of the Doppler bandwidth B_D. To determine the azimuth resolution, the speed of the platform must be considered (rather than the speed of light as used in the range resolution) measured in the coordinate system of the target

$$\rho_A = V/B_D. \tag{2.12}$$

The full Doppler bandwidth is given by [24]

$$B_D = \frac{4\,V\sin(\theta_H/2)}{\lambda}. \tag{2.13}$$

When dealing with units of radians, if θ is small, the sine of the angle is approximately equal to the angle itself. With that and given $\theta_H = \lambda/L$, plugging (2.13) to (2.12)

$$\rho_A = \frac{V\lambda 2L}{4\,V\lambda} = \frac{L}{2}. \tag{2.14}$$

This seems counterintuitive to the expression of resolution from an aperture of length L, but in SAR, the large spread of the Doppler frequencies is utilized rather than the angular beamwidth θ_H. Smaller length antenna will produce a larger beamwidth which in turn increases the Doppler bandwidth given by (2.13).

2.3 SAR Acquisition Modes

2.3.1 Data Acquisition

The 2D radar image is formed by digitizing the received reflected signal at sampling rate f_s. The radar data is arranged in a matrix of complex numbers where each row of data represents one reflected radar pulse, while the column contains information on the same target from successive reflected radar pulses at a constant time interval. Each column of data serves as the along-track dimension of the radar data and is equivalently sampled by the pulse repetition frequency f_r [24].

Figure 2.5 shows the generation of in-phase and quadrature-phase (or I-Q) components. These two signals go through a low pass filter and are digitized to render a complex number pair which is projected as one row of the 2D radar image. The reference signal is dependent on the transmitted signal.

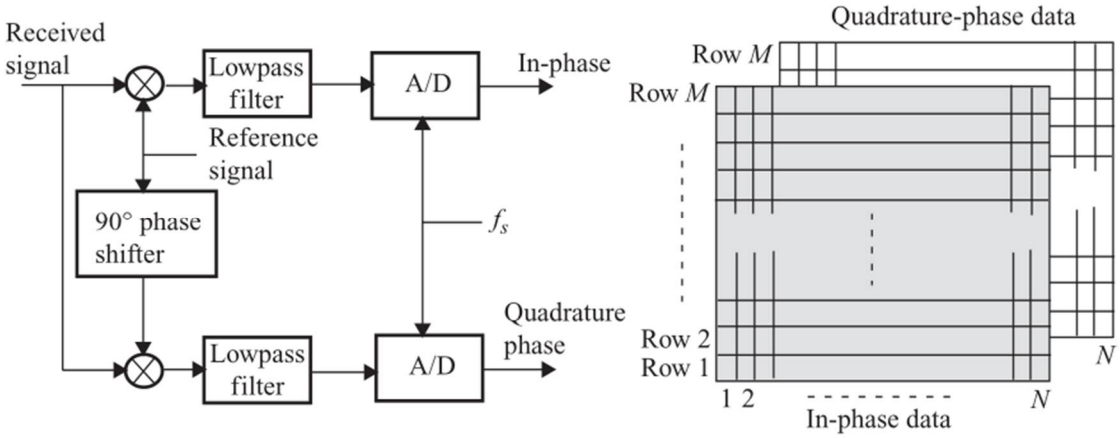


Figure 2.5 Processing of the received signal into In-phase (I) and Quadrature-phase (Q) data [24].

Two points separated by a few meters in the radar range dimension can be distinguished as long as the pulse duration T_p is sufficiently short and the sampling rate f_s of the analog digital converter is sufficiently high. In the range direction, f_s must satisfy the Nyquist requirement of $f_s \geq B$. Along the azimuth direction, since the radar moves with the vehicle at speed V and transmits a pulse at the time interval $PRI = 1/f_r$, then f_s is equal to f_r.

2.3.2 Polarimetry

Polarimetric states are the plane in which the electric field component of the electromagnetic wave oscillates. In radar remote sensing, horizontal (denoted by the subscript H) and vertical (denoted by subscript V) polarized signals are usually used. By switching between polarization states on transmit and receive, the scattering matrix $\boldsymbol{S}$ is obtained which transforms the incident (transmit, i) field vector to the reflected (receive, r) field vector [25].

$$\begin{bmatrix} E_H^r \\ E_V^r \end{bmatrix} = \overbrace{\begin{bmatrix} S_{HH} & S_{HV} \\ S_{VH} & S_{VV} \end{bmatrix}}^{\boldsymbol{S}} \begin{bmatrix} E_H^i \\ E_V^i \end{bmatrix}$$

Most SAR systems use a single antenna to transmit and receive echoes. This is referred to as monostatic sensor configuration, in which due to the reciprocity theorem, the two cross-polarized matrix components are treated as equal $S_{HV} = S_{VH}$. The scattering matrix carries useful information because reflection at object surfaces may change the polarization orientation. A more comprehensive overview of Polarimetry SAR (PolSAR) can be found in [26].

2.3.3 Interferometry

The measured SAR data consists of phase and amplitude within a resolution cell. It is therefore possible to compare the phase differences of two different images of the same region. If the combined SAR images were obtained from slightly different look angles, the relative locations of pixels in three dimensions can be measured, enabling the mapping of the surface topography. This is called a single-pass measurement. If the combined images are from the same position but at different times, the phase changes will indicate movement of surface or deformation between the period of acquisition. This is called a repeat-pass measurement. The produced image is called an interferogram. Phase is affected by interactions with the ground target, the satellite's position in orbit, and topography. These parameters need to be compensated and removed from the final interferogram to reveal the temporal change of the ground surface [27].

2.4 SAR Image Analysis

The basic quantity measured by a single-polarization SAR at each pixel is a pair of signals in the in-phase and quadrature channels. However, several weightings must be applied in the SAR processing to convert the measured voltages to geophysical units that correspond to the complex reflectivity or backscattering coefficient of the scene. Therefore, the measurements made by SAR are fundamentally determined by electromagnetic scattering processes. This means the physical properties of the terrain cause changes in both the phase and amplitude of the wave.

Seeing the SAR log intensity data in Figure 2.6 as an image, several patterns are recognizable, especially for people familiar with the area. The national stadium of Warsaw is visible on the left bottom as shown by bright lines that resemble the dome. Across to the right of the stadium is Park Skaryszewski. Straight lines intersecting in a roundabout at the center of the image are roads. L-shaped patterns on the top are characteristics of buildings.

Figure 2.6 Log Intensity SAR image of the National Stadium in Warsaw, Poland, captured by ICEYE in 2023 (left) [8]. Optical images from Bing Maps Satellite Imagery (right) [28].

For each resolution cell in the image, the in-phase component resembles the real part $A\cos\Phi$ and the quadrature component is the imaginary part $A\sin\Phi$, where A is the amplitude and Φ is the phase. From the complex reflectivity image, other products can be formed to improve the interpretation of the scene, namely the intensity $I = A^2$ and the log intensity $\log I$. The use of the word intensity is synonymous with power or energy. The log intensity image is often referred to as the "dB" image since for calibrated data each pixel corresponds to a linearly scaled estimate of the backscattering coefficient σ^0 in dB. This is done by taking the $10\log_{10}$ of each pixel in the calibrated intensity image. The intensity image has a large dynamic range of values which reduces the perception of detail; therefore, the log intensity is often preferred for visual analysis.

The observed backscatter is a combination of the characteristics of the radar system (frequency, polarization, incidence angle) with the characteristics of the surfaces (roughness, topography, correlation length, dielectric constant).

2.4.1 Surface Roughness

In general, roughness is the height variation inside of a resolution cell. If there are high variations, then the surface is considered rough. In a SAR image, the roughness depends on how large the variation of height is compared to the emitted wavelength. An illustration is shown in Figure 2.7. Leaves are too small to be visible for L-band, therefore, with that long wavelength, forests in L-band appear to be smoother than in X-band. A smooth surface will appear darker in a SAR image.

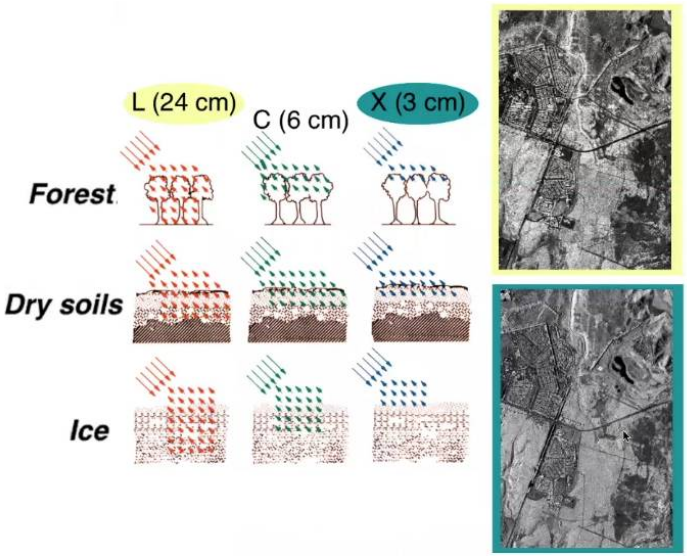


Figure 2.7 Sample images of the same area captured using L-band (top), and X-band (bottom). Illustration source from [29]. It's visible that the top image has more contrast as shown by the darker forest area. This is due to the short X-band wavelength being reflected by the tree canopies.

2.4.2 Speckle

The grainy noise-like patterns shown in Figure 2.6 are characteristics of images produced by coherent imaging systems (also lasers and sonars) known as speckle. It is important to note that speckle itself is not a noise, but the actual measurements of electromagnetic scattering. Consider a distributed target where each resolution cell contains a number of discrete scatterers. Each scatterer illuminated by the beam produces a backscattered wave with a phase and amplitude change, and their summation will be recorded as the cell's measurement. This is illustrated in Figure 2.8.

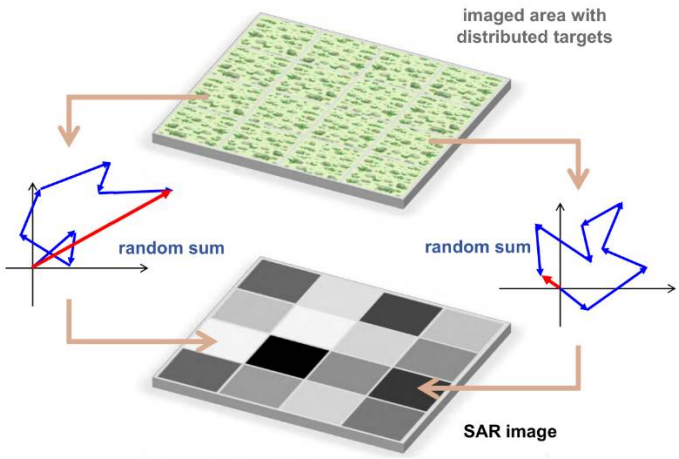


Figure 2.8 The speckle effect from radar illumination is the combination of elementary contributions within a resolution cell [30]. On the right shows the radiometric distribution of the resulting SAR image.

2.4.3 Geometric Distortions

Due to the oblique observation geometry inherent to all imaging radar systems, surface slopes, and similar terrain features lead to geometric distortions in data acquired by SAR systems. The most relevant of these distortions are foreshortening, layover, and shadow. Figure 2.9 shows the city of Islahiye, Turkey. On the left is a hill spanning from the top to the bottom of the image. The Capella Space radar is looking to the left. Foreshortening appears on the slope of the hill facing the radar where the distance between the peak of the hill and the end of the slope will be shortened. Consequently, buildings at the end of the front facing slope also experienced geometric distortions. The amount of foreshortening depends on the incidence angle and the slope angle. In areas where the incidence angle is lower than the slope angle, the top of the structure will be imaged ahead of the base, and a layover will occur. This is commonly visualized in tall buildings, where the walls of the buildings that face the sensor will be projected to the ground in the direction of the radar. The area behind a large structure that is not penetrated by the radar causes shadows. Effects of foreshortening and layover in mountainous areas can be reduced by increasing the incidence angle, but doing so will also worsen the shadow effects.

Figure 2.9 Foreshortening, layover, and shadow effect of buildings in the mountainous area of the city Islahiye, Turkey. Captured by Capella Space in 2023 [6]. Note that on the right of the image are also visible many layover and shadow effects from buildings.

3 Deep Learning

Deep Learning (DL) is a subset of Machine Learning (ML) which both are predicated on the idea of learning from examples. Artificial intelligence (AI) is a more general field that encompasses ML and DL. The field of AI can be described as the effort to automate intellectual tasks normally performed by humans, which also includes many more approaches that do not involve learning [31]. Their relation is shown in Figure 3.1.

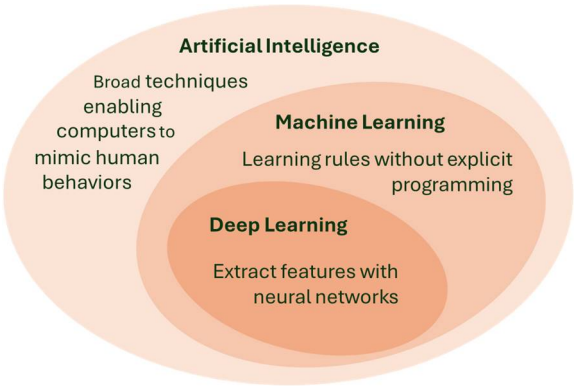


Figure 3.1 The relationship between AI, ML, and DL.

In ML, instead of programming a computer with a massive list of rules to solve a problem, a model is given data with which it can evaluate examples, and a small set of instructions to modify the model when it makes a mistake. It can then be expected that over time, a well-suited model would be able to solve the problem with sufficient accuracy. However, ML is sometimes termed shallow learning due to the architecture mostly consisting of single-layered representation with features extracted manually from the data as input. In DL, the model learns meaningful representations from training examples to solve the given task in an end-to-end manner. Mitchell [32] provides the definition of learning as:

> *"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E."*

A short introduction of each aspect will be discussed in the following subsections. Key terms are bolded the first time they are introduced to distinguish them from their literal meanings.

3.1 The Learning Algorithm

3.1.1 The Task

DL can tackle **tasks** which are too difficult to solve with fixed programs written and manually designed by humans. For example, when developing a robot that can walk, then

walking is the identified task. In the research field, DL tasks are usually described in terms of how the system should process an **example**. An example is a collection of features that have been quantitatively measured from some objects or events, e.g. features of an image are values of the pixels in the image. An example is usually represented as a vector $x \in \mathbb{R}^n$ where each entry x_i of the vector is another feature. A collection of examples (sometimes called samples or data points) is called a **dataset**.

Many DL tasks exist in the literature such as regression, classification, natural language processing, and generative modeling. Experiments in this study focus on the classification task. In classification tasks, the system is asked to specify which of the k categories some input belongs to. To solve this, the learning algorithm is usually asked to produce a function $f: \mathbb{R}^n \rightarrow \{1, \dots, k\}$. Classification tasks depending on the label types can be categorized into: Binary (there are two classes, usually called the positive and negative class), Multi-class (there are more than two classes), and Multi-label (the class predictions are not exclusive, meaning each object can belong to more than one class simultaneously).

3.1.2 The Performance Metric

To evaluate the skill of a DL algorithm, a quantitative **measure** of its performance should be designed. The selection of this performance metric is usually specific to the task being carried out by the system. To prepare for real-world scenarios, the algorithm's performance is usually measured on data it has not seen before. This data is called a test set.

The choice of performance measure may seem straightforward and objective, but it is often difficult to choose a metric that corresponds well to the desired behavior of the system. For example, in classifying medical images of malignant and benign tumors, should the system's sensitivity be increased to not miss any malignant samples (which could cost the loss of lives), or be reduced to not have many false predictions (which could relieve many patients from a costly mistreatment)?

3.1.3 The Learning Experience

Learning is the means of attaining the ability to perform a given task. The **experience** in this case is being exposed to an entire dataset. DL algorithms can be broadly categorized as supervised or unsupervised based on what kind of experience they are allowed to have during the learning process.

In supervised learning, the algorithm experiences a dataset containing features where each example of a random vector x is associated with a label or target y and the model tries to estimate $p(y|x)$.

In unsupervised learning the model experiences several examples of a random vector x, then attempts to pick up useful properties of the structure of this dataset, usually by learning the probability distribution that generated this dataset $p(x)$.

3.2 Artificial Neural Networks

The foundational unit of the human brain is the neuron. At its core, the neuron is optimized to receive information from other neurons, process this information in a unique way, and send its result to other cells. Neural networks emerged from this drive for biologically inspired intelligent computing - and went on to become one of the most powerful and useful methods in the field of artificial intelligence. Despite this, deep learning research nowadays takes inspiration from mathematics, statistics, and computer science, rather than neuroscience. Deep Learning is closely associated with Artificial Neural Network (ANN) that consists of multiple layers stacked one after the other. Learning is performed automatically by composing lower-level features and then building up to more complex ones.

3.2.1 The Neuron

An artificial neuron takes in some number of inputs $x_1, x_2, \ldots . x_n$, each multiplied by a specific weight, $w_1, w_2, \ldots, w_n$. These weighted inputs will be summed and produce the logit z of the neuron.

$$z = \sum_{i=0}^{n} w_i x_i \,. \tag{3.1}$$

The logit will then be passed through an activation function f_a, to produce the output of that neuron, which can then be transmitted to other neurons. By reformulating the inputs to a vector $\boldsymbol{x}$, and the weights to a vector $\boldsymbol{w}$, the output of the neuron y is:

$$y = f_a(\boldsymbol{x} \cdot \boldsymbol{w} + b), \tag{3.2}$$

where b is the bias term. This linear neuron can be used to solve simple linear functions. In the real-world, data is generally not linearly separable. Therefore, to learn more complex relationships (as shown in Figure 3.2), nonlinearity needs to be introduced to the neurons. These non-linear transformations are called activation functions.

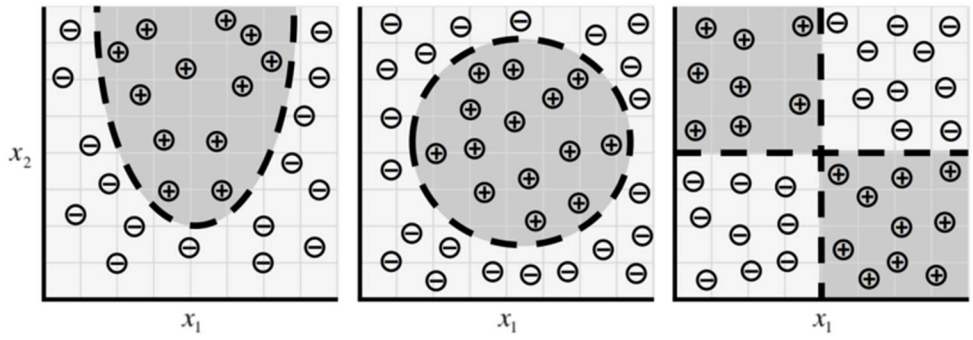
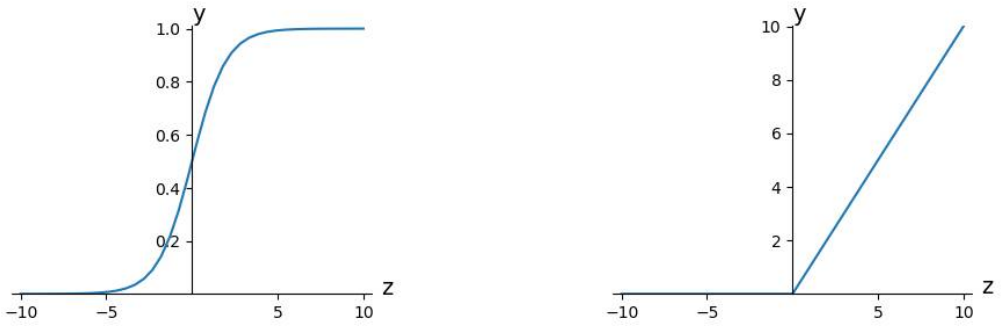


Figure 3.2 As the data takes more complex forms, more complex models are needed to describe them [33].

An example is the sigmoid function shown in Figure 3.3a, which scales the logit z to the value between 0 and 1. If $z = 0$ the output is exactly 0.5, showing a characteristic S-shape. It uses the function

$$f_a(z) = \frac{1}{1 + e^{-z}}. \tag{3.3}$$



a) b)

Figure 3.3 The output y of nonlinear activation functions: a) sigmoid and b) ReLU as the logits z varies.

The rectified linear unit (ReLU) is typically the default choice of activation function in modern neural networks since it is a function with two linear pieces [34]. ReLUs preserve the many properties that make linear models generalize well and are easy to optimize with gradient-based methods [35]. ReLU uses the function $f_a(z) = \max(0, z)$.

3.2.2 A Network of Neurons

The simplest form of ANN, shown in Figure 3.4, consists of three layers of neurons, the input layer, the hidden layer, and the output layer. Information flows from the input to the output. Each connection has a weight and a bias with a respective activation function. A neural network can be measured by its capacity, which is the total number of weights and biases (both considered as parameters of the network), or by its depth, which is the total number of layers.

When a network is trained using supervised learning, it is provided with an input and an output. The input is fed through the network and the parameters will shape the input to some degree and then pass it to the next layer.

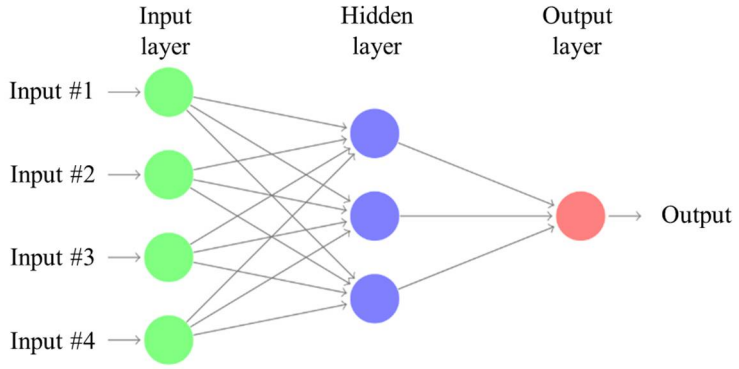


Figure 3.4 A basic neural network with three layers.

After reaching the output layer, the calculated value is compared with the original output and an error tells how far they are. A special function called the loss function (sometimes called the cost function or the error function) will determine how the difference in output is calculated.

For a deep learning model, the architecture refers to how each layer in the network is connected, while the backbone refers to the feature extraction part of the model. Neurons from one layer are partially or fully connected to neurons from adjacent layers. When connections from far apart layers are made, these are called skip connections. Every connection will have a parameter that adjusts its value, called weights. These weights are also termed as the parameters of the network and are used to indicate its capacity.

3.2.3 Gradient Descent

To obtain optimal value for the weights of a network, optimization is needed to maximize the performance of the model by iteratively tweaking its parameters until the error is minimized. Gradient descent is a key technique for optimizing nearly any deep learning model [34]. It consists of iteratively reducing the error by updating the parameters of the model in the direction that incrementally lowers the loss function. Suppose a simple linear neuron with a single input with the weight w, shown in Figure 3.5. The error function $\mathcal{L}(w)$ is obtained by considering the error over all possible values of w. Without knowing $\mathcal{L}(w)$, the error can still be minimized by taking the gradient at the current position. For example, at position A, there is a negative slope, indicating w_A should be increased to minimize the error. Likewise, at position B, the positive slope indicates w_B should be decreased.

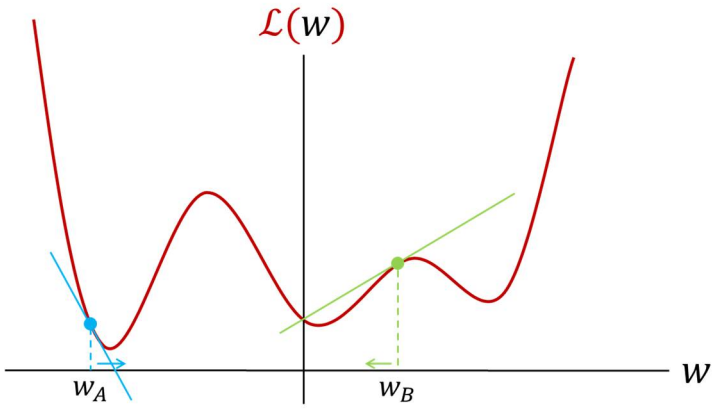


Figure 3.5 Visualizing the gradient descent algorithm for a simple neuron with a single parameter w.

The gradient indicates the direction of minima (think of it as a valley where the error is lower compared to nearby points). How large the step depends on the steepness of the slope. However, in certain cases, the cost function can be rather mellow, which can potentially prolong the training time. Therefore, the gradient is typically multiplied by a hyperparameter called the learning rate α. Selecting the right learning rate is a tradeoff between speed and accuracy. A small value will risk a long training time, while a value too large will risk skipping away from a minimum.

To calculate the change for each weight, the gradients are evaluated by taking the partial derivative of the error function with respect to each of the weights from the k-th layer

$$\Delta w_k = -\alpha \frac{\partial \mathcal{L}}{\partial w_k}. \tag{3.4}$$

Applying this algorithm to the whole dataset (called the batch gradient descent) of examples can sometimes lead to problems. The error curve might be a flat line (in high-dimensional space it is known as a saddle point) which might lead to premature convergence. The solution to this is by estimating the error with respect to a subset of examples, which results in a dynamic error curve that potentially could help navigate through flat regions. This is called the mini-batch gradient descent. The number of examples in this subset is called the minibatch size (or sometimes just batch size) and is another hyperparameter for training [31].

3.2.4 Information Theory

Information theory deals with the problem of encoding, decoding, transmitting, and manipulating information in a form as concise as possible. The central idea of information

theory is to quantify the information content in data. This quantity is called the entropy $\mathcal{H}$ of a distribution P, and is described by the Shannon entropy

$$\mathcal{H}(P) = \sum_{x} -P(x) \log_{\varkappa} P(x), \tag{3.5}$$

where $\varkappa$ is the base of the logarithm. One of the fundamental theorems of information theory states that in order to encode data drawn randomly from the distribution P, at least $\mathcal{H}(P)$ nats are needed to encode it. When using the natural logarithm with base e, one nat is the amount of information gained by observing an event of probability $1/e$. In other fields, the base-2 logarithms are used and therefore the units are called bits, which is essentially a rescaling of information measured in nats [34].

Shannon entropy of a distribution is the expected amount of information in an event drawn from that distribution. It gives a lower bound on the number of bits (if the logarithm is base 2) needed on average to encode symbols drawn from a probability distribution P. Distributions that are nearly deterministic (where the outcome is nearly certain) have low entropy; distributions that are closer to uniform (e.g. $p = 0.5$ for a binary random variable) have high entropy.

Given two separate probability distributions $P(x)$ and $Q(x)$ over the same random variable X, the difference between these two distributions can be measured using the Kullback-Leibler (KL) divergence:

$$D_{KL}(P||Q) = \sum_{x} P(x) \log \frac{P(x)}{Q(x)}. \tag{3.6}$$

For discrete variables, the KL divergence is the extra amount of information needed to send a message containing symbols drawn from a probability distribution P, when using a code that was designed to minimize the length of messages drawn from a probability distribution Q. The KL divergence has useful properties: it is non-negative, and it will be 0 if and only if P and Q are the same distribution in the case of discrete variables. Due to these properties, KL divergence is often used to measure the distance between two distributions.

A quantity closely related to KL divergence is the cross-entropy

$$C_E(P, Q) = H(P) + D_{KL}(P||Q). \tag{3.7}$$

Cross-entropy C_E calculates the number of bits required to represent or transmit an average event from one distribution compared to another distribution. The result will be a positive

number measured in bits and will be equal to the entropy of the distribution if the two probability distributions are identical.

In supervised classification tasks, cross-entropy is useful for optimizing the model. Each example has a known class label with a probability of 1.0 and a probability of 0.0 for all other labels. The model then estimates the probability of an example belonging to each class label. The difference between the true probability distribution and the predicted probability distributions can then be calculated using cross-entropy [36].

3.3 Convolutional Neural Network

Computer vision is a research field that explores automated methods for understanding the world like human vision. Traditionally, features need to be extracted manually from images to improve the signal-to-noise ratio. Features such as edges that form shapes, and patterns of light and dark patches are lower-dimensional representations of the input image, which are then used by a Machine Learning classifier to make predictions. There are fundamental limitations of this approach, which does not handle well slight variations of the input, such as different light intensity, or slight occlusion.

Among the early publications of CNNs was LeNet in 1998 [37], introduced by Yann LeCun and his team to solve the recognition of handwritten digits. Modern CNN was then popularized by Alex Krizhevsky and Geoffrey Hinton [38] who in 2012 won the image classification challenge called ImageNet [39] and improved the error rate by a large margin.

3.3.1 The convolution layer

Dense layers learn global patterns in their input feature space (for example, for an image of a digit, it treats all the pixels as features), whereas convolution layers learn local patterns—in the case of images, patterns found in small 2D windows of the inputs. This gives Convolutional Networks two interesting properties:

- Learning translation-invariant of patterns. For example, the specific pattern can exist at any location in the image, rotated or flipped, and will still be recognized.
- Learning spatial hierarchies of patterns. The first convolution layers will learn low-level features such as edges, which the next layer will use as input. The deeper through the network, the more complex and abstract visual concepts can be learned. Ultimately, the final hidden layer learns a compact representation of the image that summarizes its contents such that data belonging to different categories can be easily separated.

The convolution layer operates over a 3D array usually called feature maps. It consists of the spatial axes (width and height) and the depth axes (also called the channels). For a natural RGB image, the depth will be 3 corresponding to the number of color channels: red, green, and blue. Once the input RGB image is processed, the depth no longer corresponds to the color dimension, but they rather stand for filters. For each convolution layer, there are two hyperparameters: the kernel size, which determines the size of the patch extracted from the input (e.g. a 3x3 filter), and the output depth of the output feature map, which is the number of filters computed using the convolution operation.

A convolution works by sliding these filters over the 3D input feature map (dimensions: height, width, depth). Each 3D patch will result in a 1D vector (dimension: output depth) which is then reassembled to its corresponding patch location. The effectiveness of CNNs is to reduce spatial dimensions (which reduces computation but maintains coverage of the entire image) while increasing the number of feature maps (which increases the probability of recognizing salient features). This downsampling can be done in several ways:

- The sliding window operates on a chosen tile, which is the center of the window, and its neighbors. Unless the filter size is 1x1, the patch outside the range of the input feature map cannot be selected, therefore it reduces the spatial dimension. For example, consider a 4x4 input feature map which will be convolved with a 3x3 filter. There are only 4 possible tile locations to center a 3x3 window, resulting in a 2x2 output feature map. To produce the output feature map of the same size as the input, padding can be used to extend the input to fit center convolutional windows around every input tile.
- Strides are another hyperparameter that affects the output size. Stride is the distance between the center tiles of each sliding window. Using a stride of 2 for example, will result in a downsampling by a factor of 2.
- Pooling layers can be used to aggressively downsample feature maps. Unlike strided convolutions (using a stride >1), the pooling operation does not have a learnable parameter, therefore reducing computation. Pooling works similarly to convolution where a sliding window is applied to the feature map, but the output is simply a reduced operation of the patch, e.g. maximum, average, or minimum.

3.3.2 Segmentation Models

Segmentation is the task of partitioning the image into regions based on the similarity (alike characteristics within the same class) and discontinuity (the border or edge between different

classes) of pixel intensity. For binary class segmentation such as building footprint extraction, there are only 2 classes: the positive examples, i.e., pixels belonging to a building's region, and negative examples, which are the rest of the pixels (non-building). The segmentation model is given a pair of image inputs and semantic labels for training. The iterative process outputs a single channel similar-sized image classifying each pixel as belonging to one of the classes. In multi-class segmentation (illustrated in Figure 3.6), typically the output is an image with the number of channels corresponding to the number of classes.

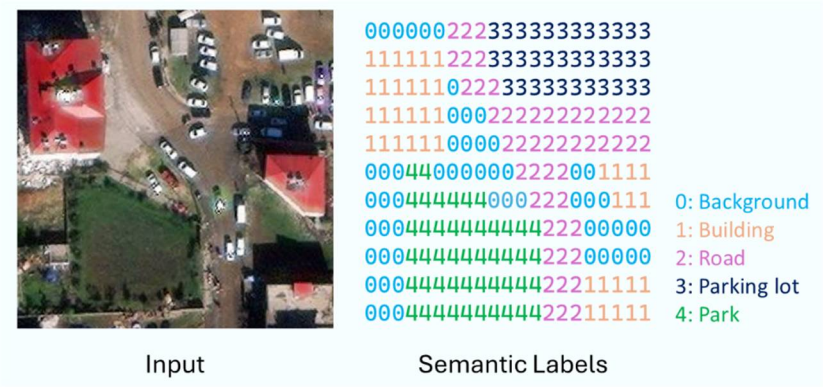


Figure 3.6 Illustration of multi class segmentation. Satellite image from Maxar [5].

For remote sensing tasks, UNet [40], DeepLab v3 [41], PSPNet [42], and Feature Pyramid Network (FPN) [43] are commonly used as network architectures. These are categorized as encoder-decoder type architectures, which are commonly used for segmentation tasks. The encoder part is usually called the backbone and acts as the feature extractor. Popular CNN architectures benchmarked on the ImageNet dataset are typically used as backbones such as [44], Inception [45], and EfficientNet [46]. Each backbone also has a range of capacity (measured in the number of weights/parameters) to allow flexibility for optimizing performance or inference speed.

Despite different network configurations, the way data propagates in a segmentation model is typically similar. A review of segmentation architectures and backbones can be found in [47], [48]. Below is a brief explanation of the FPN architecture (shown in Figure 3.7), used in previous studies [15], [49], and in chapters 5 and 7 of this dissertation.

FPN utilizes feature pyramids, which aim to better capture multi-sized objects by using multi-scale input. In the encoder, the image input is scaled down using a convolution operation with a stride of two (red arrows), which cuts the image dimension in half at each pyramid level. As the data flows up the pyramid, the top layer will have the least width and height (the original input's size divided by 32) but the richest semantic information (1632 feature maps or

channels). In a classification task, this is compressed further to output a vector with the same size as the number of classification labels [43]. For a segmentation task, an output with the same spatial size as the input is required. Therefore, the top layer needs to be upscaled.

A 1 × 1 convolution filter is applied to the final layer in the encoder pyramid to reduce the number of feature maps to 256, without modifying the image dimension. As data flows down the decoder's pyramid, the width and height increase twice using nearest neighbors upsampling (green arrows). In the skip connections (yellow arrows), feature maps from the same pyramid level in the encoder and decoder were concatenated. A 1 × 1 convolution was used to scale the feature maps from the encoder pyramid to 256. This provides context for better localization as the image gradually recovers in pixel resolution. Afterward, feature pyramids from the decoder go through a Conv and Upsample operation (black arrows), resulting in modules with 128 feature maps and image dimension 1/4 of the original input. These are then stacked channel-wise, creating a module of 512 feature maps. A final Conv and Upsample operation reduces the number of channels to 1 and restores the image dimension back to the original input [50].

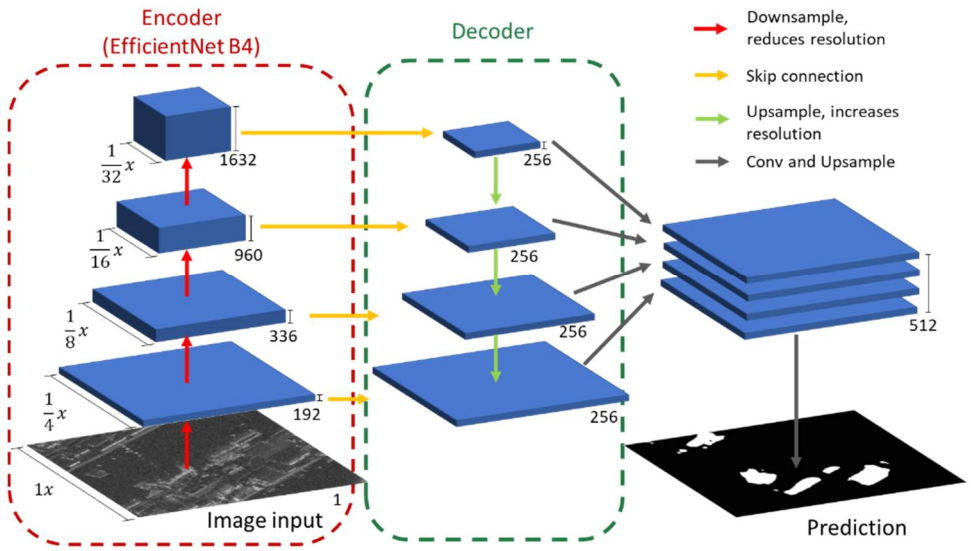


Figure 3.7 Network configuration of FPN, a type of encoder-decoder architecture for segmentation tasks, illustrated for segmenting buildings in SAR image (binary class). EfficientNet backbone was used as the encoder.

3.4 Evaluation Metrics

The most common method to evaluate classification performance is computing the accuracy by comparing predictions with their ground truth. However, counting the number of correct matches alone does not usually work well in real-world problems due to class imbalance [51].

To give better insights on how the model distinguishes positive and negative classes in a binary classification problem (see Figure 3.8), the predictions can be categorized as:

- True positive (TP): the model correctly predicted the positive class.
- False positive (FP): the model incorrectly predicted a positive class (actual class is negative). This is also known as type I errors.
- False negative (FN): the model incorrectly predicted a negative class (actual class is positive). This is also known as type II errors.
- True negative (TN): the model correctly predicted the negative class.

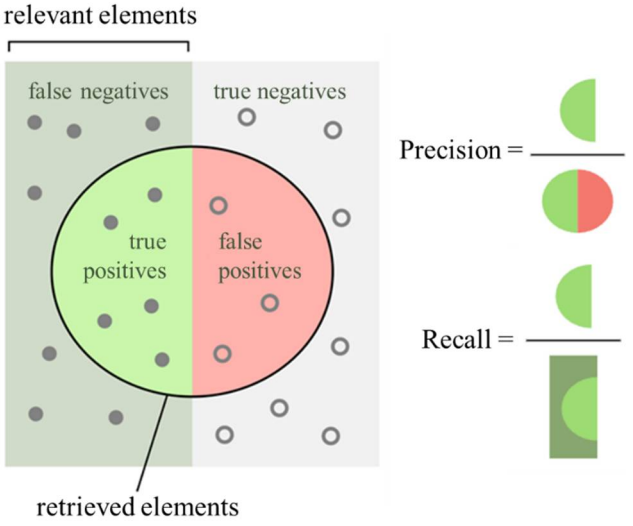


Figure 3.8 Recall and precision in binary classification [52].

The two types of errors conflict with each other, as minimizing one might increase the likelihood of the other occurring. This optimization depends on the specific problem and its consequences. To balance the trade-off, the following classification metrics are often used:

- Recall: in the context of binary classification, recall is the same metric as sensitivity or True Positive Rate. Consider the recall metric when prioritizing to minimize false negatives (e.g., in cancer diagnosis)

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{3.8}$$

- Precision: also called the positive predictive value, it describes the portion of predicted positives that are correctly classified

$$\text{precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})}. \tag{3.9}$$

Usually, precision and recall scores are not reported in isolation. Instead, values for one are compared at a fixed value of the other (e.g., precision at recall of 0.5). They commonly exhibit an inverse relationship where it's possible to increase one at the cost of the other. The F_1 score is the harmonic mean of precision and recall, therefore symmetrically represents both in one metric. It is defined as

$$F_1 = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}. \tag{3.10}$$

4 Deep Learning for Building Unit Damage Assessment using SAR: Progress and Challenges

4.1 Introduction

Identifying building damage after the occurrence of large-scale natural disasters can optimize logistics and resources to prioritize areas with higher concentrations of damage. For example, during the 2023 Syria-Turkey earthquake, 50 thousand people were found trapped under the debris of collapsed buildings [2]. Instead of a costly and risky ground survey, remote sensing images such as SAR can cover large areas allowing a quick and comprehensive overview of the affected areas.

Optical and SAR images are typically the main choice for BDA post-disaster. Optical images show the earth's surface similar to how humans see, therefore being easier to interpret. However, SAR is more reliable in providing measurements owing to its active sensors that can penetrate through clouds and independent of sunlight. Despite the radar images being less intuitive than optical images, in the event of a large-scale natural disaster, any data to help emergency response is valuable. In addition, SAR can penetrate through clouds, fog, smoke, and dust, which typically occludes a scene after a disaster. With the many constellations of small SAR satellites [53], collecting post-disaster data will be trivial.

For rapid damage assessment based on SAR data, there are mainly three techniques involved: first, by analyzing a single post-event VHR SAR data, second, by doing change detection of pre and post-event data pair from the same sensor, and last, by combining optical and post-event SAR data [54]. The disturbance of linear characteristics of buildings in radar images can reflect the levels of damage, mainly due to loss of elevation. Debris from damaged buildings also shows as strong scatterers. Manual interpretation of damage from SAR should consider the orientation of the building, the geometric features of the SAR system (look angle and direction) and the surrounding environment of the target building [54].

An excellent classification of BDA methods using SAR was reviewed in [55]. Methods were classified based on the unit of analysis (block units or building units), chosen SAR feature for analysis (intensity, polarimetric, interferometry or coherence), and availability of pre-event data (only post-event analysis or change detection using both pre- and post-event).

The surge in neural networks success paves the way for DL based approaches in remote sensing and signal processing. Neural networks are powerful feature extractors, utilizing the

abundance of data to learn hidden patterns from labels. Inspired by the works in [55], this review aims to cover a more specific aspect of SAR BDA using DL algorithms.

First, the interpretation of building damage in radar images was explained using common SAR features: intensity, coherence, and polarimetry. Then, the state-of-the-art DL methods for BDA were reviewed, starting with optical-based BDA, then building detection in SAR, and ending with SAR-based BDA. Third, open-source SAR data for disaster analysis was reviewed. Finally, the challenges and opportunities for DL-based SAR BDA were discussed.

The main contributions of this review chapter are the following:

- Summary of state-of-the-art DL-based methods for building-unit damage assessment.
- Review of open-source data for disaster analysis using SAR and DL.
- Challenges and opportunities for DL-based approaches for SAR BDA.

4.2 Physical interpretations of building damage in SAR

SAR data describes the radar reflectivity of the scene. The physical property of the terrain causes changes in both phase ϕ and amplitude A of the scattered electromagnetic wave. Depending on the acquisition mode of spaceborne or airborne SAR, the intensity, phase, and polarimetry features can be used to identify building damages.

4.2.1 Intensity

Intensity refers to the mean amplitude of the recorded backscatter that is influenced by operating parameters of the radar system, such as incidence angle and wavelength, and the characteristics of ground targets, such as their dielectric properties and roughness [55]. Generally, an intact building will show a layover pattern towards the sensor and cast shadows in the opposite direction. Typically a strong L-shaped pattern emerges, showing the echoes formed by the wall and the ground [56]. When a building collapses, the layover, the L-shaped pattern, and even shadow patterns might disappear or decrease, creating a more random pattern. Two different building conditions are highlighted in Figure 4.1, the building on the right, despite showing some debris, is still standing. Meanwhile, the building on the left was destroyed. The radar patterns are recognizably different, where the intact building still showed layover towards the right of the image, an L-shaped pattern, and shadow. While the destroyed building showed random patterns from the remaining debris.

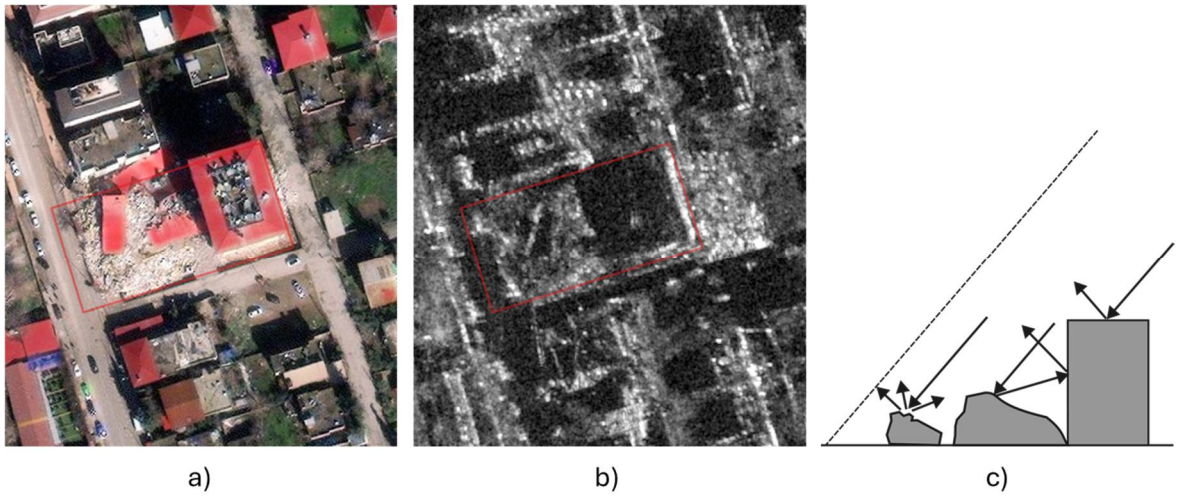


Figure 4.1 Image of a collapsed building next to an intact building in a) optical images from Maxar [5] and b) VHR SAR from Capella space [6]. In c) the collapsed building resulted in an interruption of the linear characteristics in the SAR intensity value.

The difference on absolute value of intensity from bitemporal SAR images can be used to quantify ground changes caused by a disaster [55]. However, only using intensity change is not reliable, since higher change can be observed not only in damaged areas [57]. Features derived from intensity information such as correlation coefficient [58] and texture analysis [59], [60] are typically used to obtain a more accurate detection.

In most cases, suitable archived pre-event SAR images are not available. This leaves only a single post-event SAR image for analysis, which opens a new research topic on extracting features to classify damage grades in buildings. Without polarimetric or interferometric mode, the post-event SAR image will have the best spatial resolution (i.e. spotlight mode), and enough information to identify individual buildings. However, when limited to only a single channel, there is not enough information to obtain the scattering mechanisms of the scene. This prompts research on using data-driven methods such as neural networks to train robust feature extractors for this complex task.

4.2.2 Coherence

When there are two images acquired by the same satellite system and covering the same area at different times, their electromagnetic wave interferences can be exploited. This is the principle of Interferometric SAR (InSAR). The phase in SAR data indicates the relative value of the returned backscattering waves in a full period. It is very sensitive to the distance between the satellite sensor and the ground target and can therefore be applied for ground change detection like shown in Figure 4.2.

Coherence γ indicates the cross-correlation of phase information in two SAR images. It is computed using [61]

$$\gamma = \frac{\mathbb{E}[u_1 u_2^*]}{\sqrt{\mathbb{E}[|u_1|^2]}\sqrt{\mathbb{E}[|u_2|^2]}}. \tag{4.1}$$

where u_1 and u_2 are the amplitude values of image 1 and image 2, u^* denotes the complex conjugate of u, and $\mathbb{E}[x]$ is the expected value of x.

The decorrelation of coherence usually indicates ground changes and can be used to quantify damage in disasters [62], [63]. A technique popularized by NASA's Advanced Rapid Imaging and Analysis (ARIA) project used the difference of coherence between two pre-event SAR image pairs and coherence between a pre-event and a post-even pair. The former is called pre-event coherence while the latter is called co-event coherence. This method can differentiate between areas where coherence is always low and areas where it has decreased, e.g., due to building collapse or landslides [64].

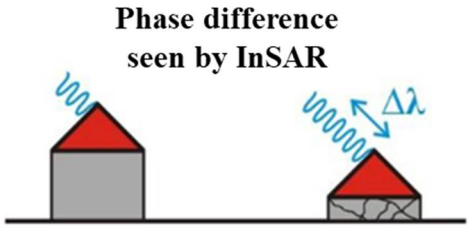


Figure 4.2 Interferometric analysis uses the phase difference between the pre- and post-disaster SAR acquisition enabling the distinction between both buildings.

4.2.3 Polarimetry

A radar system transmits an electromagnetic wave, with a given polarization state, that reaches the scatterer of interest, and then receives the reradiated energy. It is possible to infer some information about the scatterer considering the properties of the scattered electromagnetic wave with respect to the transmitted wave [26]. Electric field vectors of energy pulses emitted by a radar system can either be polarized in a horizontal (H) plane or in a vertical (V) plane. Regardless of wavelength, a SAR platform can transmit H or V electric field vectors, and then receive H or V return signals, yielding four types of polarization: HH, HV, VH, or VH, where the first letter denotes the transmit polarization and the last letter denotes the receiving polarization.

A SAR system can be designed to work in a single polarization, dual polarization, or full (quad) polarization. The polarization features of multi-polarized SAR data are sensitive to

dielectric constants, physical properties, geometry, and the orientation of ground targets. Therefore, they can greatly improve the ability of imaging radar to acquire various information about the targets.

The addition of polarimetric features has been shown to improve coherence change detection. Building damage assessment using Sentinel-1 satellite imagery in combined features of dual polarization VV and VH yields higher accuracy than using either type of polarization [57]. A similar study was found using dual polarization HH and HV from ALOS-2 satellite imagery [65].

Full polarization (also called PolSAR) information can provide richer descriptive features for understanding the scattering mechanisms of ground targets. The full polarization data can be represented as the scattering matrix S [66] represented by

$$S = \begin{bmatrix} S_{HH} & S_{HV} \\ S_{VH} & S_{VV} \end{bmatrix}. \tag{4.2}$$

The scattering matrix can be formed into a Pauli scattering vector k_p with the reciprocity condition applied

$$k_p = \frac{1}{\sqrt{2}}[S_{HH} + S_{VV} S_{HH} - S_{VV} 2S_{HV}]^{\intercal}, \tag{4.3}$$

where $\intercal$ indicates a matrix transpose. The coherency matrix T can be obtained by multiplying k_p with its conjugate transpose k_p^H, given by [26].

$$T = \langle k_p k_p^H \rangle = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix}. \tag{4.4}$$

Various decomposition approaches have been proposed, such as eigenvalue-eigenvector based decomposition [67], the Freeman-Durden decomposition [68], and the four-component scattering model by Yamaguchi [69]. Scattering mechanisms such as the double-bounce, surface, and volume scattering can be derived from polarimetric decomposition as indicators for building damage assessment in disasters [70] [71]. By comparing two pre-event PolSAR images with one post-event PolSAR image, changes in decomposition components can infer a building's condition after a tsunami [72]. Damage in built-up areas can also be described by other polarimetry parameters, such as polarization coherence, which can be used to characterize surface roughness [73], and the polarimetric orientation angle, which has a close relationship with building orientation [74].

In a single post-event PolSAR image, the polarimetry features can still be potential indicators, since damaged and undamaged structures usually show different characteristics in different decomposition components. For instance, intact parallel buildings are usually characterized by double-bounce scattering, whereas collapsed buildings can be characterized by volume scattering [75]. However, undamaged buildings whose orientation is not parallel to the SAR flight pass and the collapsed buildings share similar dominated scattering mechanisms, i.e., volume scattering, creating ambiguity [76].

4.2.4 Unit of analysis

Damage can be assessed in a block unit or building unit, as shown in Figure 4.3. An important factor is the spatial resolution of the SAR image, which is affected by the radar system and imaging mode. Medium-resolution images that are >20 m/pixel (by today's standards) will show a 400 m^2 building as a single resolution cell. This makes it difficult to identify features from individual buildings, therefore, it is more reasonable to use block-unit analysis. Block units can either be a consistent grid or irregular blocks that follow the urban boundaries. For submeter resolution SAR image, edges and texture from each building can be observed, enabling building unit analysis.

Figure 4.3 Example of different unit analysis: (left) Building unit in binary class damage system of intact buildings and destroyed buildings. (right) Block unit in four class damage system. Optical images from Maxar [5].

4.3 State of the art

Solving the task of building unit damage assessment in SAR using DL methods involves the awareness of neighboring research fields that are closely related to the problem.

4.3.1 Building footprint extraction in SAR

Since the unit of analysis is individual buildings, the research field of Building Footprint Extraction (BFE) in SAR comes into inspiration. In 2017, neural networks were used to classify small patches of building areas in [77] resulting in a coarse prediction of built-up areas. Progress was made in 2018 when CNN was demonstrated capable of individual building detection in [78] from a VHR TerraSAR-X image and OpenStreetMap (OSM) building footprints as reference labels. Adriano et al. [79] developed a deep segmentation model for multiple cities, making an important observation of CNN's generalization problem related to local urban scenes. The performance is lower for cities with dense and high-rise buildings such as Shanghai and Hong Kong due to the mixtures of layover patterns from nearby structures. The SpaceNet 6 competition in 2020 [80] provided high-quality data for BFE in SAR. The openly available dataset consists of full polarimetric airborne SAR data with 0.5m spatial resolution, covering the port of Rotterdam, Netherlands. This dataset has been the benchmark in various BFE tasks in SAR or multi-modal SAR and Optical [81].

4.3.2 Optical-based building damage assessment

Classifying damage of individual buildings from satellite images is widely used in crowd-sourced assessments for humanitarian initiatives. Optical images are the main source since it is easier to interpret by non-experts. DL methods are actively proposed to classify damage from optical images, to provide faster and more consistent analysis than crowd-sourced human labelers. The xView2 challenge, launched in 2020 played a major role in advancing DL methods for BDA. The organizers released the xBD dataset which consists of large-scale pairs of before and after optical images of a disaster [82]. The dataset covers five types of disasters from various geographic regions of the world, with damage labels grouped into four levels following the HAZUS method from FEMA [83]. A sample from the dataset is shown in Figure 4.4. From the literature, the Siamese network is used to classify differences from bitemporal RS images and grade the damage. Two-stage networks are also used, involving segmentation tasks using Unet [84] or transformer architecture [85], [86], then a smaller attention network that fuses or aggregates features from different time periods to classify the change. RescueNet was proposed for joint segmentation and damage assessment that can be trained in an end-to-end manner and simplifies multi-task supervision during the training stage [87].

a) Pre-disaster

b) Building damage overlayed on post-disaster

Disaster Level	Structure Description
0 (No Damage)	Undisturbed. No sign of water, structural or shingle damage, or burn marks.
1 (Minor Damage)	Building partially burnt, water surrounding structure, volcanic flow nearby, roof elements missing, or visible cracks.
2 (Major Damage)	Partial wall or roof collapse, encroaching volcanic flow, or surrounded by water/mud.
4 (Destroyed)	Scorched, completely collapsed, partially/completely covered with water/mud, or otherwise no longer present

c) Damage-scale in xBD

Figure 4.4 A sample from the xBD dataset of the 2018 Tsunami in Palu, Indonesia [82], showing a) the pre-disaster image, b) the post-disaster image with the building damage overlaid, and c) the four-level damage scale used for labeling.

DL methods in remote sensing data have the common issue of generalization [88] where different geographic scenes unseen in training exhibit lower performance. This is an issue in emergency response situations where training samples are almost unavailable. Therefore, deep learning models need to be robust to the distribution shift of unseen data [89]. Some studies explored transfer learning approaches for BDA using the xBD dataset [82] for its variety in geographic location and disaster type [90]. A different version of [82] was proposed to simulate real-world scenarios where the testing split is a new type of disaster or location, unseen in the training set [91]. Other studies applied transfer learning for an individual event rather than a group of disasters [92], [93]. In such cases, generalization is difficult to assess since each disaster behaves differently, and a model trained on one disaster may not work equally in a different event. Wiguna et al. [89] conducted an extensive experiment on different network architecture and loss functions for predicting building damage in historical disasters and evaluated them on new unseen events. They conclude that a combination of cross-entropy loss and focal loss yields better scores. Their transformer network achieved state-of-the-art results in xBD with a mean F1 score of 77.79 for all four damage classes. However, transformers have large parameters that take considerable processing resources. From a humanitarian aid viewpoint, the value of automating damage assessment lies in speed improvement rather than pushing marginal improvements in accuracy.

Disaster impact information is good for immediate response, but long-term recovery programs require a combination of other socio-economic data. Therefore, prediction output in the form of geo-referenced vector data is important for analysis. The work from Microsoft lab

proposed a model that is three times faster than the winning solution of xView2 challenge with only a 0.05 reduction in mean F1 scores [94].

4.3.3 Building-unit damage assessment in SAR

Not much research has been proposed for building unit damage assessment in SAR since a very high-resolution is needed to ensure sufficient pixels can describe each building footprint for analysis. Several studies combined the block unit damage assessment with building footprint information that sets the boundaries for the calculation of average parameter value [95], [96]. This allows a compromise of SAR resolution for rich polarimetric features. Brett [97] proposed an algorithm to extract bright curvilinear features that represent double bounce mechanisms in urban areas. The extracted features were used as a mask to identify locations where features of interest have changed from the bitemporal pair of SAR images. Bai et al. [98] trained a SqueezeNet neural network on a single post-event TerraSAR-X image to detect built-up areas and a residual network to classify washed-away buildings within the built-up areas.

Beyond SAR data, multimodal features have shown promising results. Rao et al. [99] combined high-resolution building inventory data, ground shaking intensity maps, and pre-and post-event InSAR-derived surface changes to perform multi-level and binary damage classification for four recent earthquakes. They compared their predicted damage labels with ground truth data from on-site surveys and achieved successful identification of over 50% of damaged buildings using binary classification for three out of the four earthquakes studied.

Data availability needs to be considered following a disaster. Adriano et al. [79] collected a multimodal dataset of bitemporal SAR and optical images for three large-scale disasters with various geographic coverage. Their work evaluates CNN for building-unit damage assessment by considering five data availability scenarios: whether SAR or optical modality is available in only post-event or together with a pre-event. They conclude that the highest accuracy model was trained on the scenario of bitemporal optical data, meanwhile, scenarios with SAR data involved are still challenging.

4.4 Advancing open research on building damage assessment

4.4.1 Public Dataset

For training deep learning models, a relatively large dataset with high-quality labels is needed. Many public datasets for humanitarian assistance and disaster relief (HADR) applications were proposed, as shown in Table 4.1. However, only a single dataset utilizing SAR is available.

Table 4.1 Overview of public datasets for natural disaster analysis

Dataset name	Image type	Temporal	Task	# of labeled class
ABCD [100]	Optical (Satellite)	Unitemporal	Classification	2
fMoW [101]	Optical (Satellite)	Multitemporal	Classification, Change Detection	63
xBD [82]	Optical (Satellite)	Bitemporal	Change Detection, Segmentation	4
RescueNet [102]	Optical (UAV)	Unitemporal	Segmentation	10
MSCDU [103]	Optical, MS, SAR (Satellite)	Bitemporal	Change Detection, Segmentation	2
QuickQuake [104]	SAR (Satellite)	Unitemporal	Classification	2

4.4.2 Open data providers (SAR data)

As with most research on BDA using SAR, the data for analysis were sourced from commercial providers. This limitation is holding back the advancement of DL approaches, which generally produce high-quality data. Fortunately, there is an increase of open and free SAR data collected by the twin Sentinel-1A/B sensors of the European Union (EU) Copernicus constellation, which allows fast mapping of damage after a disastrous event using radar data. Commercial satellite companies also launch open data programs for various applications including humanitarian disaster response. ICEYE [8], Capella [6], and Umbra [7] are among the commercial small satellite X-band SAR providers that actively share SAR images freely. ALOS-2 data from the Japanese Aerospace Exploration Agency (JAXA) in 3 m resolution StripMap mode were recently released as free to use for the 2024 Noto Peninsula earthquake [105]. To accommodate analysis, Maxar also regularly publishes VHR optical images, mainly for aiding disaster response [5]. A rich collection of remote sensing data and assessment was released for the 2023 Turkey-Syria earthquake as shown in Figure 4.5.

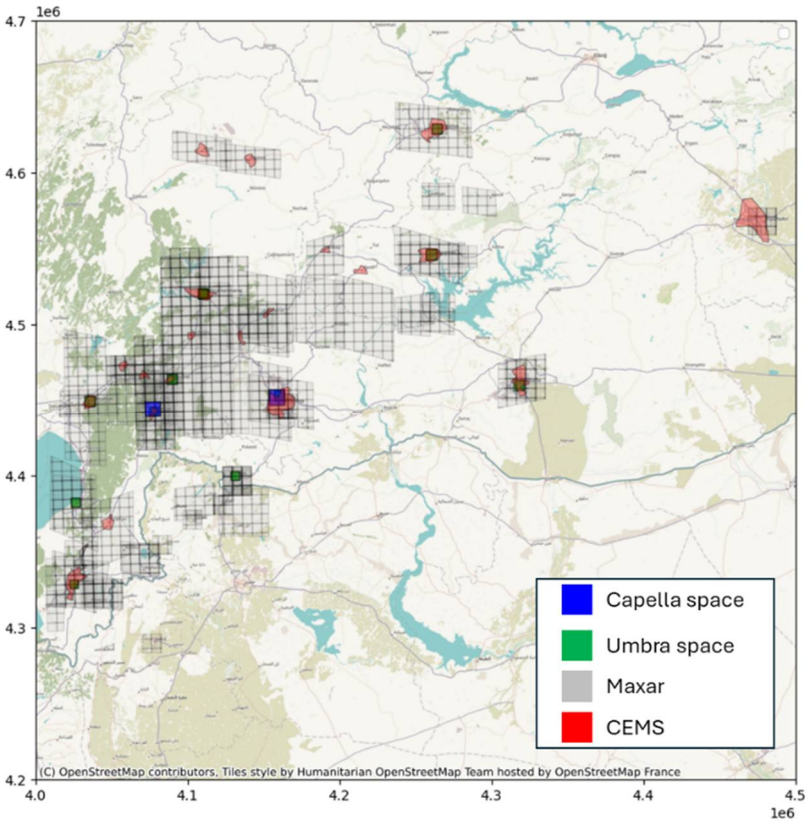


Figure 4.5 Extent of various open data for humanitarian purposes in the 2023 Turkey-Syria earthquake. Highlighted in blue and green are VHR X-band SAR data provided by Capella and Umbra space respectively. Highlighted in gray are large coverages of VHR optical, before and after the event, provided by Maxar. Highlighted in red are the building damage assessments for 20 cities in Turkey conducted by the CEMS.

4.4.3 Damage Assessment (labels)

Many international charters were built to archive disasters such as the CEMS [106], NASA's Jet Propulsion Lab (JPL) Advanced Rapid Imaging and Analysis (ARIA) [107], International Disasters Charter [108], and Sentinel Asia [109]. CEMS typically provides a manual or semi-automatic delineation map of various disasters in various forms including vector data. Meanwhile, other charters share only the analysis map, which limits their direct use as reference labels.

OpenStreetMap (OSM), an open geographic database updated and maintained by collaborative volunteers, also has a community named Humanitarian OpenStreetMap Team for crowd-sourced remote damage assessment using satellite imagery. Such initiatives are often used for rapid GIS-based mapping in humanitarian responses [110]. However, OSM contributors tend to overestimate destroyed or collapsed buildings and underestimate major or minor damaged ones [111]. This was concluded in an assessment study comparing remote

damage assessment with field surveys, where the differences are mainly due to the limitations of the satellite imagery in terms of resolution and field of view making it difficult to assess the extent of damage [111].

Global building footprint data are essential tools when working on building-unit analysis. The recently launched GlobalBuildingMap dataset [112] covers global areas for temporal coverage between the year 2018 and 2019. However, they have a spatial resolution of only 3 m which potentially misses out on smaller size buildings. Microsoft [113] and Google [114] also have large coverage of building footprint datasets with spatial resolutions of 0.5 m and between 0.3 - 0.6 m, respectively.

4.5 Discussion

Accurate identification of damaged buildings using SAR involves analyzing the change of SAR features before and after the event [55]. However, such VHR SAR data are typically tasked after a disaster has occurred preventing the use of change detection methods.

4.5.1 Challenges

Various studies mentioned in Section 4.2 attempt to quantify damage levels based on SAR properties used for analysis. However, the level of damage defined by natural hazards guidelines (e.g. from earthquake or tsunami analysis) is too complex to be identifiable from space. Thus, with more granular levels of damage, i.e. light or moderate grades, the accuracy of damage grades decreases [111], because cracks on the walls and detached joints are almost undetectable in satellite imagery. Nevertheless, from the point of view of emergency mapping and site reconnaissance, the main concern is the distribution of damage for prioritizing resources.

There is no unified benchmark for objective comparisons of remote sensing methods. In the computer vision community, the large-scale ImageNet [39] dataset is typically used to evaluate newly developed deep learning models. However, proprietary high-quality data are typically an advantage in the remote sensing community. For the task of building damage assessment, xBD is typically used as the benchmark [115]. Meanwhile, in SAR, several large-volume datasets have been published that address object detection [116], change detection, and multi-modality study [103], but still none yet for damage assessment.

4.5.2 Future directions

Many deep learning approaches use supervised learning, which requires large amounts of high-quality labels for training. This becomes a bottleneck in certain fields like medical imaging

and SAR remote sensing where labeled data is difficult to collect due to costly measurements or due to rare events. In most deep learning-based SAR studies, the reference labels are typically from crowd-sourced initiatives using optical data. This creates disparities between the ground truth and the radar features. Several proposed solutions are using simulated data or semi-supervised learning.

Simulated data are typically used in fields that require complex interaction within an environment such as self-driving cars and various tasks in robotics. In SAR, various simulation tools have been developed for urban areas, primarily used for mission planning, scientific analysis of complex backscattering, and for geo-referencing [25]. Simulators can be used for sensor design, algorithm development, and training.

Ray tracing methods can be used to simulate how objects appear in SAR images. This in theory can provide infinite synthetic training data or obtain measurements representing various environment conditions and acquisition modes, which is a difficult task to attempt in real-world scenarios. SAR simulators were used to improve object detection of ships [117] and military ground vehicles [118].

However, the practical use of synthetic SAR data is impeded by simplified reflection models and less reliable simulation of surface interactions. Difficulties of synthetic data generation scales with more details or more complex environments, such as an urban scene. Detailed building models such as buildings, balconies, and other façade details, are necessary to match the simulated 3D urban area with real-life radar signals [119]. These 3D models are commonly created through the photogrammetric analysis of aerial imagery or data obtained from airborne light detection and ranging (LiDAR) systems. A triangle mesh of the roof and vertical walls of buildings can be generated using a 2.5D contouring method [120]. For open-source research, the newly released Building3D dataset [121], which contains city-scale 3D building models in the form of point clouds, wireframes, and triangle mesh, can be useful for simulating an urban landscape. In future works, post-disaster building models can be used to simulate unique radar signatures from various damaged structures. Ho et al. showcased the potential of simulated SAR data for damage assessment. Using 3D building models of varying damage conditions, unique backscatter changes from layover and debris can be observed. Moreover, various look angles can be used, increasing the variety of simulated damaged patterns which can be harnessed by machine learning models [122].

Another solution to address the scarcity of labeled data is to use methods that rely on less labeled data, such as Self-Supervised Learning (SSL). Unsupervised deep learning models still

generally rely on pretraining of labeled data, which can pose issues of adapting between different modalities for remote sensing data [123]. In contrast, SSL methods can exploit unlabeled SAR data through pretext tasks such as predicting the angle of a rotated image or identifying if an augmented view of a sample is similar. A particularly popular method is the contrastive learning approach, which attempts to bring similar sample pairs closer in the latent space and separate dissimilar pairs apart [124]. A SAR feature extractor using SSL was proposed in [125] where the pretext tasks were designed specifically for SAR, such as log amplitude shift, sub-aperture decomposition, and despeckling. The trained model was evaluated on multiple datasets and downstream tasks, demonstrating the generalization capability of SSL methods.

4.6 Conclusion

SAR shows great potential for identifying damaged buildings in post-disaster crises. Multiple SAR features were summarized to classify intact buildings and damaged ones, with the best approach being multitemporal full-polarization data spanning over the period of the event. However, PolSAR data and suitable pre-event SAR data are not always available, therefore deep learning methods are among the best solutions when only post-event data are available.

A review of deep learning-based building damage assessment was presented. For this task, more studies used optical-based solutions compared to SAR. Optical features were also found to yield much higher damage assessment compared to SAR. This is mainly attributed to the lack of open and public high-quality data for advancing this research. Open SAR datasets are becoming more available, but labels are mostly still inferred from crowd-sourced data, which are known to have inconsistencies.

Finally, for future research, three solutions were proposed: first, a more integrated data collection for a comprehensive study on disaster assessment, which aims to have a high-quality benchmark dataset for this task, second, generating synthetic data through SAR simulators and accurate post-disaster building damage models, and third, adopting semi-supervised learning approaches to maximize the usage of unlabeled SAR data.

5 Data Augmentation for Building Footprint Segmentation in SAR Images: An Empirical Study

5.1 Introduction

Buildings are the main structures in any urban area. A building's footprint is a polygon surrounding a building's area when viewed from the top. Maintaining this geographic information is vital for city planning, mapping, disaster preparedness, or other large-scale studies. SAR provides consistent imagery compared to optical sensors, therefore, enabling consistent updates on the source of geospatial data. However, its unique properties are difficult for non-experts to analyze. This fact leads to the exploitation of automated methods such as DL using CNNs.

Automated building detection in VHR SAR images was demonstrated using CNN in [78], [126]. However, such a task is challenging due to complex backgrounds and multi-scale objects. High-rise buildings are particularly challenging due to a phenomenon called layover, which projects the building's wall at the ground towards the sensor, confusing pattern recognition algorithms. An extensive search space on various architectures, pre-trained weights, and loss functions for segmenting building footprints from optical and SAR images was performed in [127]. It was found that the diverse building areas and heights in different cities were problematic. Small-area buildings, mostly found in Shanghai, Beijing, and Rio, were undetectable, while high-rise buildings (mostly in San Diego and Hong Kong) degraded the model's performance due to extreme geometric distortions. Those models performed well in cities such as Barcelona and Berlin because most of the buildings were of moderate size and height.

Predicting well on unseen data or the ability to generalize is the main goal of training a deep learning model. It is a generally accepted fact that deep neural networks perform well on computer vision tasks by relying on large datasets to avoid overfitting [128]. Overfitting happens when a model fits its training set too well. This results in low accuracy predictions on novel data. For the task of building footprint extraction, a handful of datasets from optical sensors exist [129],[130], but unfortunately, not many datasets with VHR SAR data are available for public usage.

As discussed in the previous chapter, for data that are expensive to collect and label, such as radar or medical images, a common technique to boost performance is using data

augmentation (DA). DA increases the set of possible data points, artificially increasing the dataset's size and diversity. It potentially helps the model avoid focusing on features that are too specific to the data used for training, therefore, increasing generalization (the ability to predict well on data not seen during training) without the need to acquire more images [131].

In remote sensing, Illarionova et al. [132] performed object augmentation to increase the number of buildings in optical remote sensing images and demonstrated better building extraction performance. In SAR imagery, Yang et al. [133] showed improvements in paddy rice semantic segmentation by applying quarter-circle rotations and random flipping. Random erasing [134] on target ships was performed in [135] to simulate information loss in radar imagery and improve the robustness of object detection.

In this chapter, extensive experimentation on DA methods was explored using the SpaceNet6 [80] dataset for automated building footprint extraction. Performance comparisons were demonstrated, and algorithm effectiveness and trade-offs were discussed.

5.2 Methods

5.2.1 Dataset overview

SpaceNet6 was a competition to extract building footprints from multi-sensor data. The SAR data consists of quad polarization X-band sensor taken from an aerial vehicle, covering 120 km^2 of Rotterdam port, Europe. The large images were split into tiles of 450 m x 450 m, with 0.5 m spatial resolution in both range and azimuth direction. Images were captured using two flight orientations: north facing (orient1) and south facing sensor (orient0). Figure 5.1 shows the tile over a base map of Rotterdam city, marking the position of images from orient1 in green and orient0 in red. The direction of flight is indicated by the azimuth (az) arrow, while the sensor's direction is given by the range (rg) arrow. Each orientation creates different characteristics of how a building looks, namely the shadows and layovers.

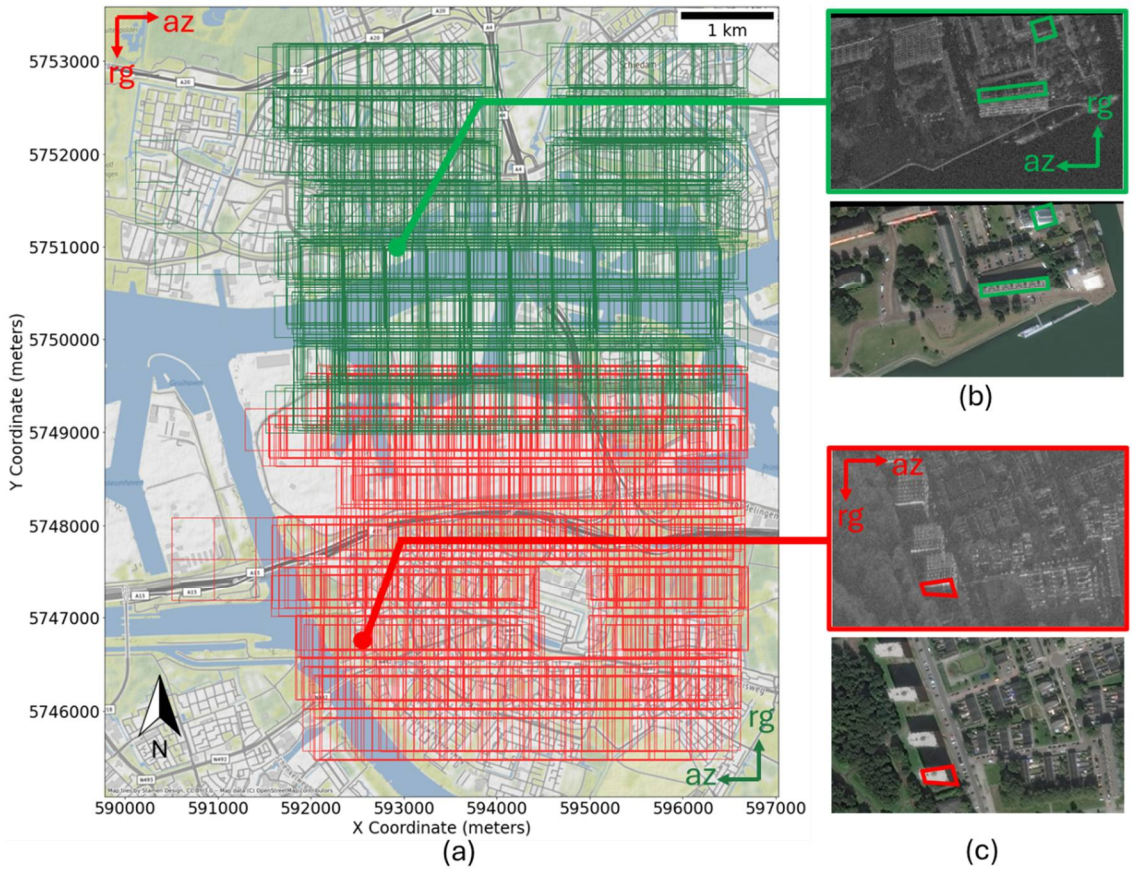


Figure 5.1 (a) The map of Rotterdam Port area (UTM Zone 31N) overlaid with tile boundaries for all 3401 tiles in the SpaceNet6 training set [80]. The optical RGB image is provided for comparison. Green tiles are orient1 while red tiles are orient0. To showcase the data augmentation methods in this study, only the green tiles (orient1) were used for training. Some building footprints are highlighted as an example of layover effects in (b) orient1 and (c) orient0 tiles. Notice the direction of a layover is always projected towards the sensor's position (near range) while the shadow is cast away from the sensor.

For training the Building Footprint Extractor algorithm, only the HH polarization was used out of the quad polarization data. Moreover, from the two flight orientations, only tiles from orient1 (covering the northern part of the city) were used. These constraints were applied to showcase the effectiveness of data augmentation methods which help solve overfitting due to limited data.

For evaluating the algorithm's performance, a separate dataset was processed from the expanded data version of the competition. This expanded data was released after the competition was finished, consisting of unprocessed Single Look Complex (SLC) SAR data covering a different part of Rotterdam port, i.e. the eastern part of the area shown in Figure 5.1a. This was used due to the high overlap between tiles in the training data, which can cause data leakage (part of the same samples shown in training and validation data). As with the

training data, only the orient1 data and HH polarization were used. These SLC data were processed similarly to the SpaceNet6 dataset as explained in their paper [80] which were then split into the same tile size as the training data.

5.2.2 Segmentation Model

For general building footprint extraction, there are only 2 classes: the positive examples, i.e., pixels belonging to a building's region, and negative examples, which are the rest of the pixels (non-building). Building footprints taken from overhead images typically have various sizes. To differentiate a building from the background, enough pixels should feature the whole or most parts of the building. This means a higher spatial resolution is required for detecting buildings with a smaller area while a larger coverage is needed to see large buildings. A common method in computer vision to help the model learn these multi-sized objects is to use multi-scale input, i.e., the input image downscaled to different pixel resolutions. Feature Pyramid Network (FPN) [136] utilizes image pyramids as explained in Section 3.3.2. The model used in this study combines the FPN architecture with the EfficientNet B4 backbone, which from previous studies [16], showed better results compared to UNet [40] architecture or ResNet [44] backbone. EfficientNet is a family of CNN models generated using compound scaling to determine an optimal network size [46] while B4 is one of their models with 17.5 million parameters.

5.2.3 Training and Evaluation

The training was performed in a Kaggle Kernel, a cloud computing environment equipped with a 2-core processor and an Nvidia P100 GPU (Graphics Processing Unit) with 16 GB of video memory (VRAM). The training pipeline was built using the TensorFlow framework and the Segmentation-Models library [137]. Adam [138] was used as the optimizer with default parameters and a cosine annealing learning rate scheduler [139] was used to modify α.

For model evaluation, the Intersection over Union (IoU) metric was used which is the ratio of overlapping between the predicted area and the real area (Figure 5.2). In this case, it is a pixel-based metric. A higher IoU indicates better predictive accuracy. True Positives (TP) are pixels labeled as building and are correctly predicted as building. True Negatives (TN) are pixels labeled as background and are correctly predicted. False Negatives (FN) are misclassified background pixels, while False Positives (FP) are misclassified pixels of buildings. IoU is calculated using

$$\text{IoU} = \frac{y_{gt} \cap y_{pred}}{y_{gt} \cup y_{pred}} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}. \tag{5.1}$$

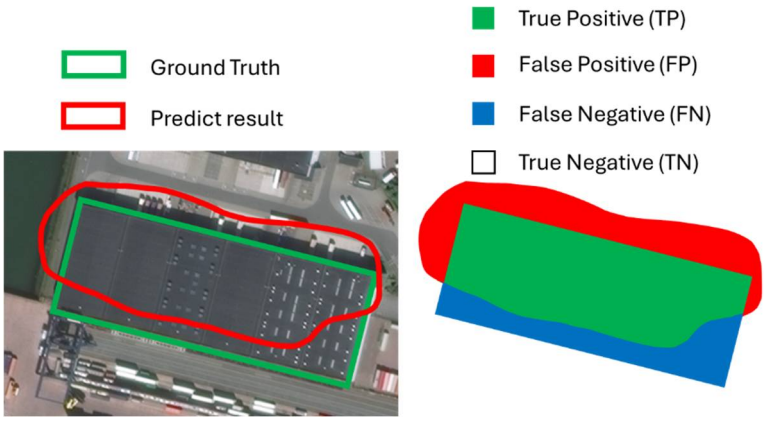
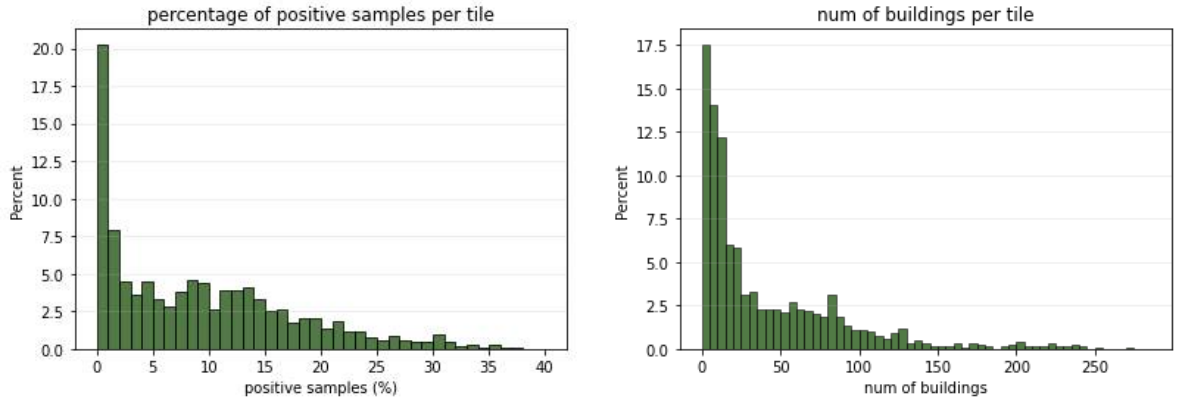


Figure 5.2 An example of how IoU is calculated over an optical image of a warehouse building. Image from SpaceNet6 dataset [80].

Calculating statistics over each image tile in the training set, 20% of tiles have less than 1% positive samples (pixels classified as buildings) (Figure 5.3a). This indicates that most tiles contain high negative samples (background pixels). One must be cautious in selecting a loss function for training a model on a skewed data distribution such as this because the negative samples will dominate the predictions. For example, using a binary cross-entropy as the loss function, the model will obtain a minor error even if it predicted the whole image as background pixels.



(a) (b)

Figure 5.3 Per image tile statistics for the training set, normalized. (a) shows the distribution of positive samples compared to the total number of pixels in an image tile and 20% of tiles have less than 1% of total pixels categorized as buildings. (b) shows the number of building counts for each image tile. Most tiles (17.5%) contain less than 5 buildings.

Several loss functions were experimented with, and it was concluded that Dice Loss [140] leads to better convergence for this dataset. It is based on the Dice Coefficient, which is used to

calculate the similarity between two samples based on the degree of overlapping, resulting in a loss or error score ranging from 0 to 1, where 0 indicates a perfect and complete overlap. Dice loss is simply 1-Dice Coefficient

$$Loss_{Dice} = 1 - 2\frac{y_{gt} \cap y_{pred}}{y_{gt} + y_{pred}}. \tag{5.2}$$

5.2.4 Ablation study

The impacts of various data augmentation methods will be assessed in an ablation study, which uses the same model and training configuration with different transformations during the data loading process. Subsets of the training dataset were used as training and validation data during the ablation study. To avoid confusion these sets will be named mini-training set and mini-validation set, which contains 37% and 23% samples from the main training set, respectively. After concluding which augmentation works well for the mini dataset in the ablation study, combinations of positively impactful transformations were applied to the main dataset.

5.3 Data Augmentation

This section describes the data augmentations used in this chapter and how they were implemented during the model's development. In general, the geometric transformations (including reduce transformation) were applied using TensorFlow operations, while pixel transformations were applied using the Albumentation library [141]. Transformation methods have names in **Capitalized Bold** format, while Class names from the Albumentation library used to implement those transformations are in *CamelCase* italic format.

5.3.1 Reduce Transformation

These operations were intended to maintain a square aspect ratio and reduce the image input to fit the GPU's memory. All resizing methods used bilinear interpolation, downsampling the tiles to 320 by 320 pixels, which allowed the batch size of eight for the single P100 GPU. Two main resize methods were tested:

- **Pad Resize**: no-data regions are added to create a square aspect ratio, taking the minimum pixel value of 0.0, and centering the image.
- **Distorted Resize**: the rectangle-shaped tile is resized to the square target resolution, distorting the shape of the image, but no black regions are present.

This downsampling process can be exploited to introduce randomness that further increases the diversity of the training samples. Cropping at random locations gave better details than just resizing the whole image as shown in Figure 5.4. However, because it introduces randomness, these methods cannot be used as a reduction method for the validation dataset:

- **Random Crop**: a random region is cropped out of the rectangle image. This preserves pixel scale since no downsampling is performed.
- **Random Crop and Resize**: crops a random location with a random scaling, then downsample it to the target resolution.

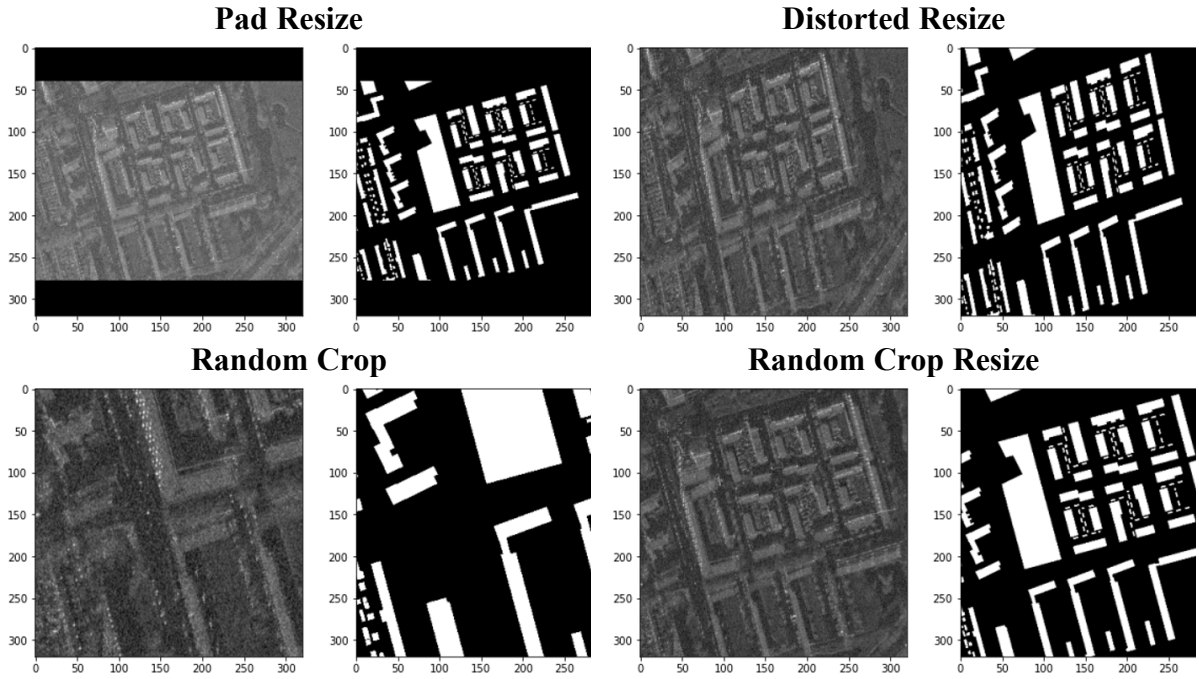


Figure 5.4 Summary of the used reduce transformations.

5.3.2 Geometric Transformation

In computer vision tasks, geometric transformations are cheap and easy to implement. However, it is important to be aware of choosing the transformations' magnitude that preserves the label in the image. For example, in optical character recognition, rotating a number by 180° can result in a different label interpretation in the case of the numbers six and nine.

Flipping an image along the horizontal or vertical centerline is a common data augmentation method. Referring to Figure 5.1a, the range direction rg for this dataset is on the vertical or y-axis, while the flight direction az is on the horizontal or x-axis. The **Horizontal Flip** does not alter the properties of a radar image. It would be as if the vehicle carrying the sensor was moving

in the opposite direction. In contrast, the **Vertical Flip** makes the shadows and layovers appear on the opposite side, creating inconsistency.

Rotation helps the model learn the invariant orientation of a building. **Rotation90** or quarter circle rotations $\{90°, 180°, 270°\}$ and **Fine Rotation** with a randomized angle range, e.g., $[-10°, 10°]$ were chosen. Similar to vertical flip, the quarter circle rotation affects the imaging properties of the radar. The fine rotation exposes an area where image data is unknown which was filled with the lowest value.

Shear is a distortion along a specific axis used to modify or correct perception angles. Despite SAR being a side-looking imaging device, the processed SAR image appears flat owing to the orthorectification process that corrects geometric distortions. In **ShearX**, the edges of the image that are parallel to the x-axis stay the same, while the other two edges are displaced depending on the shear angle range. **ShearY** is the exact opposite. Figure 5.5 illustrates shear in both directions. The shear rotation was set to be randomized between an angle range of $[-10°, 10°]$.

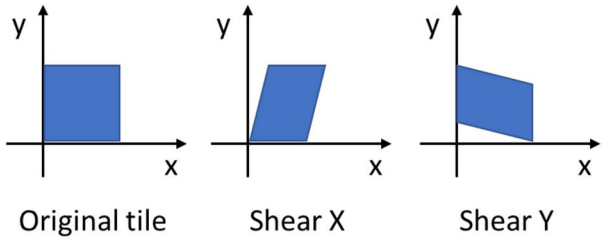


Figure 5.5 Shear transformations in 2 directions.

Random Erasing is an augmentation method inspired by the drop-out regularization technique [142]. It simply selects a random patch or region in the image and erases the pixels within that region. The goal is to increase robustness to occlusion by forcing the model to learn an alternative way of recognizing the covered objects. The erased patches were filled with the lowest pixel value. Random erasing was implemented using *CoarseDropout* class. The region's width and size were randomized from 30 to 40 pixels, and the number of regions created was randomized from two to ten patches. The proposed geometric transformations are shown in Figure 5.6.

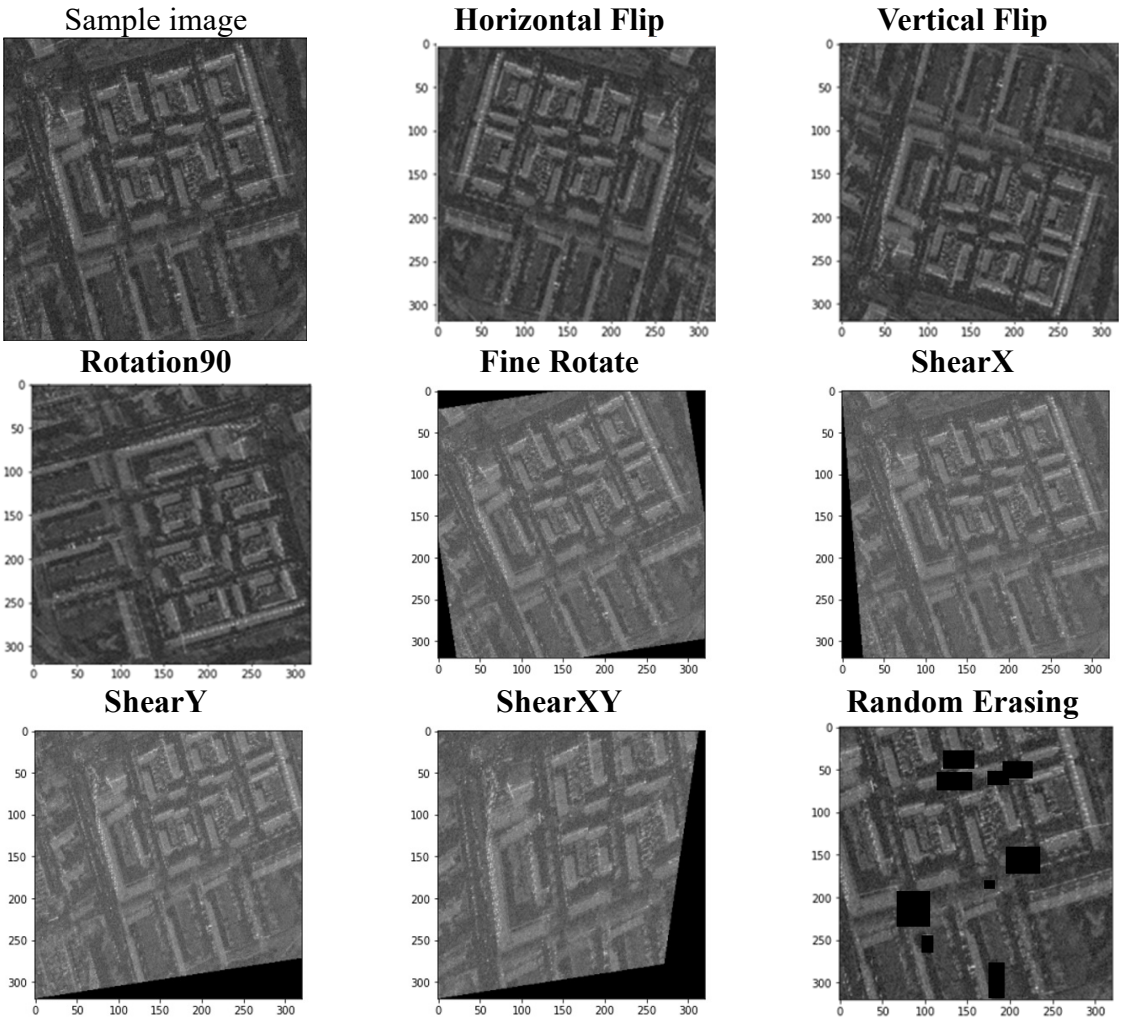


Figure 5.6 Summary of the used geometric transformations.

5.3.3 Pixel Transformation

In airborne sensors, unknown perturbations of the sensor's position relative to its expected trajectory can cause several defects such as radiometric distortions and image defocusing. To increase the model's robustness when encountering these defects, noise injection methods were applied. The common **Gaussian Noise** was generated using the *GaussNoise* class, while **Speckle Noise** was amplified by multiplying each pixel with random values using the *MultiplicativeNoise* class. Some images suffered from defocusing due to an unpredicted change in the flight trajectory, causing fluctuations in the microwave path length between the sensor and the scene [143]. This defocusing effect was simulated by applying **Motion Blur** with random kernel size using the *MotionBlur* class.

Sharpening with a high pass filter was used to improve edge detection. Consequently, this also increases other high-frequency components such as speckle. The histogram equalization was applied using Contrast Limited Adaptive Histogram Equalization (**CLAHE**) to maximize

contrast and improve edge visibility. The proposed pixel transformations are shown in Figure 5.7.

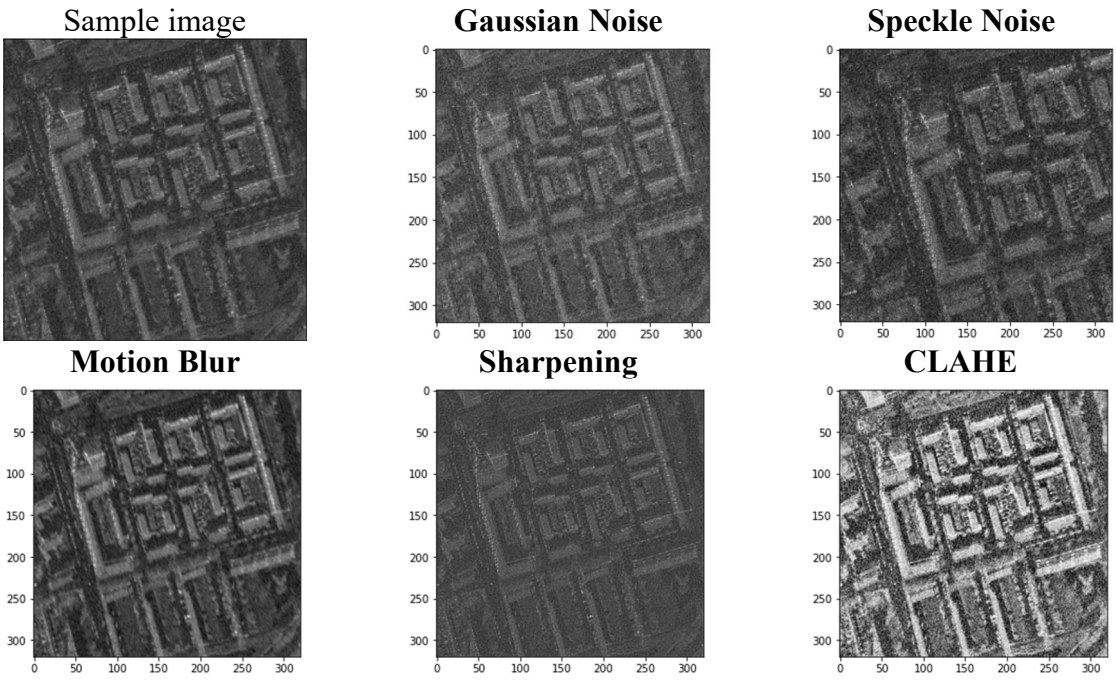


Figure 5.7 Summary of the used pixel transformations.

5.3.4 Speckle Filters

Speckle reduction filters such as a Box filter can smooth the speckle using a local averaging window. This is effective in homogenous areas, but in applications requiring high-frequency information such as edges, filters that can adapt to local texture can better preserve information in heterogeneous areas [144]. A previous study [49] has shown a slight performance gain by applying low pass filters with varying strength on the UNet model. In this research, the use of well-known adaptive speckle filters was applied as a form of data augmentation, namely Enhanced Lee (**eLee**) filter, **Frost** filter, and Gamma Maximum A Posteriori (**GMAP**) filter.

In Figure 5.8, two sample crops are shown for comparing filtration results. A good filter should retain the average mean of an image while reducing speckle [145]. In homogenous areas, the standard deviation should ideally be 0. Speckle filters were applied in MATLAB to the SAR intensity image (linear scale) and later converted back to the log intensity image (dB scale). The results of filtering are shown in Figure 5.8 (c) and (d). GMAP filter was measured to retain average value and reduce variance slightly better than eLee and Frost filter.

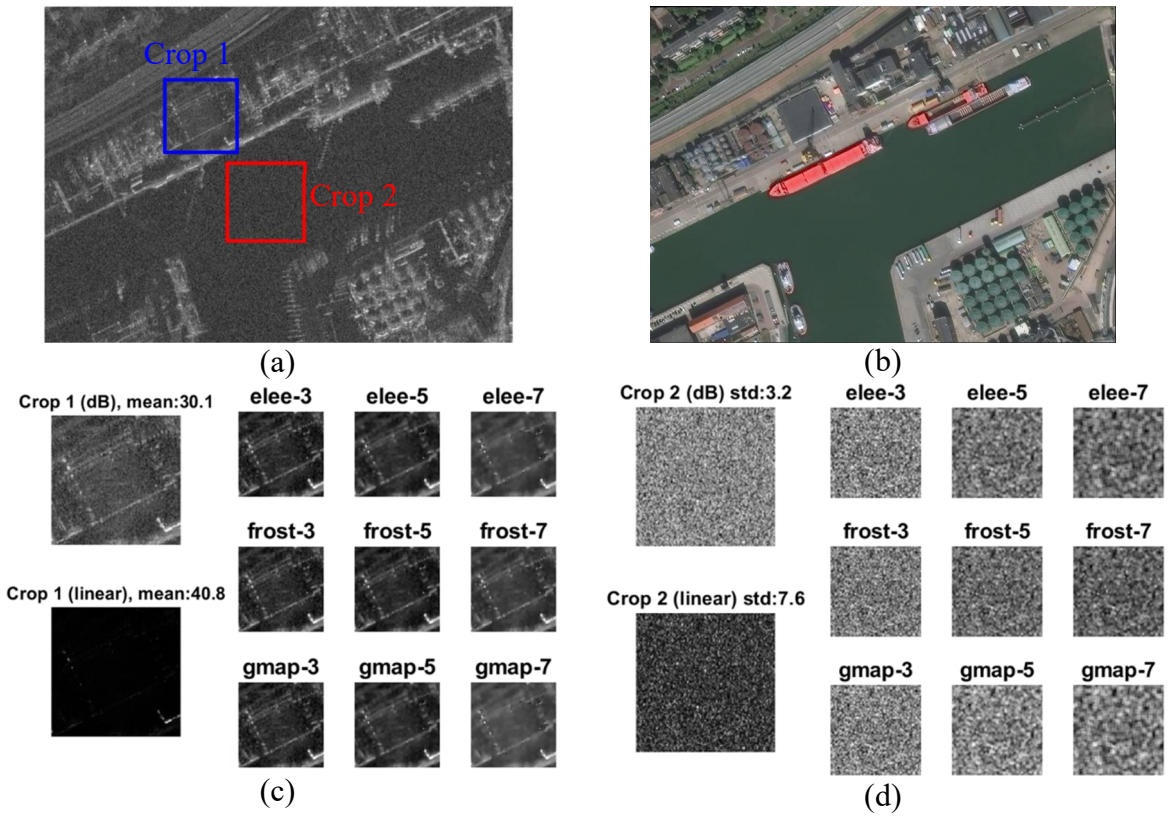


Figure 5.8 A sample area in SpaceNet6 dataset [80] of a SAR log intensity (dB) image (a) with its equivalent optical RGB (b). Two distinct crop regions were analyzed: a building area (crop 1) and a homogenous water area (crop 2). The results of filtering with selected speckle filters and various kernel sizes are given in (c) and (d). The image in linear scale appeared dark because of the wide dynamic range.

5.3.5 Data Augmentation Design and Strategy

Previous subsections address the DA methods, while this subsection discusses how those methods were applied. The frequency of DA is controlled by setting a probability of 50% chance for an augmented sample to load instead of the original sample. The magnitude of the transformation was also randomized in a value range, increasing the variation in every iteration of training, except for flipping and quarter circle rotation, which had a limited set of transformations.

The order of transformations is important when multiple augmentations are combined during the main experiment. Pixel transformations are applied first to prevent the presence of no-data pixels from affecting the results. Following it is a reduced transformation, and finally, a geometrical transformation. When using multiple pixel transformations, it is important to combine them into a “One Of” group. By chance, only one of the transformations will be applied, preventing the creation of a disastrous result. In geometric transformation, there is no

grouping, so an image has a chance to go through all transformations, which might increase no-data pixels but are generally less harmful than multiple filtering operations.

As categorized in [146], there are three stages of applying DA: Online (on-the-fly), Offline, and Test Time Augmentation (TTA).

In Online DA, the input data is manipulated during training. This can lead to a bottleneck if a fast accelerator is used in training, but the augmentation algorithm is slow, leaving the accelerator mostly waiting for data. The advantage is that it does not store the inflated data in storage. On the other hand, Offline DA allows complex manipulation and will not bloat training time. However, since it is applied before training, it takes up storage, and the variations are pre-determined (less randomness). Offline DA was used only for speckle filtered images since they were processed outside the TensorFlow environment, and an image was stored for every applied filter. Other transformations in this study used Online DA, which can have a finer degree of randomness in every iteration.

In TTA, N_T additional images are generated from each test image x, where N_T is the number of augmentations applied during the inference or prediction stage. The model will then predict on $N_T + 1$ samples and the average sum will be taken as the final prediction. This method of predicting multiple transformed versions of the input mimics the theory of ensemble learning, where a group of models using different architectures or trained on different data combines their predictions to increase generalization. This was investigated in [147], concluding that TTA helped reduce overconfident incorrect predictions compared to when using only a single model.

In a classification task, averaging predictions is straightforward since the output is an array with a size equal to the number of classes. In a segmentation task, one must be cautious to perform augmentations that modify the location of labels (in this dataset, the building footprints). If such methods are used, the solution is simply to revert back to the transformation before averaging the predictions.

5.4 Results

5.4.1 Ablation Study

To measure the impact of each augmentation method, an isolated experiment was conducted. The model was trained on the mini-training dataset and evaluated on the mini-validation dataset. Results are shown in Table 5.1. *Loss* and *IoU* are the scores for the training set, while *Val Loss* and *Val IoU* are scores from the validation dataset. The training lasted for 60 epochs. The four metrics were taken at the best epoch, which is when the model obtained

the highest *Val IoU*. This demonstrates the best score for each augmentation method compared to taking the score on the last epoch, which was always the worst due to overfitting. As observed in Figure 5.9, the gap between training and validation scores was less when an augmentation method was used, which delayed overfitting, enabling the model to move into what is known as a Local Minimum or a temporary performance peak.

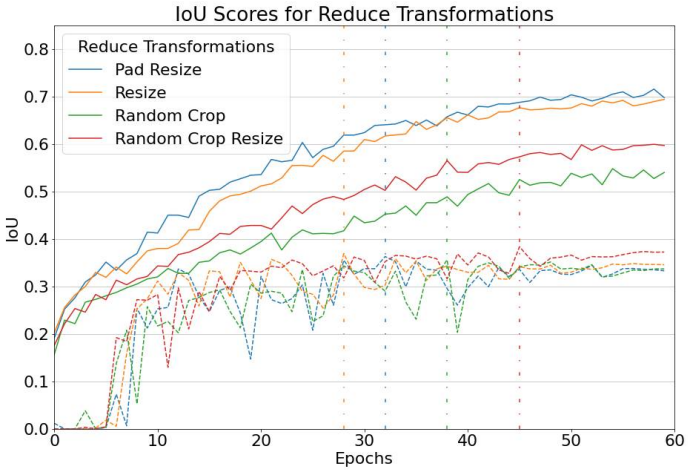


Figure 5.9 *IoU* scores comparing the four reduce transformations. The solid lines show training *IoU* scores, while the dashed lines show *Val IoU* scores. The loosely dotted vertical lines show where the best epoch (highest *Val IoU*) for a given method. Adding more variations to the input delays the overfitting and is shown by a later best epoch.

Using **Random Crop Resize** had the biggest performance gain compared to other augmentations. Randomizing the crop size gives the chance to see the image at different scales and details. **Random Crop** did not perform well due to the small static crop size of a 160 m × 160 m area, increasing the chance of encountering partial parts of a building.

Geometric transforms generally increased performance, except for **Vertical Flip** and **Rotation90**. Both were detrimental to the performance as they caused the extreme displacement of shadows and layovers' location compared to the actual building footprint. Pixel transforms were not as effective, giving similar or slightly worse scores than the baseline. These augmentations affect the recognition of texture, an important feature when the edges of a building or its shape are unrecognizable due to occlusion or noise. However, this filtering can also be destructive as it also amplifies non-building patterns.

Table 5.1 Results of ablation study. All scores are in percentage units. For loss, the lower is better. For IoU, the higher the better. Scores are color-coded in comparison to **Pad Resize**, where green color projects positive gain while red projects negative gain.

Method	*Loss*	*Val Loss*	*IoU*	*Val IoU*
Reduce transformations				
Pad Resize	22.04	47.40	64.10	36.33
Distorted Resize	26.35	46.57	58.56	36.95
Random Crop	34.75	47.97	48.92	35.63
Random Crop Resize	27.59	44.98	57.34	38.59
Geometric transformations				
Horizontal Flip	27.35	46.04	57.58	37.84
Vertical Flip	35.01	53.23	48.52	31.00
Rotation90	45.43	58.10	37.90	27.15
Fine Rotation [−10,10]	24.75	45.73	60.80	37.93
ShearX [−10,10]	22.07	47.42	64.11	36.32
ShearY [−10,10]	23.36	45.76	62.44	37.88
Random Erasing	22.48	47.84	63.56	36.01
Pixel transformations				
Motion Blur	25.79	48.10	59.45	35.88
Sharpening	20.82	48.20	65.81	36.02
CLAHE	31.20	48.36	52.69	35.53
Gaussian Noise	31.67	46.99	52.67	36.66
Speckle Noise	25.82	46.76	59.43	36.85
Speckle Filter—eLee	23.94	49.43	61.61	34.56
Speckle Filter—Frost	23.39	49.34	62.30	34.78
Speckle Filter—GMAP	20.36	47.39	66.40	36.38

Training scores were generally lower when augmentations were applied, as the model struggled to find the underlying function among the additional variations. A strong increase in training scores for the **GMAP** speckle filter indicates better recognition of the training data. However, these variations were not shown among the validation data, hence the lower validation scores.

To further validate the effects of proposed data augmentations, an ablation study was also performed on various mini datasets described as follows:

- SAR orient0, which had the north facing sensor, opposite orient1 (see Figure 5.1)
- PAN, also from the same SpaceNet6 dataset but uses the single band panchromatic images instead of SAR

- Inria [129], a VHR optical RGB aerial imagery at 30 cm spatial resolution. To enforce the limited data configuration, only 15 images each from Austin and Chicago were used as the mini training dataset. As for the mini testing set, 10 images from Vienna were used. Each image was divided into 25 image tiles of 1000 × 1000 pixels. The RGB images were converted into grayscale for a fair comparison with the other single-channel mini datasets.

Figure 5.10 shows that performance gain and loss on the other mini datasets mostly agree with results in SAR orient1. **Random Crop Resize**, **Horizontal Flip**, and **Fine Rotation** showed consistent gains over all datasets. Meanwhile, **Rotation90** showed consistent dips in performance, which are more prominent in datasets from SpaceNet6. The result from PAN highlights the method's impact on a similar geographic region (Rotterdam Port) but a different modality, while the result from Inria highlights the impact when exposed to the different urban settlements of multiple cities. However, due to the stochastic nature of deep neural networks, using an optimized model and training method fitted to one dataset might not translate to an optimal solution on another dataset, which has a different distribution [128]. Therefore, these directive insights should be further tweaked when working on a different dataset.

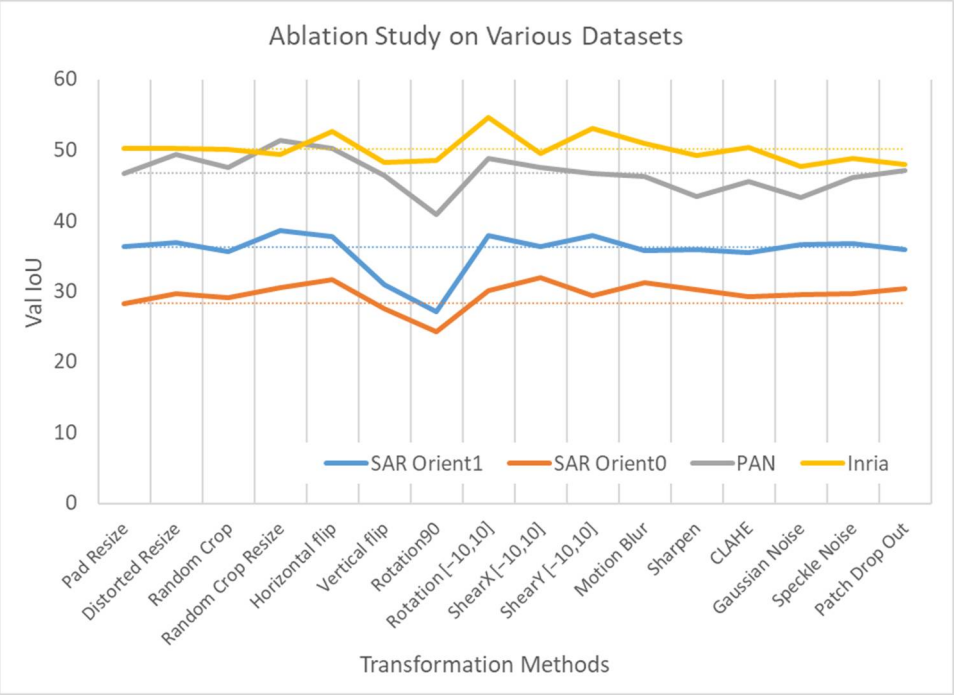


Figure 5.10 Results of the Ablation Study compared with three other datasets. SAR orient0 and PAN are from the same SpaceNet 6 dataset, while Inria is taken from the Inria Aerial Image Labeling dataset [129]. The *IoU* scores for the baseline method (**Pad Resize**) are marked with dotted horizontal lines for a straightforward comparison to other augmentation methods.

5.4.2 Main Experiment

Based on conclusions from the ablation study, several combinations of positive augmentation methods were applied to the main training set and evaluated on the prepared validation set. A similar segmentation model and equivalent training parameters were used except for a longer duration of 90 epochs. Again, the highest *Val IoU* was selected as the best epoch. The following augmentation schemes were applied:

- Baseline: No changes after applying a reduce transformation.
- Light Pixel: Motion Blur, Sharpen, and Additive Gaussian Noise.
- Light Geometry: Horizontal Flip, Fine Rotation [-10,10], ShearY [-10,10].
- Heavy Geometry: Horizontal Flip, Fine Rotation [-20,20], ShearX [-10,10], ShearY [−10,10], Random Erasing.
- Combination: Light Pixel + Light Geometry.

Only the Baseline experiment used **Pad Resize** as the reduce method, while the other combinations used **Random Crop Resize**. For every augmentation scheme, the model's performance was taken at the best epoch and shown in Table 5.2. Due to different datasets, the scores in Table 5.1 should not be directly compared to results from this main experiment.

Table 5.2 Results of combining multiple augmentations. All *Loss* and *IoU* scores are in percentage. For *Loss*, lower is better. For *IoU*, higher is better. Scores are color-coded where a darker green indicates a better value.

Augmentation Scheme	*Loss*	*Val Loss*	*IoU*	*Val IoU*	Best Epoch
Baseline	17.94	44.54	69.82	42.13	47
Light Pixel	23.47	44.57	62.28	42.72	81
Light Geometry	28.24	39.85	56.24	47.25	60
Heavy Geometry	28.90	41.02	55.46	46.12	67
Combination	29.04	41.27	55.36	46.05	74

In line with the results from the ablation study, geometric transformations had better scores than pixel transformations. Increasing the magnitude of transformation did not lead to an increase in performance, as shown by the lower scores obtained in Heavy Geometry. Increasing the diversity of transformations in Combination also did not improve performance despite consisting of transformations that showed positive impacts during the ablation study.

All models predicted well on medium-height elongated residential buildings (Figure 5.11a). Applying augmentation increases confidence, modeling a more accurate shape characterized by

rooftop patterns. However, fine details of the building structure and small buildings remained undetected.

For an image tile of large negative samples (pixels belonging to non-building), pixel augmentations drive extra attention to high backscattering objects such as container storages and large shipping/port equipment made of metal (Figure 5.11b, 4th row). This leads to an increase in false positives. Geometric augmentations were less prone to this. However, Geometric augmentations overfit non-building objects with building-shaped backscatters, such as the fences surrounding a sports field (Figure 5.11c, 5th row). A combination of geometric and pixel augmentations seems to tune down these false positives and correctly recognize them as non-object patterns.

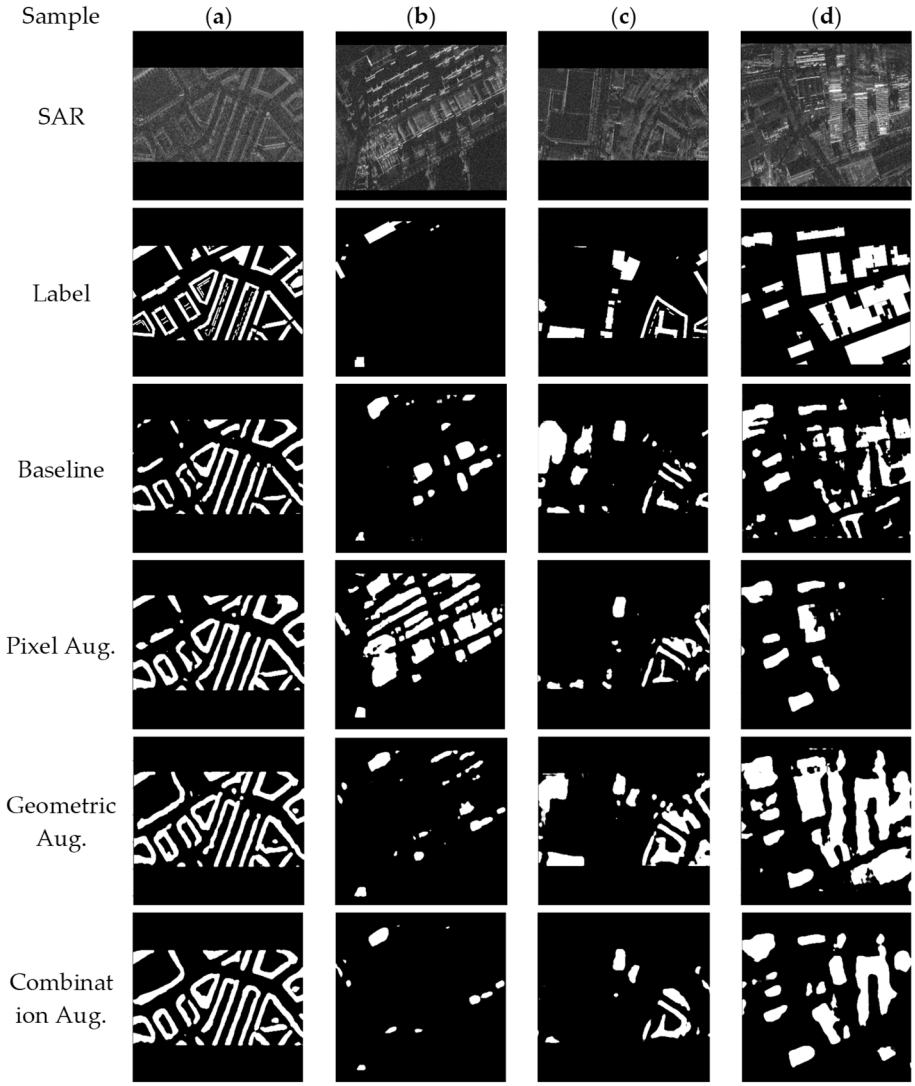


Figure 5.11 Comparison of predictions from the trained models of different scene objects: (a) medium-height residential buildings, (b) containers in a shipping port, (c) outdoor sports field, and (d) high-rise buildings.

Occlusion was the biggest problem, especially related to high-rise buildings in dense areas (Figure 5.11d). All models failed to recognize buildings occluded by the overlay of a neighboring high-rise building. Interestingly Geometric augmentations tend to classify the overlaid parts as positives.

5.4.3 Test-Time Augmentation

The state of the models in the main experiment was saved at their best epoch, and TTA was applied after the training ended. Two experiments were applied using TTA:

- TTA_1: Pad Resize was used for reducing image resolution. Transformations include Horizontal Flip, Rotate $\{-10°, 10°\}$ and ShearY $\{-10°, 10°\}$. After predicting each sample variation, an inverse transformation was applied, and the average sum was used as the final prediction. Total predictions per test sample: six.
- TTA_2: The rectangle image tile was divided into two square patches with some overlapping in the middle. This slightly increased the detail by utilizing the whole image space and removing the black bars. Afterward, TTA_1 was performed on each tile. Finally, the two prediction patches were combined by averaging the pixel values in the overlapping region. Total predictions per test sample: 12.

TTA was applied to the Baseline model and best model from the main experiment, which was trained on the Light Geometry scheme. TTA comes at the cost of additional inference time t_{test}, which is a scaling factor compared to the Baseline's inference time. It increases proportionally to the number of augmentations applied, e.g. for TTA_1 scheme with 6 methods, t_{test} grew to 3.3 times the baseline inference and grew to 6.3 times for TTA_2 with 12 methods.. Compared to the time required by re-training a model, the additional inference time to implement TTA was negligible. Results for TTA are shown in Table 5.3.

Table 5.3 Results of applying TTA to the Baseline model and Light Geometry. All loss and IoU scores are in percentage. For loss, lower is better. For IoU, higher is better. Scores are color-coded where a darker green indicates a better value.

TTA Scheme	Val Loss	Val IoU	t_{test}
Baseline	44.54	42.13	1.00
Baseline + TTA_1	41.79	44.65	3.35
Baseline + TTA_2	48.01	38.25	6.37
Light Geometry + TTA_1	39.05	47.90	3.39
Light Geometry + TTA_2	37.11	49.69	6.39

The Baseline and Light Geometry model benefits from TTA_1, which consists of simple geometric transformations. Interestingly, when TTA_2 was applied to the Baseline model, the performance was lower, as it predicted fewer positive samples from the two square patches. The Baseline model, trained on images with a fixed scale, had less confidence in predicting medium-sized buildings compared to the Light Geometry model, which had the chance to see variations of scaling thanks to the Random Crop Resize reduction method.

5.5 Discussion

SAR images have unique properties that differ from optical images. Therefore, several transformations, as shown in the ablation study, can result in poor performance. Selecting augmentation methods requires knowledge of the biases in the training data, either through statistical analysis, in the case of a large dataset, or through the manual inspection of samples. This helps reduce the search space instead of trying every available method.

Tiling is required in remote sensing images as it is impossible to fit a large raster directly to a model. The choice of target resolution will affect the detection of multi-scale objects, such as buildings. Introducing randomness by varying the scale and crop size during dataset loading is a cheap way of boosting performance since there is no need to store extra images, as in the case of tiling with overlapping regions. However, cropping too much will increase the chance of a large object covering the whole space and hinder performance. No-data regions are inevitable when tiling a large raster, and in the author's experience, it is better to remove them before feeding the image tile to the model.

This study shows that pixel transformations are not as effective as geometric transformations. The reason might be that kernel filters, which are the base of most pixel transformations, are already an integral component of the CNN model itself, thus, learnable by an adequately sized model.

TTA was demonstrated to be a cheap method to boost test scores. However, applying a set of augmentations during the test did not achieve better scores when compared to applying the same set of augmentations during training. The model predicted the varying test samples better, had it seen these variations during training. Therefore, applying augmentations in both stages will result in better scores. When using shear and fine rotations in TTA, the angle must be kept low because it removes some portion of the image (outside the image boundary) where it will not return when doing the inverse transformation after prediction. This is why quarter-circle

rotations and flips are more commonly used as TTA because they retain the full image after inverse transformation

5.6 Conclusion

This chapter presents several data augmentation methods for semantic segmentation of building footprints in SAR imagery. By artificially increasing the training dataset, the model's generalization on unseen samples was improved in the validation set, thereby reducing overfitting. The results show a 5% increase in *Val IoU* score when comparing the best augmentation scheme to the baseline model (no augmentation). Data augmentation can be very helpful in situations with limited data, either due to proprietary licenses or an expensive collection process.

For building detection in SAR, geometric transformations were more effective than pixel transformations. However, some transformations (such as vertical flip and quarter circle rotations) that alter key features of a building in SAR images were proven to be detrimental. Therefore, data augmentation must not be overused, especially since it takes more resources to train (either storage or processing time), which does not always lead to a better result. Additionally, TTA showed further performance gain compared to augmentations applied only during training. **Thus, it can be concluded that the first thesis of the dissertation has been confirmed**.

6 Detecting Large Scale Event from SAR Time Series

6.1 Introduction

Change Detection (CD) is a vital task in remote sensing analysis. One of its useful aspects is disaster assessment where SAR is more suitable than optical images, as it can observe an area even in bad weather conditions, which typically follows a disaster.

CD generally involves generating a difference image DI and then classifying change from it. For the first step, the log ratio operator is commonly used to highlight changes from pairs of bitemporal SAR images [148] which also lowers the impact of speckle noise [149]. Another approach to generating DI is based on similarities of statistical distribution between multitemporal SAR images, such as the Kullback-Leibler divergence [150] and the complex Wishart distribution for full polarimetric SAR [151]. For the classification step, the change classes can be determined using thresholding [152], or clustering [153].

With increasing amounts of remote sensing data, conventional algorithms started to be replaced by data-driven models such as neural networks. Complex pre-processing steps from conventional methods are difficult to scale and usually involve semi-automated analysis [154]. It is a well-known fact that neural nets perform better with larger data samples [155]. However, in the topics of remote sensing, it is challenging to generate labels to train detection algorithms, especially in SAR, due to unintuitive visual properties that are unique in radar images [15]. Recent research directions are slowly adopting unsupervised learning methods which minimize or even remove the need for labels for training [156].

Autoencoders are used to learn efficient encodings by reconstructing the input data without requiring labels [157]. Stack autoencoders were used as pre-text tasks in [158], and fine-tuned for detecting changes caused by wildfires. The reconstruction loss of an autoencoder can be used to determine the degree of change, by training an autoencoder only on no-change samples and assuming that changed samples cannot be reconstructed as good [159].

In this chapter, an autoencoder was trained to reconstruct multitemporal SAR data from ESA's Sentinel-1. The goal is to detect large event changes caused by various types of natural disasters. Therefore, higher temporal resolution (meaning a shorter gap between acquisition of the same area) is preferable to spatial resolution (which can induce unnecessary details). The autoencoder was used to learn representations of SAR data leading to a flood event. It was then

used to predict changes from other disaster types by taking the distance of encodings in the embedding space between bitemporal pairs of images as a measure of change.

6.2 Dataset

The inspiration for the data collection method was from the WorldFloods dataset [160]. It is a publicly available collection of satellite imagery of historical flood events from several existing databases in "machine-learning ready form". One of the databases where the flood extent map was derived is CEMS [161], which provides a catalog of emergency responses in relation to different types of disasters. The rapid mapping products have an event-specific vector package which was mostly derived either through manual photo-interpretation or semi-automatic extraction. This vector data is used as a reference for observed changes.

6.2.1 Flood Training Dataset

The multitemporal SAR data was collected over the Area of Interest (AOI) attached in the vector package. To programmatically search and collect SAR data, the Google Earth Engine [162] was used to access Sentinel-1 data. This approach allows the scaling of data collection process to potentially improve or better assess the algorithm. The Ground Range Detected (GRD) images were already pre-processed and georeferenced. Each pixel represents the backscatter (σ^0) in the logarithmic scale. The Interferometric Wide (IW) swath mode was used which has the default spatial resolution of 10 m. The polarization mode is in VH and VV. To reduce unnecessary changes from different viewing directions, the descending orbit was chosen. Additionally, for each location, a similar orbit pass number was chosen for all temporal images.

The temporal resolution or revisit time of Sentinel-1 is twelve days. With two Sentinel-1 satellites in orbit, a six-day revisit can be achieved, and some locations near the poles can even have up to a three-day revisit. Natural changes can be short (e.g. vegetation growth, sea waves), multiple days (e.g. floods, hurricanes), multiple weeks (e.g. wildfires, ice/glacier movement), or cause long-term landscape change (caused by e.g. landslides or earthquakes). The revisit time for Sentinel-1 (three to six days) is adequate to detect the disaster events in this study, which consist of floods, wildfires, and landslides.

For studies using optical images such as Sentinel-2 multispectral data [160], [163], permanent water areas were considered as floods since the overflow water from rivers or lakes is the common cause of fluvial floods [164]. From optical imagery, a change of color to brown is commonly observed in these water bodies due to the carrying of sediments from nearby land.

However, it is not possible to observe such changes in SAR as all water bodies still have low backscatters due to specular reflection. Therefore, only the observed event labels were considered as floods.

6.2.2 General Event Evaluation Dataset

The goal is to test if the change detection algorithm that was trained to detect flood events can generalize to other large-scale natural events in different locations. This set will be used to evaluate the model's performance, consisting of three types of large-scale events: floods, wildfires, and landslides. Each event has two locations. The dataset for testing was sampled from [7], consisting of multitemporal Sentinel-2 images. Similarly, with the flood dataset, the metadata was used to collect Sentinel-1 images over the same location and roughly similar timeframes. Table 6.1 shows the metadata of all locations used in this study.

Table 6.1 Metadata for the flood training dataset and the general-event evaluation dataset. For reference labels from CEMS, the identifier uses the code EMSR or EMSN, while others use reference labels obtained from the validation set in [7].

Location	Event identifier	Ref date	Sentinel-1 Post date (X_3)	Event	Area (km^2)
Training data					
France	EMSR 265	2018-01-25	2018-01-25	Riverine flood	887.9
Albania	EMSR 273	2018-03-22	2018-03-23	Riverine flood	202.8
Madagascar	EMSR 274	2018-03-18	2018-03-20	Flood by storm	48.2
Spain	EMSR 279	2018-04-15	2018-04-18	Riverine flood	2109.3
Italy	EMSR 330	2018-10-19	2018-10-23	Flash flood	244.8
Validation data					
USA	Carr Fire	2018-07-23	2018-08-17	Wildfire	1469.9
Australia	Riveaux Road Fire	2018-12-28	2019-02-27	Wildfire	1008.9
Greece	EMSR 271	2018-02-28	2018-03-01	Flood	586.6
France	EMSR 324	2018-10-16	2018-10-17	Flood	403.1
Iceland	Fagraskógarfjall landslide	2018-07-07	2018-07-09	Landslide	26.0
Chile	EMSN 053	2018-12-16	2017-12-24	Landslide	39.4

6.2.3 Notation

The reference labels have a binary class of 0 as no change, and 1 as change. For each event location, three pre-event images and a single post-event image were collected as close as possible to the reference date of the event. The temporal data is denoted as X_t, where X is an

image of a location, t is the temporal index where $t \in \{0,1,2\}$ is the pre-event images and $t = 3$ indicates the post-event image.

Figure 6.1 shows all timeframes for the location Albania, mapped to a false color of R=VH, G=VV, B=VV, for visualization purposes. The VV channel exhibits stronger backscatter for man-made structures (such as buildings and bridges), and for certain crop types that are sensitive to the double bounce scattering mechanism [165]. These features are highlighted in cyan in the false color composite. Meanwhile, most vegetation and agriculture areas are sensitive to the cross-polarization VH backscatter due to multiple scattering between branches and volume scattering. These features are highlighted in red. In both VV and VH, open waters have low backscatter due to specular reflection, therefore appearing as black.

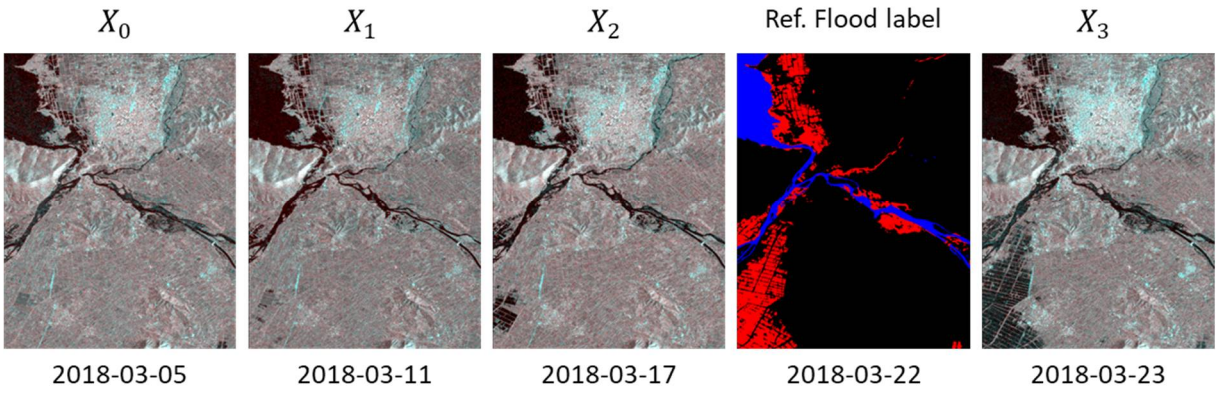


Figure 6.1 Multitemporal images from the flood in Albania. The reference flood label mask from the CEMS database: EMSR 273, highlights the flood that happened between X_2 and X_3. The red color in the flood mask indicates the observed flood extent, while the blue color indicates permanent water such as lakes and rivers. Only the flood extent was used as the reference. Below each image is the date of acquisition.

6.2.4 Preprocessing

It is not feasible to directly feed a large resolution image through a deep neural network. Handling varying aspect ratio input is also a challenge in the model's design, as the layer configurations need to be initiated. The common practice in remote sensing applications is to divide the image into smaller tiles (sometimes called chips or patches). This way, the input is a reasonable size image with a consistent shape. The tile size of 32 by 32 pixels was selected to crop input images. For every location in the training data, the full SAR image X_t, is divided into tiles $x_t^{r,c}$ where r and c denotes the row and column index of the tile.

The mean and standard deviation for the training data were calculated and used to normalize samples before feeding to the neural network. The data distribution of log intensity SAR images

is typically already a normal distribution [21]. Therefore, normalization transforms them into a standard normal distribution with decorrelated features, which was empirically proven to help training [166]. The preprocessing for both the flood dataset and general event dataset was similar.

6.3 Methodology

Remote sensing data is commonly used for detecting changes on Earth. Certain sensors are more sensitive to certain physical changes, which will be reflected by changes in radiance or backscatter value in the recorded image. Change detection (CD) attempts to compare images of the same location taken at different times. The simplest CD model would be to subtract two images represented by an N-dimensional vector X_τ^i and X_t^i taken at time τ and time t, to obtain a difference image $DI_{\tau,t}^i$, where $i \in 1, \dots, N$, is the spatial index. A threshold $\mathcal{T}$ is then used to determine how much of an intensity difference represents "change". Typically, $\mathcal{T}$ is expressed in terms of standard deviations away from the mean difference value, indicating "no change" when the difference is closer to the mean [167].

However, this simple model is not practically useful as the desired "change" is hard to define. Therefore, attempts should be made to transform pixel intensity values to make interesting changes more prominent and mitigate uninteresting changes. In SAR, natural changes such as vegetation growth, and the presence of speckle, can be challenging to develop change detection algorithms.

6.3.1 Autoencoders

Autoencoders are neural networks that are designed to indirectly copy its input to its output. Traditionally, it was used for dimensionality reduction or feature learning. Autoencoders consist of two parts: an encoder and a decoder. The encoder $\mathcal{E}$ creates a hidden representation (also called the latent space) z by an affine mapping of the input x given by $z = f(x)$. The decoder $\mathcal{D}$ maps the hidden representation z back to the original input space to generate the reconstruction of x given by $\hat{x} = g(z)$.

During training, the reconstruction loss $\mathcal{L}(x, \hat{x})$ will be minimized, which measures the difference between the input and the reconstructed input. Using regularization methods such as sparsity, training to find a denoising function, or penalizing derivatives [34], the autoencoder is restricted from copying the exact values from x, therefore, forcing it to learn meaningful properties of the input data. Autoencoders are trained with the bias-variance tradeoff in mind. On one hand, one would want the autoencoder to be able to reconstruct the input by reducing

the reconstruction error. On the other hand, one would want the learned low representation to be able to generalize [168]. In other words, the goal is not to obtain the best reconstruction $\hat{x}$, but rather how good are the representations learned in the form of z. See Figure 6.2 for the training pipeline.

A small autoencoder was used consisting of three convolution layers with kernel size 3 and stride size 2. The encoder compresses the input spatial size by half while increasing the depth of features. It outputs an embedding vector of size 128. The decoder expands the embeddings back to the input's original shape using upsampling layers.

6.3.2 Training

During training, the input tile $x_t^{r,c}$ was reconstructed by the autoencoder to output $\hat{x}_t^{r,c}$ and the weights of the network are updated based on the reconstruction loss. A low learning rate of 4×10^{-5} was used with a step decaying scheduler. Training was performed on an RTX A4000 GPU with a batch size of 256 for 20 epochs.

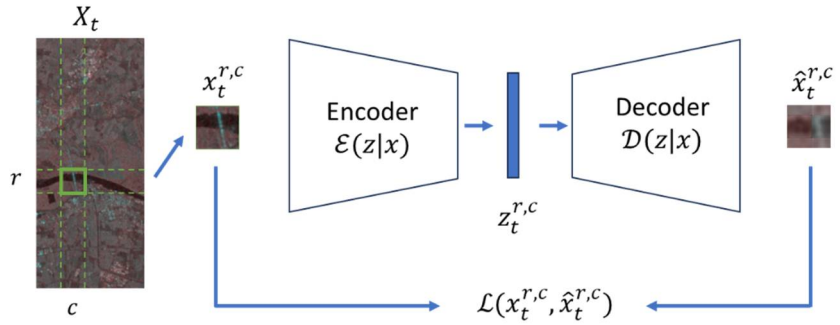


Figure 6.2 Schematic of the autoencoder during training.

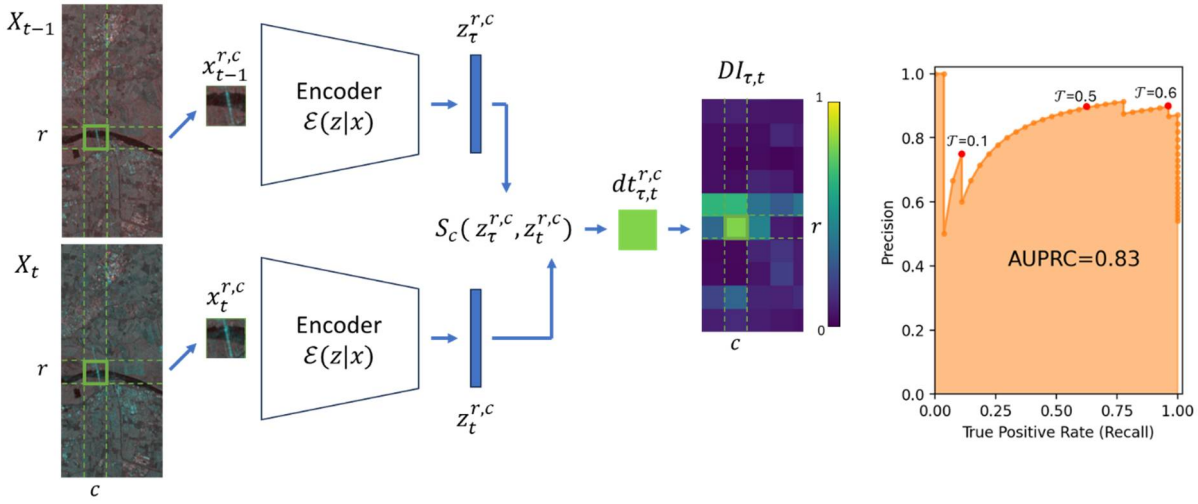
6.3.3 Evaluation

During the evaluation, only the encoder was used. Each input tile $x_t^{r,c}$ was encoded into $z_t^{r,c} \in R^d$ where d is the size of the embedding vector. Embeddings from adjacent timeframes were used to generate the difference tile $dt_{\tau,t} = S_C(z_\tau^{r,c}, z_t^{r,c})$ where $\tau = t - 1$ and S_C is the cosine similarity. The assumption is that as dt increases, there should be a greater magnitude of change occurring between the timeframes [18], [169]. The difference tiles $dt_{\tau,t}^{r,c}$ will be rejoined back based on their row and column position to create a difference image $DI_{\tau,t}$. See Figure 6.3 for the schematics.

A threshold value is further needed to binarize dt and assign the label Change or No Change. For pre-event pairs, the False Positive Rate (FPR) was used to measure the probability of false detection, i.e. the negative labels that were predicted as positives. The reference change labels

were only available for the last pair (co-event), therefore, the reference labels for pre-event pairs were all assigned as negative labels since it is assumed that no meaningful changes appeared before the event.

For the co-event pair (between $t = 2$ and $t = 3$), recall was used to measure the probability of detection. Selecting a threshold can create trade-offs between false positives (lowering threshold classifies more items as positive) and false negatives (vice-versa). The Precision Recall Curve (PRC) shows this trade-off for different threshold values. The Area Under the Precision Recall Curve (AUPRC) is a useful metric to aggregate the performance across those different thresholds. The score ranges between 0 to 1.



(a) (b)

Figure 6.3 (a) Schematic of autoencoder during inference. (b) Illustration of AUC of PRC over different thresholds $\mathcal{T}$.

6.4 Results

6.4.1 Reconstructing Flood Events

The autoencoder was trained on five flood locations each with four temporal SAR images. Training aims to minimize the reconstruction loss which is the Mean Squared Error (MSE). Figure 6.4 shows the comparison between the input and its reconstruction from one of the flood locations. The reconstructed tiles show a less detailed and slightly blurrier version of the original, also with visible texture discontinuation along the edges of the 32 by 32 tiles.

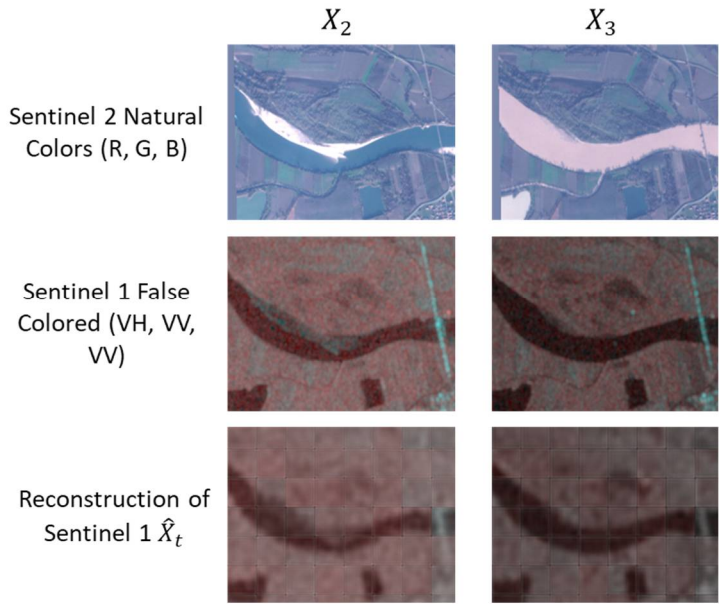


Figure 6.4 A crop from the flood location in Italy. The optical images from Sentinel-2 are shown for comparison.

6.4.2 Predicting on Other Events

The trained model was evaluated on six locations of three different types of events, utilizing the difference between encoded features from bitemporal tile pairs to measure the degree of change. The difference image $DI_{\tau,t}$ shows a probability map between 0.0 and 1.0 based on the normalized cosine similarity. To assign a label class, a threshold $\mathcal{T}$ is needed to split the probability map. As with any binary classification problem, assigning the value of $\mathcal{T}$ is a compromise between objectives. The effects of different $\mathcal{T}$ are shown in Table 6.2. Higher $\mathcal{T}$ reduces the number of predicted positive classes, therefore having fewer false positives as shown by the lower mean False Positive Rate (mFPR), but as a trade-off, it has poor change detection ability as shown by the low mean recall (mRec). Meanwhile, reducing $\mathcal{T}$ increases the predicted positive classes, leading to better detection ability, but as a consequence, the number of false positives also increases. The AUPRC score summarizes the performance over a range of threshold values, therefore, being a single metric to judge the model's skill.

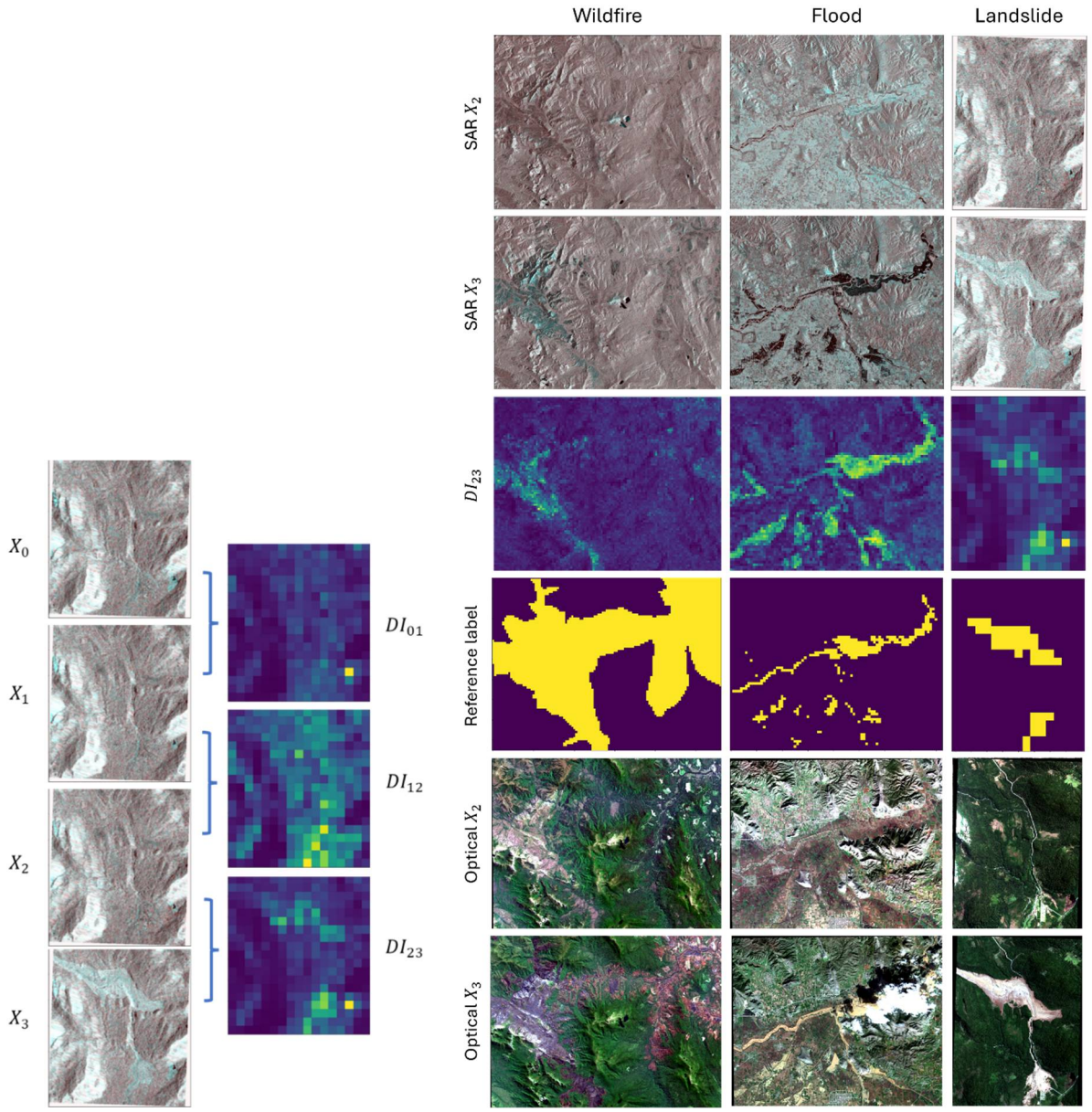
A full prediction from one of the landslide locations is shown in Figure 6.5a. Visible from the difference image between frames 1 and 2, DI_{12}, that surrounding vegetation impacts a lot of false detection from the pre-event pair, which ideally should be mostly dark similar to DI_{01}.

Table 6.2 Metrics averaged across all 6 locations of the evaluation set. The subscript numbers denote the SAR temporal pairs.

$\mathcal{T}$	Pre-event pairs		Co-event pair		
	$\mathbf{mFPR_{01}}$	$\mathbf{mFPR_{12}}$	$\mathbf{mFPR_{23}}$	$\mathbf{mRec_{23}}$	$\mathbf{mPrec_{23}}$
0.15	0.0023	0.0043	0.0281	0.0359	0.4729
0.1	0.0278	0.0437	0.0636	0.0600	0.3988
0.07	0.0757	0.1210	0.1286	0.0983	0.3494
0.05	0.1359	0.2018	0.2130	0.1713	0.3610
0.03	0.3051	0.4337	0.4781	0.4175	0.3802
0.01	0.8669	0.8879	0.9323	0.9149	0.4081

One location for each event and its predicted DI is shown in Figure 6.5b. Each event type shows a different kind of change in the SAR intensity image. Flood changes are depicted by dark pixels from the specular scattering due to surface water. Despite training from images of floods, the evaluation for floods was poorer than other events. This was due to a high number of false positives in surrounding agricultural areas, which most likely have a drop in backscatter due to increased moisture after heavy rains [170].

For wildfires, the burnt areas have lower backscatter in both VH and VV channels [171]. However, in this wildfire area, the drop was not consistent throughout the whole labeled burnt area. Some parts had more decreases than others. The burnt area from wildfires in radar images does not show as clearly as in optical images. Despite this, AUPRC for wildfires shown in Table 6.3 are very good (> 0.73) as the predicted changes are well within the reference burnt areas.



(a) (b)

Figure 6.5 Before and after SAR images and their change detection results from 3 different event types.

Finally, for landslides, the effect of land movement results in a prominent change of backscatter from the partial or total removal or modification of vegetation, which displays clear boundaries from unaffected areas. Landslides obtained one of the highest AUPRC scores.

Table 6.3 Change detection evaluation based on the AUPRC metric between last pair prediction with the ground truth label.

Location	Event	AUPRC	$\mathcal{T}_{optim}$
USA	Wildfire	0.7342	0.0160
Australia	Wildfire	0.7505	0.0321
Greece	Flood	0.6856	0.2435
France	Flood	0.4193	0.1622
Iceland	Landslide	0.7996	0.1043
Chile	Landslide	0.6757	0.0468
All locations		0.3968	0.3971

Interestingly, AUPRC scores for all locations shown in Table 6.3 were much lower (0.39) compared to scores for individual locations. AUPRC summarizes performance over multiple thresholds, however, the pixel distribution of DI for each event can vary. When calculated, the optimal threshold value for each event type was different, for example, lower than 0.05 for wildfires, and between 0.15 and 0.25 for floods. This poses a challenge for a single classifier to generalize on detecting change for different events. Other than the distinct appearance of changes, the pixel distribution of backscatter values could also affect this threshold difference.

6.5 Discussion

From extensive experiments carried out in this chapter, the challenges in using simple autoencoders for change detection are mainly two things. First is the disconnection between the goals of training and inference. The former aims to minimize the reconstruction loss, while the latter aims to improve the classification accuracy of detecting change. This leads to difficulties in hyperparameter tuning during training, since it's not clear when to stop the training. The second challenge is the noise produced by natural changes in the SAR scene which were unique for different disaster types. This affects negative detection rate, i.e. predicting from pre-event pairs of images as no change. Manually setting $\mathcal{T}_{optim}$ in post-processing was still necessary to obtain good performance. It was observed that the larger the scene, the more frequent noisy changes appear. Reducing the AOI to be as close as possible to the event of change will result in a better performance score, as observed in several scenes considering their total areas. However, this would defeat the purpose of developing large-scale event detection and an end-to-end manner is more desirable.

6.6 Conclusion

In this section, an unsupervised approach was proposed to detect general large event changes from multitemporal SAR images. A scalable workflow has been proposed for utilizing a public database of disaster events as search parameters for collecting SAR images. The autoencoder was trained to reconstruct pre-event SAR images and learn the underlying representations. The trained autoencoder was used to detect changes from bitemporal SAR pairs by computing the distance between their embeddings.

Detection results were observed to be sensitive to threshold values, which were used to binarize the difference image DI to Change or No Change classes. The distinct surrounding areas and unique characteristics of change from each event type resulted in different distributions in DI prompting the need for post-processing tuning. However, when using optimal threshold values, the model can detect changes from wildfire and landslide events with the best AUPRC of 0.79, despite only being trained on flood events. **Thus, it can be concluded that the second thesis of the dissertation has been confirmed.**

As discussed, the model is still sensitive to the noise from natural changes, which gets worse the larger the scene is. In future work, to improve the robustness of an end-to-end approach, a multi-scale prediction method can be used where a large SAR scene is split into large patches, determine areas with high probability of change, and finally split that patch into even smaller patches for finer localization.

7 SAR Imagery for Urban Density Analysis

7.1 Introduction

Assessing the compactness of a city's structure and analyzing the density of buildings is very important in the context of the city's morphology and urban compactness [172] [173]. The elements of a compact city can be understood as the physical presence of gray infrastructure objects and thus proportion of non-permeable surfaces. This includes built-up areas, traffic routes, industrial areas, or other areas covered with artificial materials.

Frequent and cyclical analysis of infrastructure information is critical to capture dynamic changes in urban areas. Such work on global products is undertaken within ESA's Copernicus program on a continental scale using remote sensing data obtained from Sentinel-2 [174], [175]. However, there are no methods of analyzing and developing data for cities on detailed scales, where it is also important to distinguish classes in terms of building density, as in the Urban Atlas (UA) database [176]. The available and widely used satellite optical images, although intuitive, have their limitations. Radar images, which are more difficult to interpret, require additional processing and appropriate software, but it records information different from optical images that can help distinguish between land cover classes in urban areas. Moreover, the active radar sensor is independent of sunlight and can penetrate through clouds, ensuring constant observation and more frequent updates.

Urban mapping can benefit from radar images since built-up structures induce strong backscatter and thus can be distinguished well on microwave imagery [12]. The different scattering mechanisms of anthropogenic objects - buildings, concrete structures, roads, squares, or other impermeable surfaces - makes these surfaces identifiable and distinguishable in terms of the scattering factor.

There have been many studies of LULC classification using PolSAR data. However, such imaging mode is not commonly available in most satellite operations due to the limited swath width and huge data volume [177]. Single-polarized SAR data are more commonly available and with better spatial resolution. However, there is not enough information in single-polarized data to extract physical scattering mechanisms. To compensate, speckle divergence and texture analysis can be derived from the radar intensity data. Speckle divergence was used in [178] and [179] to monitor built-up areas, while other studies used SAR texture analysis to improve the classification of land cover mapping [180], building footprint extraction [181], and change

detection [182]. In the case of the dual polarization C-band, the coherence matrix and the modified dual polarimetric decomposition proposed in [183] can be used for various analyses as demonstrated in [184], [185], [186].

Preliminary results from a previous study show the possibility of using textural features to distinguish building classes in terms of compactness and that sealed areas of different densities are better classified on radar than optical images [187]. In this research, the more commonly available single polarization and dual polarization SAR were used to distinguish land use and land cover classes from the UA database with an emphasis on urban density. A comparison between an unsupervised clustering algorithm and a supervised segmentation approach was explored on various features derived from the SAR data.

The main objectives of this study are the following:

- Provide a comparison of the single polarization X-band and dual polarization C-band SAR data for LULC classification in urban areas. Features derived from the radar intensity data such as texture and speckle divergence were used as input.
- Assess the limitations of these SAR features in relation to the UA dataset used as reference labels.

7.2 Dataset

7.2.1 Study area and SAR data

Cities with diverse topographical structures and various residential, commercial, and industrial buildings are selected as the study areas: Warsaw and London. To analyze and compare them effectively, fragments of these cities with various types of urban fabric were selected as the AOI for the study as shown in Figure 7.1.

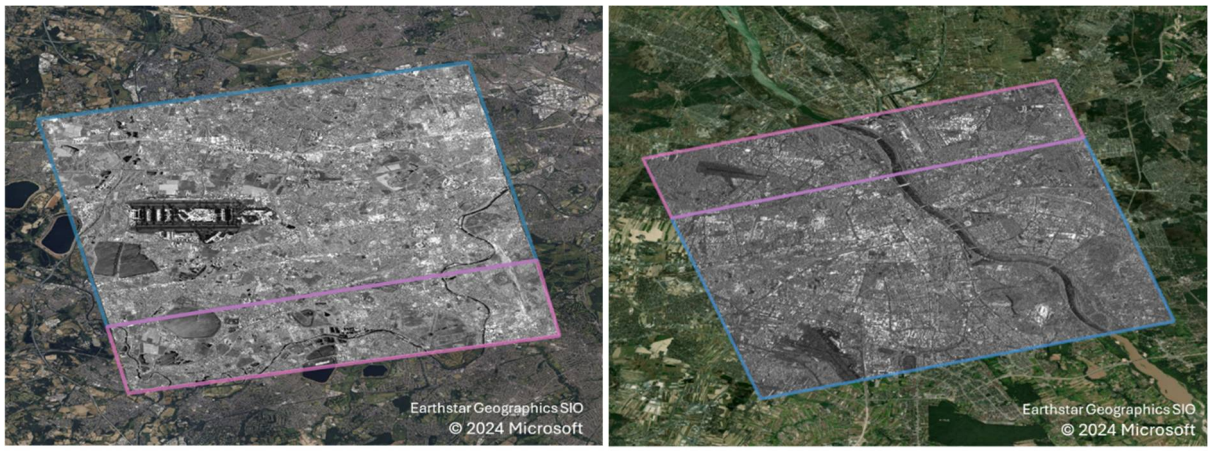


Figure 7.1 The study areas: London, UK (left) and Warsaw, PL (right). The blue polygon denotes the training area, while the pink polygon is for evaluation. The size of the training and evaluation area for London are 188 km^2 and 65 km^2 respectively, while for Warsaw, they are 202 km^2 and 65 km^2 respectively. SAR images by ICEYE [8] and basemap from Bing Maps Satellite Imagery [28].

For comparative studies, images from different sensors are needed to ensure a variety of data across a range of polarization (single and dual) and wavelengths (X and C-band). ICEYE (VV in 9.65 GHz) and Sentinel-1 (VV and VH in 5.4 GHz) are the datasets that ensure the feasibility of the experiment and test the influence of different factors on the results. The dates of the images were selected to be close to each other and cover the period without vegetation (Table 7.1). For images from ICEYE, The SpotLight Extended Area (SLEA) and Strip Map (SM) modes were used to capture the London and Warsaw areas respectively, while both scenes were acquired using the Interferometric Wide (IW) mode on Sentinel-1. Table 7.1 describes the data properties.

Table 7.1. SAR product specifications used in the study.

Parameters	**London**		**Warsaw**	
Imaging Mode	ICEYE SLEA	Sentinel-1 IW	ICEYE SM	Sentinel-1 IW
Band (frequency GHz)	X (9.6)	C (5.4)	X (9.6)	C (5.4)
Input format	GRD	GRD	GRD	GRD
Polarization	VV	VV, VH	VV	VV, VH
Orbit	Ascending	Descending	Descending	Descending
Look side	Right	Right	Right	Right
Ground resolution (m)	0.5 x 0.5	10.0 x 10.0	2.5 x 2.5	10.0 x 10.0
Date	2021-12-20	2021-12-18	2019-09-18	2019-09-19
Area (km^2)	253	253	267	267

7.2.2 Labels and urban density definition

To train and evaluate the models, the 2018 version of UA dataset of London and Warsaw was used [188]. It is the latest vector data consisting of LULC labels of various functional urban areas over European cities. The 27 LULC categories were aggregated into 6 categories with an emphasis on distinguishing dense urban areas. Table 7.2 shows the class distribution for each AOI. Some areas within the SAR coverage had no UA labels, these are categorized as NoData, which will be ignored in training and evaluation. An example of UA aggregated classes is shown in Figure 7.2. It is a snippet of the London study area featuring the Thames River and surrounding low-rise suburban houses. Large parks with small ponds are visible in the bottom left corner and on the top corners. Several industrial areas are shown in gray, which are mostly warehouses, shopping malls, and academies.

Figure 7.2 A snippet from the scene in London for comparing SAR features. Left: optical image from Bing Maps Satellite Imagery [28]. Right: corresponding UA labels [188].

Table 7.2. Class distribution of UA labels for both areas of interest

Class ID	Class Names	London (%)	Warsaw (%)
0	Background (NoData)	3.22	2.08
1	High density urban area	26.40	33.62
2	Medium density urban area	6.95	0.42
3	Road and railways network	7.49	10.72
4	Industrial area	17.24	19.08
5	Vegetation	33.31	31.64
6	Water	5.38	2.44

7.2.3 SAR features

To support the classification of land classes with reference to UA labels, several image features were extracted from each SAR data. Images from Sentinel-1 have dual polarization,

VV and VH. VV is also called co-polar, since it relates the same polarization for the incident and the scattered fields, whereas VH is called cross-polar since it relates to orthogonal polarization states. Sentinel-1 has image features for each polarization.

The following figures in this section visualize the SAR features in a grayscale colormap in the first row. The second row is their class likelihood distribution $p(\not{f}|c)$ which plots the probability density function for each class c in the given SAR feature $\not{f}$.

7.2.3.1 Log-Intensity

SAR intensity represents the reflected echo from scatterers on the ground. From Figure 7.3 it can be seen the additional detail in the urban scenes from the X-band SAR image compared to the C-band, where edges representing road lines and building blocks are visible in the former.

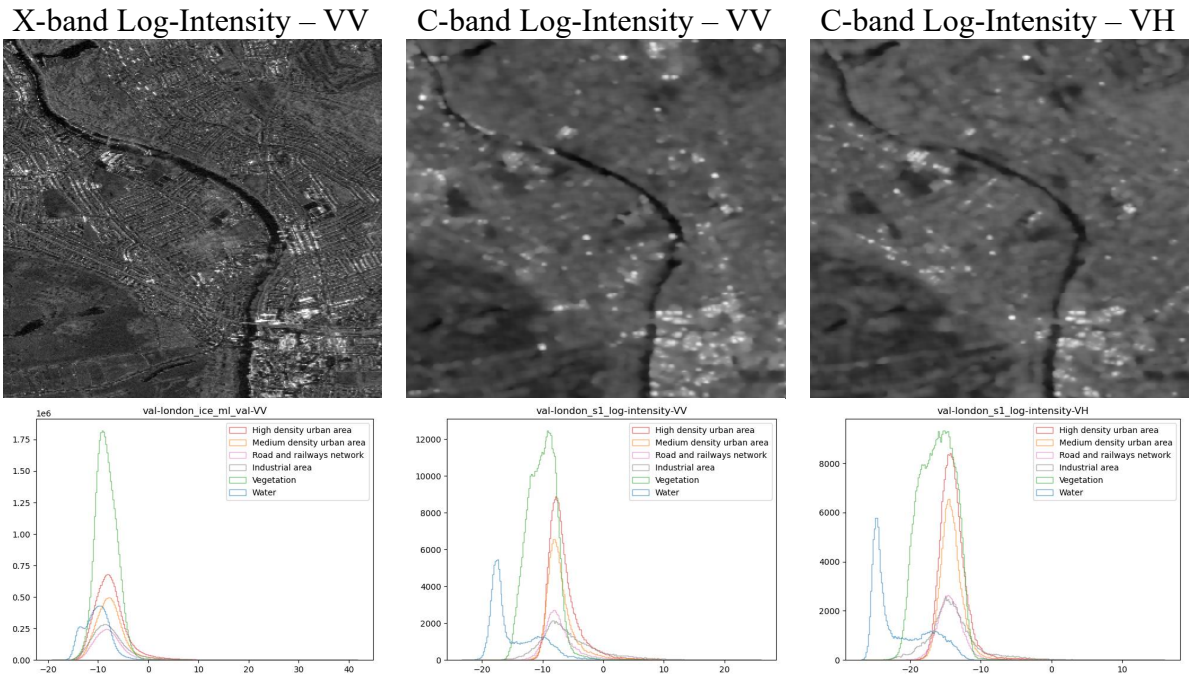


Figure 7.3 SAR intensity in logarithmic scale.

The class likelihood histogram for the log-intensity of X-band shows overlaps between classes. This indicates the difficulty in differentiating between the UA classes within this SAR feature. For the C-band log-intensity, the class Water is better distinguished than the other classes, occupying a low response in both VV and VH channels. The Water class has almost a bimodal distribution due to the difference in intensity between ponds and lakes compared to the river. Other classes for both polarizations still have significant overlaps. Several building areas appear brighter in VV, most likely due to their orientation relative to the sensor. Buildings that are oriented in a different direction than perpendicular can be mapped weaker due to the reflection from its walls. Backscattering also depends on the type of building – in the case of

residential buildings, the value of recorded reflected radiation is the lowest, it is higher for commercial areas and the highest for industrial [189].

7.2.3.2 GLCM

The detailed structure of ground objects can be effectively reflected in texture information. Buildings with a regular arrangement and shape show notable texture features in an image [181]. There are many different methods for texture analysis, e.g., Gray-Level Co-occurrence Matrix (GLCM), fractal analysis, discrete wavelet transforms, Laplace filters, Markov random fields, or granulometric analysis. In this research, texture features are extracted using GLCM based on the log intensity SAR image. Texture images derived by GLCM are the result of second-order calculations, meaning they consider the relationship between reference and adjacent pixels. Individual fragments of land cover were shown to have a higher correlation within their boundaries than between neighboring objects [190]. For a comprehensive review of statistical algorithms and mathematical formulations of GLCM, one can refer to Haralick [191] and Hall-Beyer [190].

Five texture features were derived: energy, correlation, homogeneity, contrast, and variance using a 9x9 window. GLCM - Homogeneity is shown in Figure 7.4 where similar-like pixels have high value such as water areas, while heterogeneous patterns such as buildings and infrastructure have low value. In X-band, the High density class (colored in red) occupies lower values, indicating the class is slightly distinguishable from the others, which have high overlaps. The Water class in co-pol VV is distinct, while in cross-pol VH, all classes tend to occupy a narrow response. This is consistent with other texture features in VH, indicating poor features for classification.

The GLCM - Variance is shown in Figure 7.5, which is a measure of heterogeneity based on the mean and scattering of pixel values within the GLCM window. However, class distributions have high overlap in both co-pol and cross-pol.

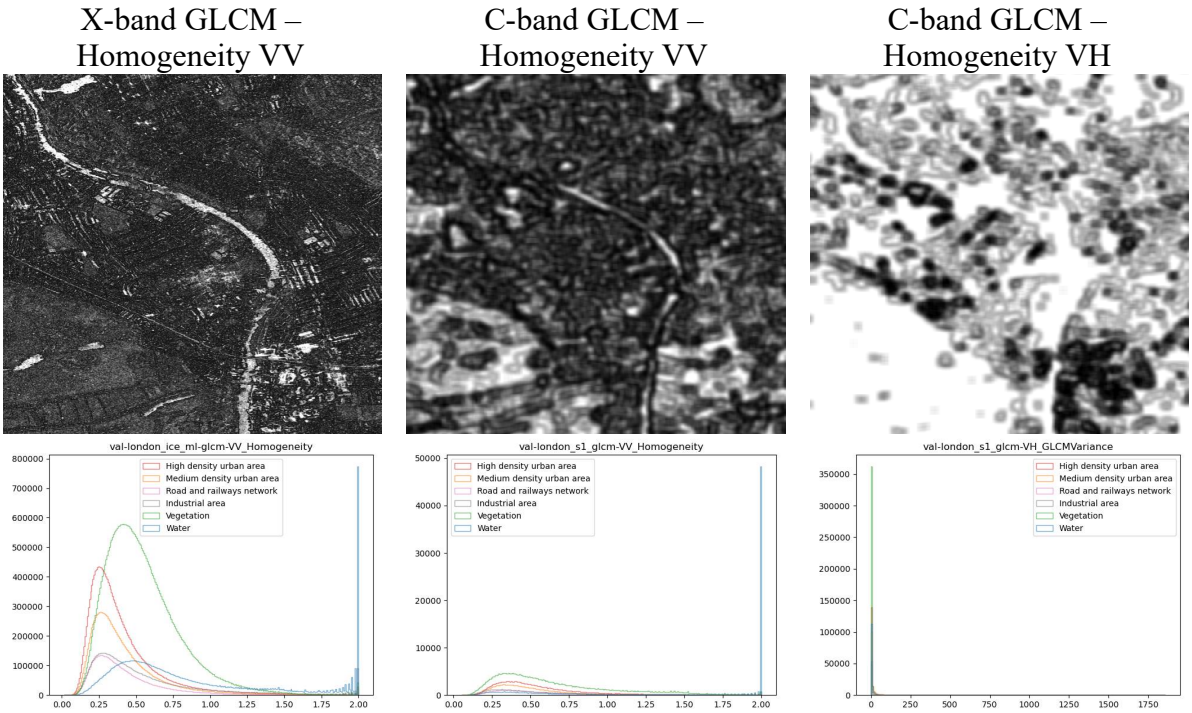


Figure 7.4 Example of the textural features: homogeneity calculated using GLCM for X-band (ICEYE) and C-band (Sentinel-1).

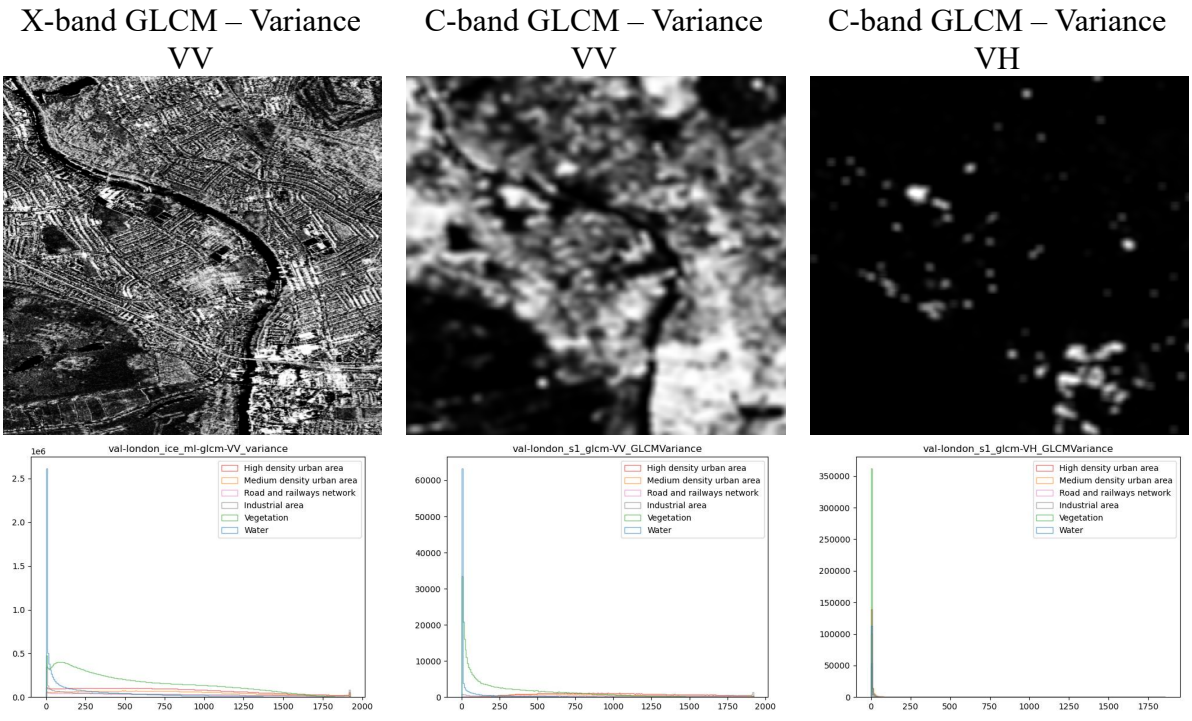


Figure 7.5 Example of the textural features: variance calculated using GLCM for X-band (ICEYE) and C-band (Sentinel-1).

7.2.3.3 Speckle Divergence

Speckle divergence from SAR log-intensity was used in [179] to delineate settlement areas that have the characteristics of bright intensity and high speckle divergence. This is in contrast with natural areas like agricultural fields, shrubland, or forest which often show relatively homogeneous texture. The class histogram in Figure 7.6 shows that all classes except Water still have overlaps. Artificial structures like buildings were shown in bright points. However, between High density and Medium density classes, there is no visible distinction.

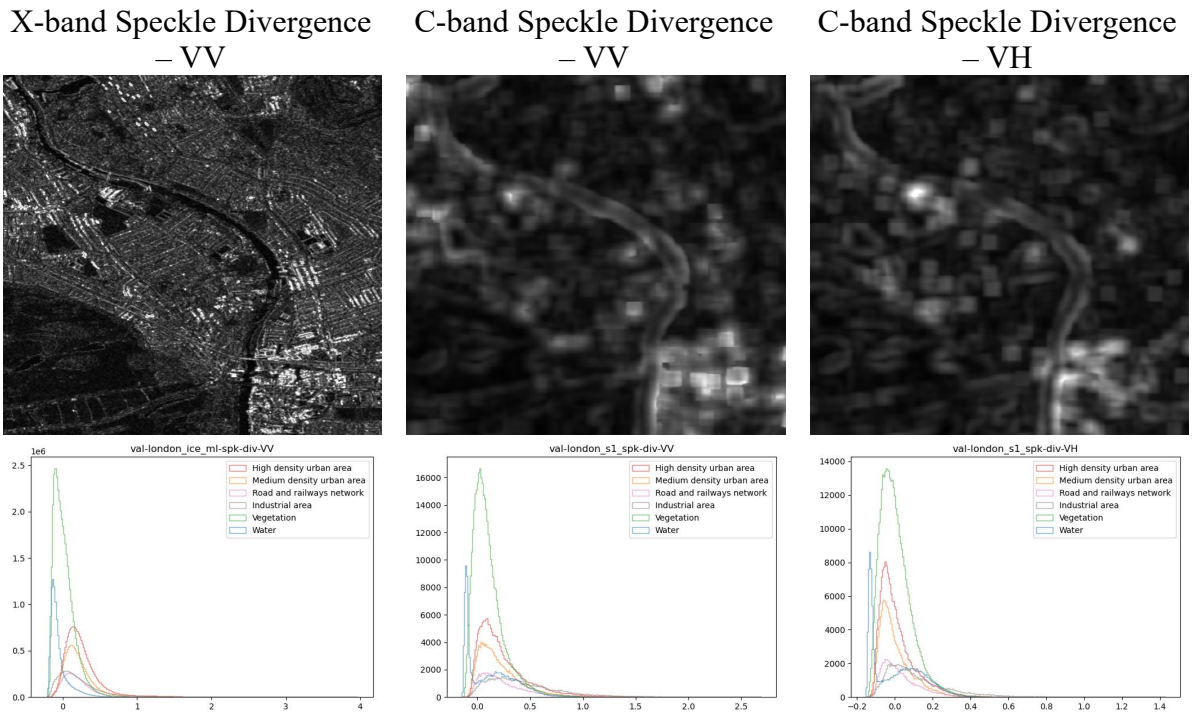


Figure 7.6 Speckle Divergence

7.3 Methodology

7.3.1 Workflow

The general methodological workflow is shown in Figure 7.7 and listed below:

- Datasets preparation
  - SAR data preprocessing with needed corrections.
  - Intensity SAR image calculation.
  - Training samples preparation according to urban classes' definition.
- Main data processing and analysis
  - Speckle divergence and texture performance of SAR data.
  - SAR image classification using supervised and unsupervised approaches.
  - Evaluation of the accuracy and comparison of the results.

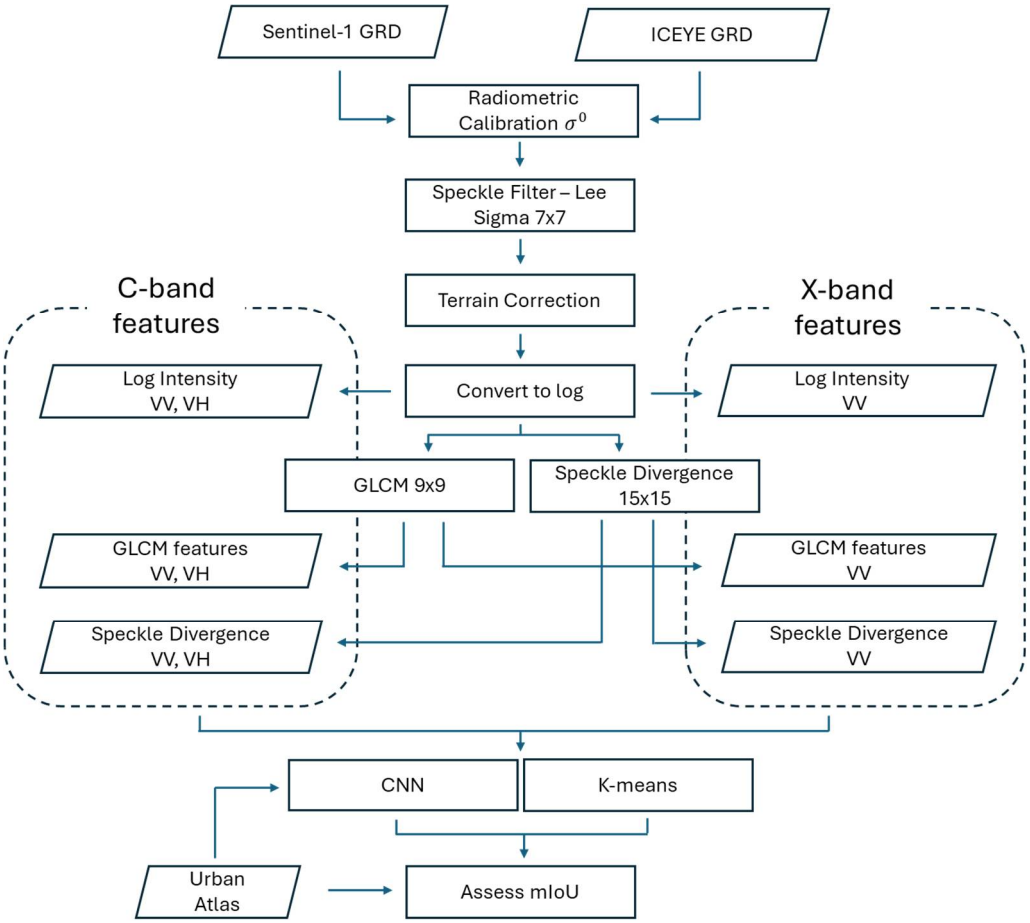


Figure 7.7. A schematic showing the preprocessing workflow in SNAP for both Sentinel-1 and ICEYE images.

Each SAR GRD image is preprocessed similarly using the Sentinel Application Platform (SNAP 9.0.0). It starts with radiometric calibration which converts the measured backscatter intensity to the normalized radar cross-section σ^0 by considering the global incidence angle of the image and other sensor-specific characteristics [192]. A speckle filter was applied using Lee Sigma with a 7x7 window size which, ideally, smooths homogenous areas while preserving edges between different surfaces. Finally, terrain correction was applied to geocode the radar image into the WGS 84 coordinate system and to correct for geometric distortions using the Shuttle Radar Topology Mission (SRTM) data as the Digital Elevation Model (DEM). Using SNAP, the speckle divergence was computed from the log intensity of VV and VH polarization, with a window size of 15x15. Meanwhile, GLCM was calculated using a window size of 9x9.

7.3.2 Algorithms

7.3.2.1 Unsupervised Classification

K-means clustering does not require the digitization of the training samples but only the number of clusters to group similar SAR features and it was used as unsupervised classification.

The K-means algorithm divided the SAR image into spatial clusters based on the mutual proximity of the data points. K-means' approach goal is grouping together identical data points and finding the underlying patterns. K-means searches a fixed number of clusters in a dataset to accomplish this aim [193]. The values of backscatter intensities and texture indices can be used to distinguish buildings from all other land cover types. Fourteen input clusters were established for the K-means algorithm which will be then aggregated to the target six classes based on the UA definition. The number of iterations after several trials was set at 20, which proved to be high enough to identify the clusters and sufficiently distinguish them in a meaningful processing time.

7.3.2.2 Supervised Semantic Segmentation

Segmentation involves partitioning the image into regions based on similarity and assigning a single class to every pixel. In this study, the UNet [194] architecture combined with ResNest26d [195] as the backbone was used for the segmentation model.

Tiling was performed on the large SAR raster with a tile size of 512 by 512 pixels with 128 pixels of overlaps for ICEYE images, and 256 pixels of overlaps for Sentinel-1 images. Overlaps increase training data since the AOI is not so large, especially for Sentinel-1 images with only 10 m/pixel spatial resolution. The UA polygons were rasterized with respect to each SAR dataset, generating label masks. NoData regions in the label masks were set to 0, whereas the first label class starts from 1 (High density) all the way to 6 (Water). During training, class 0 was ignored when computing the loss for optimization and when computing the metrics for evaluation. Both the training area and validation area in Figure 7.1 were preprocessed similarly.

The model was trained on an RTX A4000 GPU with 16GB of VRAM. The training pipeline was developed using the Pytorch framework and the Segmentation Models library [137]. Adam [138] was used as the optimizer with a learning rate of 10^{-3}. A step decaying scheduler modifies the learning rate with a decaying factor of 0.95. The batch size of 32 was used. From empirical findings, the reception field or the input size of the CNN was chosen to be 256 by 256 pixels.

7.3.3 Evaluation

The validation part of the study areas was used (pink area in Figure 7.1) to estimate the classification performance of the algorithms. In terms of binary classification, TP, FP, FN, and TN can be calculated. In multi-class classification tasks, the binary classification metrics for each class are computed, treating a target class as positive while the rest are merged as negative. The IoU was used as a pixel-based metric for classification accuracy assessment. In multiclass

segmentation, a single pixel can belong to one of the six classes (plus background as NoData). Therefore, the mean IoU (mIoU) from all classes is taken as the single metric to report.

7.4 Results

Two algorithms were compared: unsupervised clustering using K-means and supervised segmentation using Unet as the neural network architecture and ResNest as the backbone. To simplify the naming of algorithms, the former will be referred to as K-means and the latter as Unet. Table 7.3 shows the comparison of both algorithms on different SAR features. The IoU of each class is averaged to obtain the mIoU, which has a range from 0.0 to 1.0.

Unsupervised clustering has the advantage of not requiring any labels. However, K-means obtained poor results compared to Unet. As explained in the previous section, most of the UA classes have low separability in all the SAR features, making it difficult for a clustering algorithm that relies on the similarity of pixel values to distinguish them. Specifically for the classes High density, Medium density, and Industrial areas, they tend to show as bright lines from building edges. K-means mostly predicted these three classes as the same class. Predictions for each class are shown in the confusion matrix in Figure 7.8. The K-means results typically produce two large clusters, where one of them matches the wide distribution of class Vegetation. Results in C-band are better because of the dual polarization and smoother texture compared to X-band.

Table 7.3. The classification results as mean value of IoU

Location			**London**	**Warsaw**
Algorithm	**SAR data**	**Features**	**mIoU**	**mIoU**
K-Means	X-band VV	Speckle Divergence	0.1437	0.0922
		GLCM	0.1520	0.1284
	C-band VV,VH	Speckle Divergence	0.1600	0.1936
		GLCM	0.2379	0.2023
Unet+RestNest26	X-band VV	Log-Intensity	0.4471	0.4084
		Speckle Divergence	0.4172	0.3691
		GLCM	0.4387	0.4059
	C-band VV,VH	Log-Intensity	0.3573	0.3588
		Speckle Divergence	0.3093	0.3054
		GLCM	0.3407	0.3528

For Unet, the supervised learning algorithm and the powerful feature extractor of the CNN enables more pattern matching than just similar pixel values. Higher frequency changes within

dense areas were classified well by Unet, as shown in the confusion matrix in Figure 7.9. In most cases, class Vegetation had the best IoU score due to being the majority class at 33% of the total area, followed by class High density at 26%. Other classes are usually misclassified (false positives) in either of these 2 majority classes as shown by the brighter color in columns 1 and 5. Due to additional details, class Road networks are significantly better classified in X-band compared to C-band.

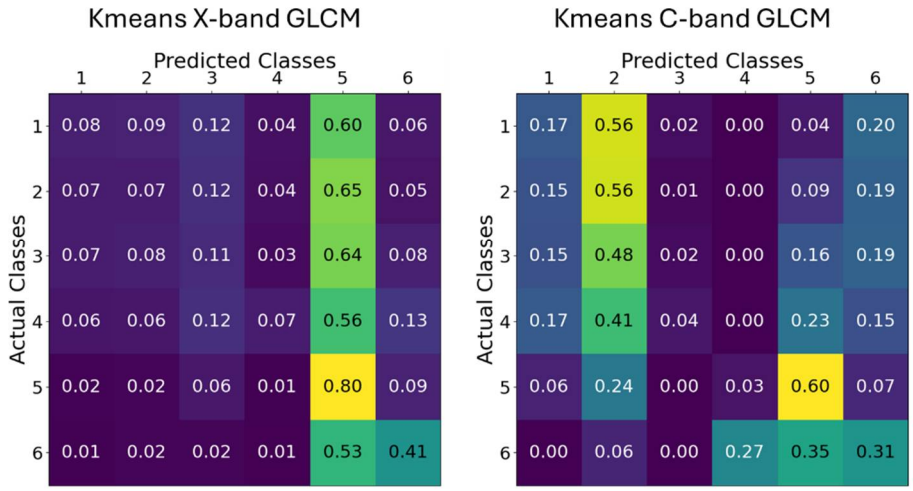


Figure 7.8. Confusion matrix for predictions from K-means with features (left) X-band GLCM (right) C-band GLCM

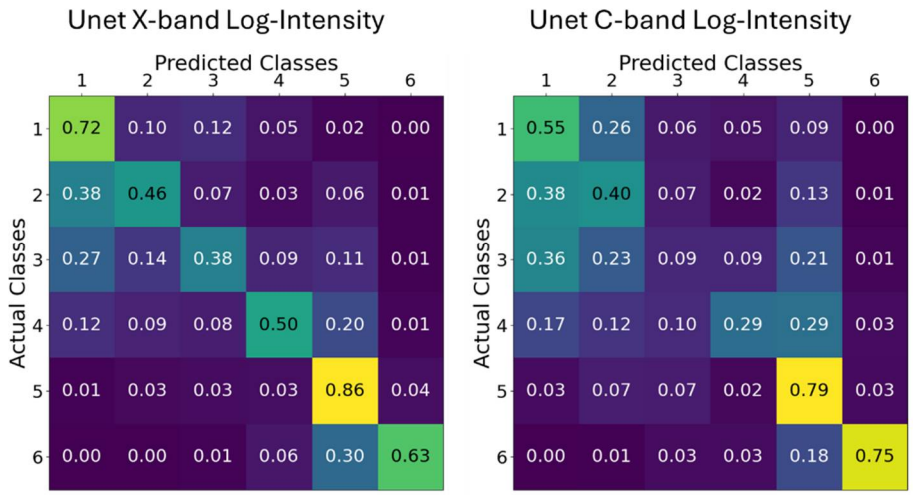


Figure 7.9. Confusion matrix for predictions from Unet with features (left) X-band Log Intensity and (right) C-band Log Intensity

A comparison of prediction results using the same area in Figure 7.2 is shown in Figure 7.10. It is noticeable that large homogenous objects such as the river were well classified using K-means. For Unet, the raster needs to be cut into smaller patches to maintain a reasonable number of weights or parameters. This means large objects were divided into different tiles,

making it difficult to do continuous prediction despite the object being relatively homogenous. This can be solved by increasing the tile size input to cover large objects. As C-band covers much more area within the tile, large objects were classified better, as shown in the first column in Figure 7.10.

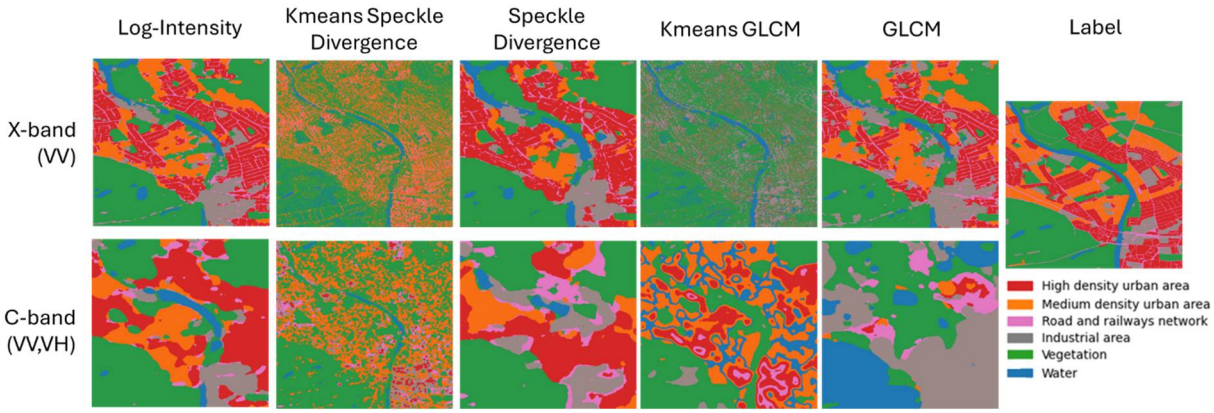


Figure 7.10. Prediction results from algorithms in Table 7.3.

7.4.1 Data Augmentation and Combining Features

Applying DA increases variations from a limited training set and can improve training of neural networks. Geometrical transformations were applied randomly to the input SAR images. Empirical results show that the best geometric DA methods were Random Rotation and Random Resized Crop. The former applies a rotation from the center of the input tile at a random angle between -30° and 30°. The latter randomly crops an area between 30-90% of the original tile size and resizes it to 256 by 256 pixels, which is the selected height and width of the input image for the Unet model.

As shown in Table 7.4, DA improves mIoU, by 0.03-0.05. This was consistent with other combinations of using Log Intensity with either speckle divergence or GLCM features. Similar to Table 7.3, the single feature of X-band Log Intensity yielded the best performance of 0.4742 mIoU. Combination from other features did not improve the score, most likely because they were derived from Log Intensity, therefore not providing new information. The robustness of CNN itself was able to efficiently extract features related to texture and edges. The comparison between combined features is shown in Figure 7.11.

Table 7.4. Unet evaluation results with Data Augmentation (DA) and combinations of different SAR features.

Location		**London**	**Warsaw**
Combination	X band Features	mIoU	mIoU
Comb 1	Log-Intensity + DA	0.4742	0.4538
Comb 2	Log-Intensity + Speckle divergence + DA	0.4663	0.4514
Comb 3	Log-Intensity + Speckle divergence + GLCM + DA	0.4636	0.4462

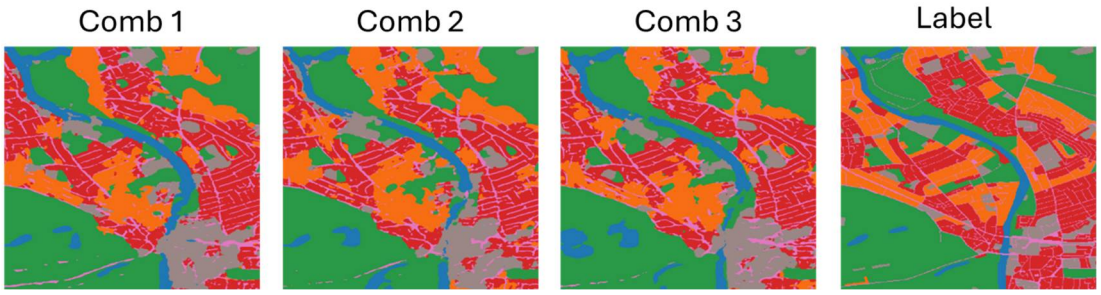


Figure 7.11. Prediction results from algorithms in Table 7.4.

7.5 Discussion

It is worth noting that most X-band SAR satellites work in VHR single polarization mode, while most C-band SAR satellites can work in polarimetric mode (either dual, quad, or compact polarization) [177]. Isolating the effects of the SAR frequency band is not possible due to different acquisition settings, therefore, analysis is focused on different details and polarimetric features from the C and X-band SAR data.

7.5.1 Weak descriptive features from single and dual polarization SAR

Despite more details in X-band SAR, only utilizing the intensity values and textural features derived from it are still weak at distinguishing land use classes. This is mainly due to similar backscatter values for completely different objects, e.g. the specular reflectance of water has a similar low backscatter as shadows created by the blind spot of high-rise buildings. Several studies have pointed out the limitations of single or dual polarization SAR for classification [183] [196]. The Unet algorithm performed better than the K-means clustering algorithm since it learned iteratively the relationship between labels and the underlying radar signature. Neural networks consider not only spectral and textural features, but also geometric and multiscale neighboring information, similar to what a human SAR analyst would do.

7.5.2 Trade-off between object size and details

The tile size of the C-band SAR image is 100 times larger than the X-band due to the different spatial resolution (10 m/pixel compared to 1 m/pixel respectively). Large objects such

as the Queen Mary Reservoir (see Figure 7.12) were delineated better in C-band. Moreover, the finer details and the shorter wavelength SAR mean that the X-band radar is more sensitive to small surface roughness. As a result, the water surface appears to have a non-homogenous texture due to backscatter from small ripples. The trade-off is with finer details, smaller objects such as residential buildings and roads are still observable, whereas in C-band, only the highway or rail stations are still visible. This is reflected in the confusion matrix for K-means in Figure 7.8. For Unet predictions, in C-band, class Water has higher True Positives of 0.75 compared to 0.63 of the same class in X-band. However, class Roads and railway networks were classified poorly compared to using X-band.

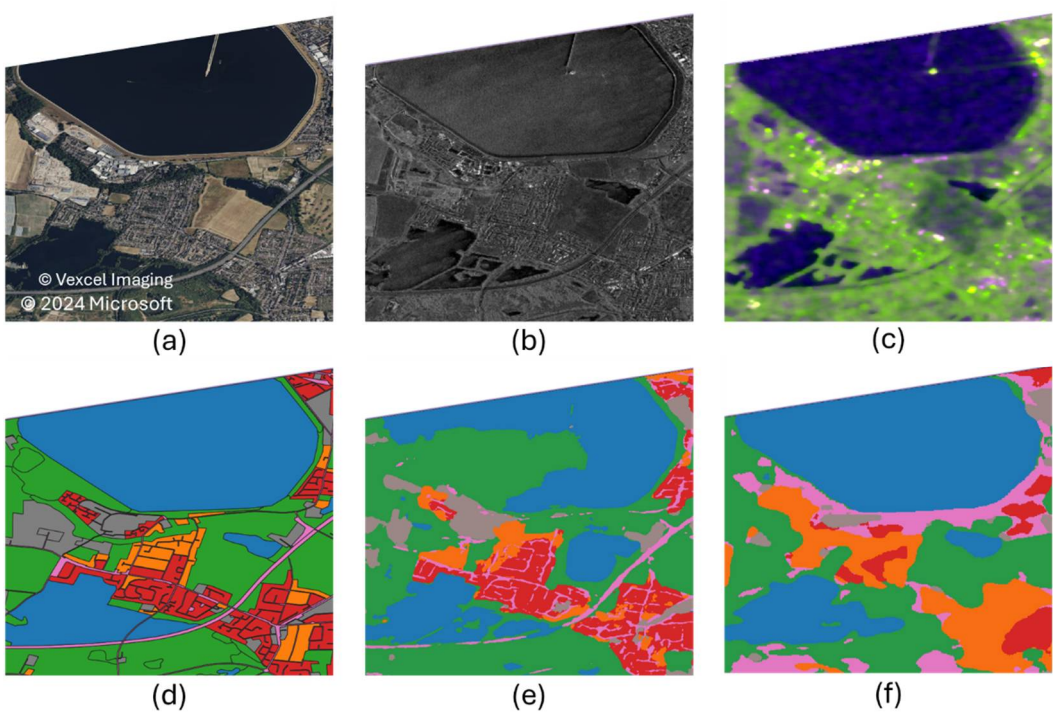


Figure 7.12. One of the largest reservoirs of fresh water in London, the Queen Mary Reservoir. (a) Optical image from Bing Maps Satellite Imagery [28], (b) X-band log intensity SAR image in VV channel, (c) C-band log intensity SAR image in the color composite of R: VV, G: VH and B: VV-VH, (d) UA labels, (e) prediction from Unet on X-band Log intensity SAR image and (f) prediction from Unet on C-band Log intensity SAR image.

7.5.3 Reliability of Urban Atlas

Objects labeled in the UA dataset were affected by their patterns of land use distribution and the organization of city blocks. This labeling process which follows the function of the land complicates the task of classifying different physical appearances as similar classes. Take example in Figure 7.2, in the optical image there are visible structures in the river slightly north,

which are Teddington Lock and Teddington Weir. Those structures, however, were not categorized as specific classes and were similarly labeled as Water.

Another large object with non-homogenous patterns is Heathrow Airport. In Figure 7.13, the SAR intensity image from X-band and C-band shows the two parallel runways facing the West-East direction represented in dark color due to specular scattering. Bright patterns in the center represent complex infrastructure such as airport terminals, bus stations, and hotels. Classifying these complex and different patterns as the same class will be difficult for any algorithm.

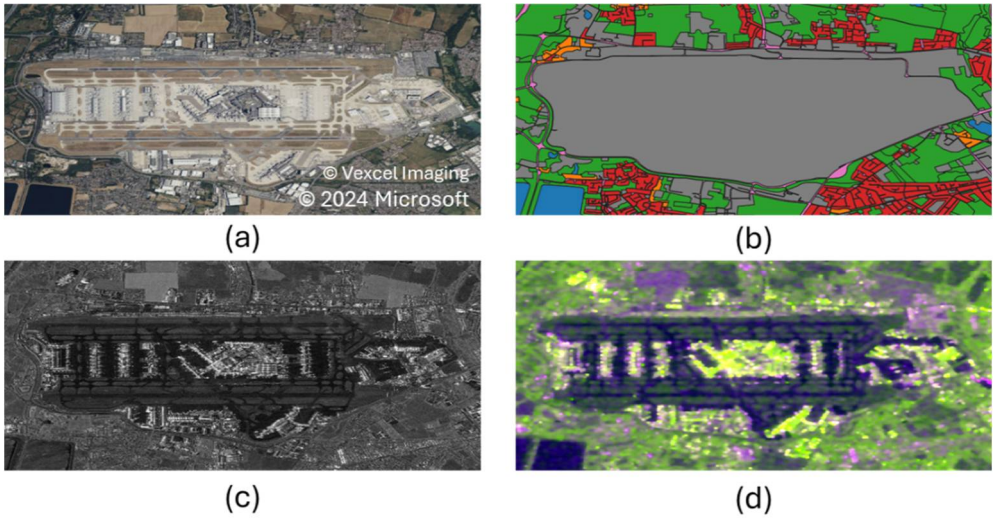


Figure 7.13. Heathrow Airport shown in (a) Optical image from Bing Maps Satellite Imagery [28], (b) UA labels where it is included as class Industrial area, colored in gray, (c) X-band log intensity SAR image in VV channel, and (d) C-band log intensity SAR image in the color composite of R: VV, G: VH and B: VV-VH.

Based on the Urban Atlas mapping guidelines [176], urban density can be categorized based on the degree of soil sealing, or the covering of the ground by impermeable material. The High density class used in this study is defined by the Continuous Urban Fabric class from UA, with >80% of soil sealing. Meanwhile the Medium density class were aggregated based on the Discontinuous Urban Fabric classes, which have different degrees of soil sealing based on imperviousness layer [188]. This complex label scheme can be challenging for the interpreter to evaluate. An example of inconsistency is shown in Figure 7.14. A selected polygon categorized as >80% of soil sealing from UA is shown in cyan. The local building footprint dataset from the National Topographic Database (BDOT10k), highlighted in red, shows the real estimate of only 30% soil sealing.

Figure 7.14. Aerial orthophoto of Warsaw from ESRI World Imagery [197] and a selected UA polygon (in cyan) labeled as Continuous Urban Fabric (>80%). Highlighted in red are building footprints from BDOT10k dataset. In this example, there is roughly only 30% occupied of artificial surfaces.

7.5.4 Discussion on accuracy

Zhu et al. [198] tested the classification of urban land cover types such as Low density residential, High density residential, and Commercial/industrial based on PALSAR and optical data. SAR data inclusion improved the overall classification by 1.1%. Relatively high producer's (80.83%) and user's (74.68%) accuracies were observed for High density residential, while the producer's and user's accuracies for Low density residential and Commercial/industrial were below average (approximately 70% or less). The Low density residential class was frequently misclassified as Forest. Commercial/industrial was sometimes misclassified as High density residential [198]. Similar results of K-means classification on 0.32-0.45 level of mIoU were reached on UAVSAR data for three different urban areas [193]. Results from Corbane [199] show that SAR backscatter from an urban environment is highly dependent on radar frequency, polarization, and viewing geometry. Therefore SAR imagery allows detecting urban features in a complementary way, but it can also become blind towards other buildings and structures depending on the viewing geometry, the incidence angle, and the urban fabric [199], [200].

7.6 Conclusion

In this study, the single polarization X-band and dual polarization C-band SAR were compared for LULC classification in urban areas. Results show that X-band with higher detail is more suitable for urban analysis despite more SAR features being present in the dual polarization C-band.

The supervised segmentation using Unet was significantly better at classifying the land classes compared to the unsupervised K-means clustering. The low separability of classes in every SAR feature is reflected in the poor performance of K-means, achieving the best performance at 0.2379 mIoU using the C-band GLCM features. Meanwhile, Unet obtained the best performance at 0.4742 mIoU using X-band log-intensity feature. Other derived SAR features did not improve the score of Unet, most likely due to the robustness of convolutional neural networks as feature extractors. The use of UA as a reference source is rather limited but incorporated with data augmentation methods could improve its potential for training algorithms for LULC classification using a single polarization SAR image. **To summarize, it can be concluded that the last thesis of the dissertation has been confirmed**.

8 Conclusion

As laid out in the introduction, this work has been devoted to verifying the feasibility of using deep learning algorithms to develop a monitoring system for disaster mitigation. Large events that result in significant changes in the urban landscape can be detected using SAR satellite imagery with global coverage. Once detected, a SAR instrument with a smaller swath but improved spatial resolution can provide input data for the classification and localization of man-made infrastructures. The continuous monitoring and automated analysis benefits from SAR technology that can measure through typical occlusions and poor weather conditions that follow a disastrous event.

8.1 Research summary

In this thesis, deep learning methods were explored for various urban analyses using SAR images. This section summarizes the key findings from experimental chapters in this dissertation, which include extraction of building footprints, large event detection from multitemporal data, and LULC classification.

Works in automated analysis of SAR data are closely related to progress in computer vision of natural images where more efficient or performant methods are tested out. This leads to a domain gap between natural images containing simple RGB channels to SAR images representing radar echoes. A closer domain gap was demonstrated to improve transfer learning techniques, where pre-trained weights from optical remote sensing images yield better performance in SAR than using pre-trained weights directly from natural images [49].

To solve the issue of limited training data that causes overfit in supervised learning, the effects of various data augmentation methods on SAR were explored. Pixel-based transformations in SAR were not as effective as in natural-colored images. Geometrical transformations were shown to be effective at delaying overfit except for vertical flip and corner rotations, which cause extreme displacement of shadows and layovers compared to the building footprint labels [15].

The increase in many SAR satellites results in an abundance of unlabeled remote sensing data. To take advantage of unlabeled data, multitemporal Sentinel-1 images were used to train an autoencoder in an unsupervised way. The ability to detect general events was demonstrated by the autoencoder, trained only on SAR images of flood events, and was able to identify changes in SAR images of wildfires and landslides. The autoencoder can learn representations

of the training data by reconstructing the input without any labels. The distances in representations from a pair of adjacent timeframe images were used to identify changes between them [18], [19].

The neural network considers not only spectral and textural features, but also geometric and multiscale neighboring information. This was demonstrated in the task of LULC classification where the importance of several SAR features as input to a classifier was analyzed. A clustering algorithm with richer textural feature inputs had improved detection for minority classes, while for neural networks, derived features did not improve overall performance. Despite relying only on a single polarization VHR SAR, the high details provide more features for identifying man-made structures. The solution was tested on two urban areas with diverse topographical structures, yielding the best performance of 0.4742 mIoU.

In general, the DL algorithms proposed in this study demonstrate the feasibility of automated analysis using SAR images. The various urban landscape and sensor configuration validates the generalization capability of the algorithm.

8.2 Future work

Due to limited data acquisition, methods in this dissertation were not explored yet using other SAR modes, such as polarimetric, interferometric, or a combination of both. As reviewed in Chapter 4, information on scattering mechanisms can better distinguish objects in urban scenes. Collecting multitemporal data spanning an event will be challenging, but if such an opportunity exists, it will be a valuable resource for disaster analysis using SAR.

Current research trends in the computer vision field are moving towards Semi-Supervised Learning (SSL) which can take advantage of the abundance of unlabeled data in SAR. This process though, is known to require significant computing resources for multi-day training even using GPU clusters. There is a chance that large companies that have the resources could develop a foundation model on SAR using SSL methods. This could significantly reduce resources as practitioners can further fine-tune those pre-train foundation models in SAR-specific tasks.

Alternatively, more labeled data can be generated using a SAR simulator, which can obtain various acquisition modes that would be expensive to perform in real-world scenarios. Variability can significantly improve the generalization capabilities of neural networks. Potentially expanding beyond the image layer to understanding the underlying physical models [201].

References

[1] USGS, “M7.8 and M7.5 Kahramanmaraş Earthquake Sequence near Nurdağı, Turkey (Türkiye) | U.S. Geological Survey.” Accessed: Apr. 20, 2024. [Online]. Available: https://www.usgs.gov/news/featured-story/m78-and-m75-kahramanmaras-earthquake-sequence-near-nurdagi-turkey-turkiye

[2] UNDP, “Türkiye-Syria earthquakes,” UNDP. Accessed: Apr. 20, 2024. [Online]. Available: https://www.undp.org/turkiye-syria-earthquakes

[3] L. Dal Zilio and J.-P. Ampuero, “Earthquake doublet in Turkey and Syria,” *Commun Earth Environ*, vol. 4, no. 1, pp. 1–4, Mar. 2023, doi: 10.1038/s43247-023-00747-z.

[4] A. England, A. Smith, G. Parrish, and S. Bernard, “Turkey and Syria’s devastating earthquakes in graphics,” Financial Times. Accessed: May 17, 2024. [Online]. Available: https://www.ft.com/content/337edef6-05c9-498c-a3f0-13776082f218

[5] Maxar, “Maxar Open Data Program.” Accessed: Jul. 31, 2024. [Online]. Available: https://www.maxar.com/open-data

[6] Capella, “Capella Space Open Data Gallery,” Capella Space. Accessed: Jul. 31, 2024. [Online]. Available: https://www.capellaspace.com/gallery/

[7] Umbra, “Open Data Program • Umbra Space.” Accessed: Jul. 31, 2024. [Online]. Available: https://umbra.space/open-data/

[8] ICEYE, “Example SAR data from ICEYE.” Accessed: Jul. 31, 2024. [Online]. Available: https://www.iceye.com/resources/datasets

[9] “Tropical Cyclone Freddy - Feb 2023 | ReliefWeb.” Accessed: May 17, 2024. [Online]. Available: https://reliefweb.int/disaster/tc-2023-000023-mdg

[10] L. Scott, “SAR data over Tropical Cyclone Freddy in the Indian Ocean.” Accessed: Sep. 28, 2024. [Online]. Available: https://cimss.ssec.wisc.edu/satellite-blog/archives/50355

[11] K. Fedra, “Urban environmental management: monitoring, GIS, and modeling,” *Computers, Environment and Urban Systems*, vol. 23, no. 6, pp. 443–457, Nov. 1999, doi: 10.1016/S0198-9715(99)00038-1.

[12] K. Molch, “Radar Earth Observation Imagery for Urban Area Characterisation,” JRC Publications Repository. Accessed: Jun. 28, 2024. [Online]. Available: https://publications.jrc.ec.europa.eu/repository/handle/JRC50451

[13] A. Mullissa *et al.*, “Sentinel-1 SAR Backscatter Analysis Ready Data Preparation in Google Earth Engine,” *Remote Sensing*, vol. 13, no. 10, p. 1954, May 2021, doi: 10.3390/rs13101954.

[14] M. Claverie *et al.*, “The Harmonized Landsat and Sentinel-2 surface reflectance data set,” *Remote Sensing of Environment*, vol. 219, pp. 145–161, Dec. 2018, doi: 10.1016/j.rse.2018.09.002.

[15] S. Wangiyana, P. Samczyński, and A. Gromek, “Data Augmentation for Building Footprint Segmentation in SAR Images: An Empirical Study,” *Remote Sensing*, vol. 14, no. 9, Art. no. 9, Jan. 2022, doi: 10.3390/rs14092012.

[16] S. Wangiyana, P. Samczynski, and A. Gromek, “Effects of SAR Resolution in Automatic Building Segmentation Using CNN,” in *2021 Signal Processing Symposium (SPSympo)*, LODZ, Poland: IEEE, Sep. 2021, pp. 289–293. doi: 10.1109/SPSympo51155.2020.9593636.

[17] S. Wangiyana, “CNN Performance Analysis for SAR Object Classification,” p. 3.

[18] S. Wangiyana, “Flood Detection Using Variational Autoencoder in SAR Images,” in *2023 Signal Processing Symposium (SPSympo)*, Sep. 2023, pp. 195–197. doi: 10.23919/SPSympo57300.2023.10302703.

[19] S. Wangiyana, “Unsupervised SAR Change Detection Using Autoencoders,” in *2024 International Radar Symposium (IRS)*, Jul. 2024, pp. 140–143. Accessed: Sep. 28, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10644931

[20] M. A. Richards, J. Scheer, W. A. Holm, and W. L. Melvin, Eds., *Principles of modern radar*. Raleigh, NC: SciTech Pub, 2010.

[21] C. Oliver and S. Quegan, Eds., *Understanding synthetic aperture radar images*. Raleigh, NC: SciTech Publishing, 2004.

[22] S.-W. Chen, X.-S. Wang, S.-P. Xiao, and M. Sato, *Target Scattering Mechanism in Polarimetric Synthetic Aperture Radar*. Singapore: Springer Singapore, 2018. doi: 10.1007/978-981-10-7269-7.

[23] M. I. Skolnik, *Introduction to radar systems*, Third edition. in McGraw-Hill electrical engineering series. Boston, Mass. Burr Ridge, IL Dubuque, IA: McGraw Hill, 2001.

[24] B. Wang, *Digital Signal Processing Techniques and Applications in Radar Image Processing*, 1st ed. Wiley, 2008. doi: 10.1002/9780470377765.

[25] U. Soergel, Ed., *Radar Remote Sensing of Urban Areas*, vol. 15. in Remote Sensing and Digital Image Processing, vol. 15. Dordrecht: Springer Netherlands, 2010. doi: 10.1007/978-90-481-3751-0.

[26] I. Hajnsek and Y.-L. Desnos, Eds., *Polarimetric Synthetic Aperture Radar: Principles and Application*, vol. 25. in Remote Sensing and Digital Image Processing, vol. 25. Cham: Springer International Publishing, 2021. doi: 10.1007/978-3-030-56504-6.

[27] NASA JPL, “Interferometry | Get to Know SAR,” NASA-ISRO SAR Mission (NISAR). Accessed: Sep. 05, 2024. [Online]. Available: https://nisar.jpl.nasa.gov/mission/get-to-know-sar/interferometry

[28] Microsoft, “Bing Maps Satellite Imagery,” 2024.

[29] F. Eric Jameson, “SAR Interferometry for Earthquake Studies,” JPL, California Institute of Technology, Aug. 16, 2018.

[30] Y. Marwan, “Synthetic Aperture Radar (SAR): Principles and Applications,” presented at the 6th ESA Advanced Training Course on Land Remote Sensing, DLR, Microwave and Radar Institute, Sep. 14, 2015.

[31] F. Chollet, *Deep learning with Python*. Shelter Island, New York: Manning Publications Co, 2018.

[32] T. M. Mitchell, *Machine Learning*. McGraw-Hill, 1997.

[33] N. Buduma, N. Buduma, and Papa Joe, *Fundamentals of deep learning: designing next-generation machine intelligence algorithms*, Second edition. Beijing Boston Farnham Sebastopol Tokyo: O’Reilly, 2022.

[34] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. in Adaptive computation and machine learning. Cambridge, Massachusetts: The MIT Press, 2016.

[35] V. Nair and G. E. Hinton, “Rectified Linear Units Improve Restricted Boltzmann Machines”.

[36] J. Brownlee, “A Gentle Introduction to Cross-Entropy for Machine Learning,” MachineLearningMastery.com. Accessed: May 30, 2024. [Online]. Available: https://machinelearningmastery.com/cross-entropy-for-machine-learning/

[37] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2012. Accessed: May 31, 2024. [Online]. Available: https://papers.nips.cc/paper_files/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html

[39] O. Russakovsky *et al.*, “ImageNet Large Scale Visual Recognition Challenge,” Jan. 29, 2015, *arXiv*: arXiv:1409.0575. doi: 10.48550/arXiv.1409.0575.

[40] O. Russakovsky *et al.*, “ImageNet Large Scale Visual Recognition Challenge,” *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015, doi: 10.1007/s11263-015-0816-y.

[41] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, “Rethinking Atrous Convolution for Semantic Image Segmentation,” 2017.

[42] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 6230–6239, 2017, doi: 10.1109/CVPR.2017.660.

[43] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 936–944, 2017, doi: 10.1109/CVPR.2017.106.

[44] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” Dec. 10, 2015, *arXiv*: arXiv:1512.03385. doi: 10.48550/arXiv.1512.03385.

[45] C. Szegedy *et al.*, “Going Deeper with Convolutions,” Sep. 16, 2014, *arXiv*: arXiv:1409.4842. doi: 10.48550/arXiv.1409.4842.

[46] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” *36th International Conference on Machine Learning, ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.

[47] Y. Mo, Y. Wu, X. Yang, F. Liu, and Y. Liao, “Review the state-of-the-art technologies of semantic segmentation based on deep learning,” *Neurocomputing*, vol. 493, pp. 626–646, Jul. 2022, doi: 10.1016/j.neucom.2022.01.005.

[48] O. Elharrouss, Y. Akbari, N. Almaadeed, and S. Al-Maadeed, “Backbones-Review: Feature Extraction Networks for Deep Learning and Deep Reinforcement Learning Approaches,” *Computer Science Review*, vol. 53, p. 100645, Aug. 2024, doi: 10.1016/j.cosrev.2024.100645.

[49] S. Wangiyana, P. Samczynski, and A. Gromek, “Effects of SAR Resolution in Automatic Building Segmentation Using CNN,” in *2021 Signal Processing Symposium (SPSympo)*, IEEE, Sep. 2021, pp. 289–293. doi: 10.1109/SPSympo51155.2020.9593636.

[50] S. Seferbekov, V. Iglovikov, A. Buslaev, and A. Shvets, “Feature pyramid network for multi-class land segmentation,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 272–275. doi: 10.1109/CVPRW.2018.00051.

[51] M. Buda, A. Maki, and M. A. Mazurowski, “A systematic study of the class imbalance problem in convolutional neural networks,” *Neural Networks*, vol. 106, pp. 249–259, Oct. 2018, doi: 10.1016/j.neunet.2018.07.011.

[52] “Precision and recall,” *Wikipedia*. Aug. 24, 2024. Accessed: Sep. 26, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Precision_and_recall&oldid=1242064846

[53] F. L. De Sousa, “Are smallsats taking over bigsats for land Earth observation?,” *Acta Astronautica*, vol. 213, pp. 455–463, Dec. 2023, doi: 10.1016/j.actaastro.2023.09.041.

[54] L. An, J. Zhang, and L. Gong, “EARTHQUAKE BUILDING DAMAGE MAPPING BASED ON FEATURE ANALYZING METHOD FROM SYNTHETIC APERTURE RADAR DATA,” *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. XLII–3, pp. 39–43, Apr. 2018, doi: 10.5194/isprs-archives-XLII-3-39-2018.

[55] P. Ge, H. Gokon, and K. Meguro, “A review on synthetic aperture radar-based building damage assessment in disasters,” *Remote Sensing of Environment*, vol. 240, p. 111693, Apr. 2020, doi: 10.1016/j.rse.2020.111693.

[56] L. Zhao, X. Zhou, and G. Kuang, “Building detection from urban SAR image using building characteristics and contextual information,” *EURASIP Journal on Advances in Signal Processing*, vol. 2013, no. 1, p. 56, Mar. 2013, doi: 10.1186/1687-6180-2013-56.

[57] S. Karimzadeh and M. Mastuoka, “Building Damage Assessment Using Multisensor Dual-Polarized Synthetic Aperture Radar Data for the 2016 M 6.2 Amatrice Earthquake, Italy,” *Remote Sensing*, vol. 9, no. 4, Art. no. 4, Apr. 2017, doi: 10.3390/rs9040330.

[58] M. Matsuoka, S. Koshimura, and N. Nojima, “Estimation of building damage ratio due to earthquakes and tsunamis using satellite SAR imagery,” in *2010 IEEE International Geoscience and Remote Sensing Symposium*, Jul. 2010, pp. 3347–3349. doi: 10.1109/IGARSS.2010.5650550.

[59] H. Wang and Y.-Q. Jin, “Statistical analysis to assess building damage in 2008 Wenchuan earthquake from multi-temporal SAR images,” in *2009 2nd Asian-Pacific Conference on Synthetic Aperture Radar*, Oct. 2009, pp. 121–123. doi: 10.1109/APSAR.2009.5374142.

[60] Q. Chen, H. Yang, L. Li, and X. Liu, “A Novel Statistical Texture Feature for SAR Building Damage Assessment in Different Polarization Modes,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 154–165, 2020, doi: 10.1109/JSTARS.2019.2954292.

[61] R. Touzi, A. Lopes, J. Bruniquel, and P. W. Vachon, “Coherence estimation for SAR imagery,” *IEEE Trans. Geosci. Remote Sensing*, vol. 37, no. 1, pp. 135–149, Jan. 1999, doi: 10.1109/36.739146.

[62] S.-H. Yun *et al.*, “Rapid Damage Mapping for the 2015 Mw 7.8 Gorkha Earthquake Using Synthetic Aperture Radar Data from COSMO–SkyMed and ALOS-2 Satellites,” *Seismological Research Letters*, vol. 86, no. 6, pp. 1549–1556, Nov. 2015, doi: 10.1785/0220150152.

[63] K. Burrows, R. J. Walters, D. Milledge, K. Spaans, and A. L. Densmore, “A New Method for Large-Scale Landslide Classification from Satellite Radar,” *Remote Sensing*, vol. 11, no. 3, p. 237, Jan. 2019, doi: 10.3390/rs11030237.

[64] S.-H. Yun, E. J. Fielding, F. H. Webb, and M. Simons, “Damage proxy map from interferometric synthetic aperture radar coherence,” US9207318B2, Dec. 08, 2015 Accessed: Jul. 15, 2024. [Online]. Available: https://patents.google.com/patent/US9207318B2/en

[65] R. C. Sharma, R. Tateishi, K. Hara, H. T. Nguyen, S. Gharechelou, and L. V. Nguyen, “Earthquake Damage Visualization (EDV) Technique for the Rapid Detection of Earthquake-Induced Damages Using SAR Data,” *Sensors*, vol. 17, no. 2, Art. no. 2, Feb. 2017, doi: 10.3390/s17020235.

[66] G. Sinclair, “The Transmission and Reception of Elliptically Polarized Waves,” *Proceedings of the IRE*, vol. 38, no. 2, pp. 148–151, Feb. 1950, doi: 10.1109/JRPROC.1950.230106.

[67] S. R. Cloude and E. Pottier, “A review of target decomposition theorems in radar polarimetry,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 34, no. 2, pp. 498–518, Mar. 1996, doi: 10.1109/36.485127.

[68] A. Freeman and S. L. Durden, “A three-component scattering model for polarimetric SAR data,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 36, no. 3, pp. 963–973, May 1998, doi: 10.1109/36.673687.

[69] Y. Yamaguchi, T. Moriyama, M. Ishido, and H. Yamada, "Four-component scattering model for polarimetric SAR image decomposition," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 8, pp. 1699–1706, Aug. 2005, doi: 10.1109/TGRS.2005.852084.

[70] Y. Yamaguchi, "Disaster Monitoring by Fully Polarimetric SAR Data Acquired With ALOS-PALSAR," *Proceedings of the IEEE*, vol. 100, no. 10, pp. 2851–2860, Oct. 2012, doi: 10.1109/JPROC.2012.2195469.

[71] G. Singh, Y. Yamaguchi, W.-M. Boerner, and S.-E. Park, "Monitoring of the March 11, 2011, Off-Tohoku 9.0 Earthquake With Super-Tsunami Disaster by Implementing Fully Polarimetric High-Resolution POLSAR Techniques," *Proceedings of the IEEE*, vol. 101, no. 3, pp. 831–846, Mar. 2013, doi: 10.1109/JPROC.2012.2230311.

[72] S.-W. Chen and M. Sato, "Tsunami Damage Investigation of Built-Up Areas Using Multitemporal Spaceborne Full Polarimetric SAR Images," *IEEE Trans. Geosci. Remote Sensing*, vol. 51, no. 4, pp. 1985–1997, Apr. 2013, doi: 10.1109/TGRS.2012.2210050.

[73] F. Mattia *et al.*, "The effect of surface roughness on multifrequency polarimetric SAR data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 35, no. 4, pp. 954–966, Jul. 1997, doi: 10.1109/36.602537.

[74] J.-S. Lee, D. L. Schuler, T. L. Ainsworth, E. Krogager, D. Kasilingam, and W.-M. Boerner, "On the estimation of radar polarization orientation shifts induced by terrain slopes," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 1, pp. 30–41, Jan. 2002, doi: 10.1109/36.981347.

[75] W. Zhai and C. Huang, "Fast building damage mapping using a single post-earthquake PolSAR image: a case study of the 2010 Yushu earthquake," *Earth, Planets and Space*, vol. 68, no. 1, p. 86, May 2016, doi: 10.1186/s40623-016-0469-2.

[76] W. Zhai, C. Huang, and W. Pei, "Building Damage Assessment Based on the Fusion of Multiple Texture Features Using a Single Post-Earthquake PolSAR Image," *Remote Sensing*, vol. 11, no. 8, Art. no. 8, Jan. 2019, doi: 10.3390/rs11080897.

[77] Z. Xu, R. Wang, H. Zhang, N. Li, and L. Zhang, "Building extraction from high-resolution SAR imagery based on deep neural networks," *Remote Sensing Letters*, vol. 8, no. 9, pp. 888–896, Sep. 2017, doi: 10.1080/2150704X.2017.1335906.

[78] M. Shahzad, M. Maurer, F. Fraundorfer, Y. Wang, and X. X. Zhu, "Buildings Detection in VHR SAR Images Using Fully Convolution Neural Networks," *IEEE Trans. Geosci. Remote Sensing*, vol. 57, no. 2, pp. 1100–1116, Feb. 2019, doi: 10.1109/TGRS.2018.2864716.

[79] B. Adriano *et al.*, "Learning from multimodal and multitemporal earth observation data for building damage mapping," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 175, pp. 132–143, May 2021, doi: 10.1016/j.isprsjprs.2021.02.016.

[80] J. Shermeyer *et al.*, "SpaceNet 6: Multi-Sensor All Weather Mapping Dataset," Apr. 14, 2020, *arXiv*: arXiv:2004.06500. Accessed: Jul. 17, 2023. [Online]. Available: http://arxiv.org/abs/2004.06500

[81] X. Li *et al.*, "Progressive fusion learning: A multimodal joint segmentation framework for building extraction from optical and SAR images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 195, pp. 178–191, Jan. 2023, doi: 10.1016/j.isprsjprs.2022.11.015.

[82] R. Gupta *et al.*, "xBD: A Dataset for Assessing Building Damage from Satellite Imagery," Nov. 21, 2019, *arXiv*: arXiv:1911.09296. Accessed: Jul. 21, 2022. [Online]. Available: http://arxiv.org/abs/1911.09296

[83] "Hazus Hurricane Model User Guidance".

[84] L. Deng and Y. Wang, "Post-disaster building damage assessment based on improved U-Net," *Sci Rep*, vol. 12, no. 1, p. 15862, Sep. 2022, doi: 10.1038/s41598-022-20114-w.

[85] H. Chen, E. Nemni, S. Vallecorsa, X. Li, C. Wu, and L. Bromley, "Dual-Tasks Siamese Transformer Framework for Building Damage Assessment," May 28, 2022, *arXiv*: arXiv:2201.10953. doi: 10.48550/arXiv.2201.10953.

[86] W. Lu, L. Wei, and M. Nguyen, "Bitemporal Attention Transformer for Building Change Detection and Building Damage Assessment," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 4917–4935, 2024, doi: 10.1109/JSTARS.2024.3354310.

[87] R. Gupta and M. Shah, "RescueNet: Joint Building Segmentation and Damage Assessment from Satellite Imagery," Apr. 15, 2020, *arXiv*: arXiv:2004.07312. Accessed: Jul. 16, 2024. [Online]. Available: http://arxiv.org/abs/2004.07312

[88] C. M. Gevaert and M. Belgiu, "Assessing the generalization capability of deep learning networks for aerial image classification using landscape metrics," *International Journal of Applied Earth Observation and Geoinformation*, vol. 114, p. 103054, Nov. 2022, doi: 10.1016/j.jag.2022.103054.

[89] S. Wiguna, B. Adriano, E. Mas, and S. Koshimura, "Evaluation of Deep Learning Models for Building Damage Mapping in Emergency Response Settings," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 5651–5667, 2024, doi: 10.1109/JSTARS.2024.3367853.

[90] V. Benson and A. S. Ecker, "Assessing out-of-domain generalization for robust building damage detection," *ArXiv*, 2020.

[91] D. Melamed *et al.*, "xFBD: Focused Building Damage Dataset and Analysis," Feb. 15, 2023, *arXiv*: arXiv:2212.13876. doi: 10.48550/arXiv.2212.13876.

[92] W. Yang, X. Zhang, and P. Luo, "Transferability of Convolutional Neural Network Models for Identifying Damaged Buildings Due to Earthquake," *Remote Sensing*, vol. 13, no. 3, Art. no. 3, Jan. 2021, doi: 10.3390/rs13030504.

[93] B. Adriano, N. Yokoya, J. Xia, G. Baier, and S. Koshimura, "Cross-Domain-Classification of Tsunami Damage Via Data Simulation and Residual-Network-Derived Features From Multi-Source Images," in *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, Jul. 2019, pp. 4947–4950. doi: 10.1109/IGARSS.2019.8899155.

[94] S. Gholami *et al.*, "On the Deployment of Post-Disaster Building Damage Assessment Tools using Satellite Imagery: A Deep Learning Approach," in *2022 IEEE International Conference on Data Mining Workshops (ICDMW)*, Nov. 2022, pp. 1029–1036. doi: 10.1109/ICDMW58026.2022.00134.

[95] P. Uprety, F. Yamazaki, and F. Dell'Acqua, "Damage Detection Using High-Resolution SAR Imagery in the 2009 L'Aquila, Italy, Earthquake," *Earthquake Spectra*, vol. 29, no. 4, pp. 1521–1535, Nov. 2013, doi: 10.1193/060211EQS126M.

[96] H. Miura, S. Midorikawa, and M. Matsuoka, "Building Damage Assessment Using High-Resolution Satellite SAR Images of the 2010 Haiti Earthquake," *Earthquake Spectra*, vol. 32, no. 1, pp. 591–610, Feb. 2016, doi: 10.1193/033014EQS042M.

[97] P. T. B. Brett and R. Guida, "Earthquake Damage Detection in Urban Areas Using Curvilinear Features," *IEEE Trans. Geosci. Remote Sensing*, vol. 51, no. 9, pp. 4877–4884, Sep. 2013, doi: 10.1109/TGRS.2013.2271564.

[98] Y. Bai *et al.*, "A Framework of Rapid Regional Tsunami Damage Recognition From Post-event TerraSAR-X Imagery Using Deep Neural Networks," *IEEE Geosci. Remote Sensing Lett.*, vol. 15, no. 1, pp. 43–47, Jan. 2018, doi: 10.1109/LGRS.2017.2772349.

[99] A. Rao, J. Jung, V. Silva, G. Molinario, and S.-H. Yun, "Earthquake building damage detection based on synthetic-aperture-radar imagery and machine learning," *Natural Hazards and Earth System Sciences*, vol. 23, no. 2, pp. 789–807, Feb. 2023, doi: 10.5194/nhess-23-789-2023.

[100] A. Fujita, K. Sakurada, T. Imaizumi, R. Ito, S. Hikosaka, and R. Nakamura, "Damage detection from aerial images via convolutional neural networks," in *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, May 2017, pp. 5–8. doi: 10.23919/MVA.2017.7986759.

[101] G. Christie, N. Fendley, J. Wilson, and R. Mukherjee, "Functional Map of the World," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA: IEEE, Jun. 2018, pp. 6172–6180. doi: 10.1109/CVPR.2018.00646.

[102] T. Chowdhury, R. Murphy, and M. Rahnemoonfar, *RescueNet: A High Resolution UAV Semantic Segmentation Benchmark Dataset for Natural Disaster Damage Assessment.* 2022.

[103] H. Li, F. Zhu, X. Zheng, M. Liu, and G. Chen, "MSCDUNet: A Deep Learning Framework for Built-Up Area Change Detection Integrating Multispectral, SAR, and VHR Data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 5163–5176, 2022, doi: 10.1109/JSTARS.2022.3181155.

[104] Y. Sun, Y. Wang, and M. Eineder, "QuickQuakeBuildings: Post-earthquake SAR-Optical Dataset for Quick Damaged-building Detection," Dec. 11, 2023, *arXiv*: arXiv:2312.06587. Accessed: Apr. 07, 2024. [Online]. Available: http://arxiv.org/abs/2312.06587

[105] Google Earth Engine, "PALSAR-2 ScanSAR Level 2.2 | Earth Engine Data Catalog," Google for Developers. Accessed: Jul. 31, 2024. [Online]. Available:

https://developers.google.com/earth-engine/datasets/catalog/JAXA_ALOS_PALSAR-2_Level2_2_ScanSAR

[106] European Union, “Copernicus Emergency Management Service,” Copernicus EMS. Accessed: Aug. 12, 2024. [Online]. Available: http://emergency.copernicus.eu/

[107] NASA JPL, “ARIA Project.” Accessed: Aug. 12, 2024. [Online]. Available: https://aria.jpl.nasa.gov/

[108] “International Disasters Charter.” Accessed: Aug. 12, 2024. [Online]. Available: https://disasterscharter.org/web/guest/home

[109] “Sentinel Asia.” Accessed: Aug. 12, 2024. [Online]. Available: https://sentinel-asia.org/

[110] F. Biljecki, Y. S. Chow, and K. Lee, “Quality of crowdsourced geospatial building information: A global assessment of OpenStreetMap attributes,” *Building and Environment*, vol. 237, p. 110295, Jun. 2023, doi: 10.1016/j.buildenv.2023.110295.

[111] C. Westrope, R. Banick, and M. Levine, “Groundtruthing OpenStreetMap Building Damage Assessment,” *Procedia Engineering*, vol. 78, pp. 29–39, 2014, doi: 10.1016/j.proeng.2014.07.035.

[112] X. X. Zhu, Q. Li, Y. Shi, Y. Wang, A. Stewart, and J. Prexl, “GlobalBuildingMap -- Unveiling the Mystery of Global Buildings,” May 22, 2024, *arXiv*: arXiv:2404.13911. Accessed: Aug. 12, 2024. [Online]. Available: http://arxiv.org/abs/2404.13911

[113] Microsoft, “Microsoft Building Footprints.” Accessed: Aug. 12, 2024. [Online]. Available: https://planetarycomputer.microsoft.com/dataset/ms-buildings

[114] W. Sirko *et al.*, “Continental-Scale Building Detection from High Resolution Satellite Imagery,” Jul. 29, 2021, *arXiv*: arXiv:2107.12283. Accessed: Jul. 16, 2024. [Online]. Available: http://arxiv.org/abs/2107.12283

[115] Z. Xia, Z. Li, Y. Bai, J. Yu, and B. Adriano, “Self-Supervised Learning for Building Damage Assessment from Large-scale xBD Satellite Imagery Benchmark Datasets,” Jun. 29, 2022, *arXiv*: arXiv:2205.15688. Accessed: Nov. 28, 2022. [Online]. Available: http://arxiv.org/abs/2205.15688

[116] Y. Li *et al.*, “SARDet-100K: Towards Open-Source Benchmark and ToolKit for Large-Scale SAR Object Detection,” Mar. 11, 2024, *arXiv*: arXiv:2403.06534. Accessed: Sep. 05, 2024. [Online]. Available: http://arxiv.org/abs/2403.06534

[117] I. G. Rizaev and A. Achim, “SynthWakeSAR: A Synthetic SAR Dataset for Deep Learning Classification of Ships at Sea,” *Remote Sensing*, vol. 14, no. 16, p. 3999, Aug. 2022, doi: 10.3390/rs14163999.

[118] B. Camus, T. Voillemin, C. L. Barbu, J.-C. Louvigné, C. Belloni, and E. Vallée, “Training Deep Learning Models with Hybrid Datasets for Robust Automatic Target Detection on real SAR images”.

[119] S. Auer, T. Balz, S. Becker, and R. Bamler, “3D SAR Simulation of Urban Areas Based on Detailed Building Models,” *photogramm eng remote sensing*, vol. 76, no. 12, pp. 1373–1384, Dec. 2010, doi: 10.14358/PERS.76.12.1373.

[120] D. Hutchison *et al.*, "2.5D Dual Contouring: A Robust Approach to Creating Building Models from Aerial LiDAR Point Clouds," in *Computer Vision – ECCV 2010*, vol. 6313, K. Daniilidis, P. Maragos, and N. Paragios, Eds., in Lecture Notes in Computer Science, vol. 6313. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 115–128. doi: 10.1007/978-3-642-15558-1_9.

[121] R. Wang, S. Huang, and H. Yang, "Building3D: An Urban-Scale Dataset and Benchmarks for Learning Roof Structures from Point Clouds," Jul. 21, 2023, *arXiv*: arXiv:2307.11914. Accessed: Aug. 11, 2024. [Online]. Available: http://arxiv.org/abs/2307.11914

[122] C. Y. Ho, E. Mas, B. Adriano, and S. Koshimura, "Exploring the Feasibility of Ray Tracing SAR Simulation on Building Damage Assessment," *IEEE J. Sel. Top. Appl. Earth Observations Remote Sensing*, vol. 17, pp. 1046–1059, 2024, doi: 10.1109/JSTARS.2024.3418412.

[123] Y. Li, W. Hu, H. Li, H. Dong, B. Zhang, and Q. Tian, "Aligning Discriminative and Representative Features: An Unsupervised Domain Adaptation Method for Building Damage Assessment," *IEEE Transactions on Image Processing*, vol. 29, pp. 6110–6122, 2020, doi: 10.1109/TIP.2020.2988175.

[124] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A Simple Framework for Contrastive Learning of Visual Representations," Jun. 30, 2020, *arXiv*: arXiv:2002.05709. doi: 10.48550/arXiv.2002.05709.

[125] M. Muzeau, J. Frontera-Pons, C. Ren, and J.-P. Ovarlez, "SAFE: a SAR Feature Extractor based on self-supervised learning and masked Siamese ViTs," Jun. 30, 2024, *arXiv*: arXiv:2407.00851. doi: 10.48550/arXiv.2407.00851.

[126] H. Jing, X. Sun, Z. Wang, K. Chen, W. Diao, and K. Fu, "Fine Building Segmentation in High-Resolution SAR Images Via Selective Pyramid Dilated Network," *IEEE J. Sel. Top. Appl. Earth Observations Remote Sensing*, vol. 14, pp. 6608–6623, 2021, doi: 10.1109/JSTARS.2021.3076085.

[127] J. Xia, N. Yokoya, B. Adriano, L. Zhang, G. Li, and Z. Wang, "A Benchmark High-Resolution GaoFen-3 SAR Dataset for Building Semantic Segmentation," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, no. 8, pp. 5950–5963, 2021, doi: 10.1109/JSTARS.2021.3085122.

[128] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting Unreasonable Effectiveness of Data in Deep Learning Era," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2017-Octob, pp. 843–852, 2017, doi: 10.1109/ICCV.2017.97.

[129] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark," *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 2017-July, pp. 3226–3229, 2017, doi: 10.1109/IGARSS.2017.8127684.

[130] A. Van Etten, D. Lindenbaum, and T. M. Bacastow, "SpaceNet: A Remote Sensing Dataset and Challenge Series," 2018.

[131] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *J Big Data*, vol. 6, no. 1, p. 60, Dec. 2019, doi: 10.1186/s40537-019-0197-0.

[132] S. Illarionova, S. Nesteruk, D. Shadrin, V. Ignatiev, M. Pukalchik, and I. Oseledets, “Object-Based Augmentation for Building Semantic Segmentation: Ventura and Santa Rosa Case Study,” in *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, Montreal, BC, Canada: IEEE, Oct. 2021, pp. 1659–1668. doi: 10.1109/ICCVW54120.2021.00191.

[133] L. Yang *et al.*, “Semantic Segmentation Based on Temporal Features: Learning of Temporal–Spatial Information From Time-Series SAR Images for Paddy Rice Mapping,” *IEEE Trans. Geosci. Remote Sensing*, vol. 60, pp. 1–16, 2022, doi: 10.1109/TGRS.2021.3099522.

[134] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, “Random Erasing Data Augmentation,” Nov. 16, 2017, *arXiv*: arXiv:1708.04896. Accessed: Jul. 17, 2023. [Online]. Available: http://arxiv.org/abs/1708.04896

[135] T. Song, S. Kim, S. Kim, J. Lee, and K. Sohn, “Context-Preserving Instance-Level Augmentation and Deformable Convolution Networks for SAR Ship Detection,” Feb. 14, 2022, *arXiv*: arXiv:2202.06513. Accessed: May 21, 2023. [Online]. Available: http://arxiv.org/abs/2202.06513

[136] S. Seferbekov, V. Iglovikov, A. Buslaev, and A. Shvets, “Feature Pyramid Network for Multi-class Land Segmentation,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Salt Lake City, UT, USA: IEEE, Jun. 2018, pp. 272–2723. doi: 10.1109/CVPRW.2018.00051.

[137] P. Iakubovskii, *qubvel/segmentation_models.pytorch*. (Mar. 13, 2024). Python. Accessed: Mar. 13, 2024. [Online]. Available: https://github.com/qubvel/segmentation_models.pytorch

[138] D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–15, 2015.

[139] I. Loshchilov and F. Hutter, “SGDR: Stochastic Gradient Descent with Warm Restarts,” May 03, 2017, *arXiv*: arXiv:1608.03983. doi: 10.48550/arXiv.1608.03983.

[140] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. J. Cardoso, “Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations,” vol. 10553, 2017, pp. 240–248. doi: 10.1007/978-3-319-67558-9_28.

[141] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, “Albumentations: Fast and flexible image augmentations,” *Information (Switzerland)*, vol. 11, no. 2, pp. 1–20, 2020, doi: 10.3390/info11020125.

[142] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, “Random erasing data augmentation,” *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, pp. 13001–13008, 2020, doi: 10.1609/aaai.v34i07.7000.

[143] C. Oliver and S. Quegan, *Understanding Synthetic Aperture Radar Images*. in EngineeringPro collection. SciTech Publ., 2004.

[144] S. Parrilli, M. Poderico, C. V. Angelino, and L. Verdoliva, "A nonlocal SAR image denoising algorithm based on LLMMSE wavelet shrinkage," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 50, no. 2, pp. 606–616, 2012, doi: 10.1109/TGRS.2011.2161586.

[145] Z. Shi and K. B. Fung, "Comparison of digital speckle filters," *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 4, no. 3, pp. 2129–2133, 1994, doi: 10.1109/igarss.1994.399671.

[146] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, vol. 6, no. 1, 2019, doi: 10.1186/s40537-019-0197-0.

[147] G. Wang, W. Li, M. Aertsen, J. Deprest, S. Ourselin, and T. Vercauteren, "Test-time augmentation with uncertainty estimation for deep learning-based medical image segmentation," *Midl*, no. Midl, pp. 1–9, 2018.

[148] H. Hu and Y. Ban, "Unsupervised Change Detection in Multitemporal SAR Images Over Large Urban Areas," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 8, pp. 3248–3261, Aug. 2014, doi: 10.1109/JSTARS.2014.2344017.

[149] F. Bovolo and L. Bruzzone, "A detail-preserving scale-driven approach to change detection in multitemporal SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 12, pp. 2963–2972, Dec. 2005, doi: 10.1109/TGRS.2005.857987.

[150] J. Inglada and G. Mercier, "A New Statistical Similarity Measure for Change Detection in Multitemporal SAR Images and Its Extension to Multiscale Change Analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 5, pp. 1432–1445, May 2007, doi: 10.1109/TGRS.2007.893568.

[151] K. Conradsen, A. A. Nielsen, J. Schou, and H. Skriver, "A test statistic in the complex Wishart distribution and its application to change detection in polarimetric SAR data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 1, pp. 4–19, Jan. 2003, doi: 10.1109/TGRS.2002.808066.

[152] G. Moser and S. B. Serpico, "Generalized minimum-error thresholding for unsupervised change detection from SAR amplitude imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 10, pp. 2972–2982, Oct. 2006, doi: 10.1109/TGRS.2006.876288.

[153] M. Gong, L. Su, M. Jia, and W. Chen, "Fuzzy Clustering With a Modified MRF Energy Function for Change Detection in Synthetic Aperture Radar Images," *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 1, pp. 98–109, Feb. 2014, doi: 10.1109/TFUZZ.2013.2249072.

[154] P. Mastro, G. Masiello, C. Serio, and A. Pepe, "Change Detection Techniques with Synthetic Aperture Radar Images: Experiments with Random Forests and Sentinel-1 Observations," *Remote Sensing*, vol. 14, no. 14, Art. no. 14, Jan. 2022, doi: 10.3390/rs14143323.

[155] J. Hestness *et al.*, "Deep Learning Scaling is Predictable, Empirically," Dec. 01, 2017, *arXiv*: arXiv:1712.00409. doi: 10.48550/arXiv.1712.00409.

[156] Y. Wang, C. M. Albrecht, N. A. A. Braham, L. Mou, and X. X. Zhu, “Self-supervised Learning in Remote Sensing: A Review,” Jun. 27, 2022, *arXiv*: arXiv:2206.13188. Accessed: Jul. 22, 2022. [Online]. Available: http://arxiv.org/abs/2206.13188

[157] P. Baldi, “Autoencoders, Unsupervised Learning, and Deep Architectures,” in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, JMLR Workshop and Conference Proceedings, Jun. 2012, pp. 37–49. Accessed: Mar. 09, 2024. [Online]. Available: https://proceedings.mlr.press/v27/baldi12a.html

[158] P. Planinšič and D. Gleich, “Temporal Change Detection in SAR Images Using Log Cumulants and Stacked Autoencoder,” *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 2, pp. 297–301, Feb. 2018, doi: 10.1109/LGRS.2017.2786344.

[159] “Detecting Changes by Learning No Changes: Data-Enclosing-Ball Minimizing Autoencoders for One-Class Change Detection in Multispectral Imagery | IEEE Journals & Magazine | IEEE Xplore.” Accessed: Mar. 09, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/9870684

[160] G. Mateo-Garcia *et al.*, “Towards global flood mapping onboard low cost satellites with machine learning,” *Sci Rep*, vol. 11, no. 1, p. 7249, Mar. 2021, doi: 10.1038/s41598-021-86650-z.

[161] European Union, “Copernicus Emergency Management Service.” 2024. [Online]. Available: https://emergency.copernicus.eu/

[162] N. Gorelick, M. Hancher, M. Dixon, S. Ilyushchenko, D. Thau, and R. Moore, “Google Earth Engine: Planetary-scale geospatial analysis for everyone,” *Remote Sensing of Environment*, vol. 202, pp. 18–27, Dec. 2017, doi: 10.1016/j.rse.2017.06.031.

[163] V. Růžička *et al.*, “RaVÆn: unsupervised change detection of extreme events using ML on-board satellites,” *Sci Rep*, vol. 12, no. 1, Art. no. 1, Oct. 2022, doi: 10.1038/s41598-022-19437-5.

[164] “An analysis of the combined consequences of pluvial and fluvial flooding | Water Science & Technology | IWA Publishing.” Accessed: Mar. 13, 2024. [Online]. Available: https://iwaponline.com/wst/article-abstract/62/7/1491/16606/An-analysis-of-the-combined-consequences-of

[165] L. Xu, H. Zhang, C. Wang, B. Zhang, and M. Liu, “Crop Classification Based on Temporal Information Using Sentinel-1 SAR Time-Series Data,” *Remote Sensing*, vol. 11, no. 1, Art. no. 1, Jan. 2019, doi: 10.3390/rs11010053.

[166] S. Ioffe and C. Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” Mar. 02, 2015, *arXiv*: arXiv:1502.03167. doi: 10.48550/arXiv.1502.03167.

[167] M. J. Canty, *Image Analysis, Classification, and Change Detection in Remote Sensing: With Algorithms for Python*, 4th ed. Fourth edition. | Boca Raton, FL : CRC Press/Taylor & Francis Group, 2019.: CRC Press, 2019. doi: 10.1201/9780429464348.

[168] D. Bank, N. Koenigstein, and R. Giryes, “Autoencoders,” arXiv.org. Accessed: Jun. 01, 2024. [Online]. Available: https://arxiv.org/abs/2003.05991v2

[169] R. Yadav, A. Nascetti, H. Azizpour, and Y. Ban, “Unsupervised flood detection on SAR time series using variational autoencoder,” *International Journal of Applied Earth Observation and Geoinformation*, vol. 126, p. 103635, Feb. 2024, doi: 10.1016/j.jag.2023.103635.

[170] M. Vreugdenhil *et al.*, “Sensitivity of Sentinel-1 Backscatter to Vegetation Dynamics: An Austrian Case Study,” *Remote Sensing*, vol. 10, no. 9, Art. no. 9, Sep. 2018, doi: 10.3390/rs10091396.

[171] Y. Ban, P. Zhang, A. Nascetti, A. R. Bevington, and M. A. Wulder, “Near Real-Time Wildfire Progression Monitoring with Sentinel-1 SAR Time Series and Deep Learning,” *Sci Rep*, vol. 10, no. 1, Art. no. 1, Jan. 2020, doi: 10.1038/s41598-019-56967-x.

[172] M. Denis, “SELECTED ISSUES REGARDING SMALL COMPACT CITY – ADVANTAGES AND DISADVANTAGES,” *piF*, vol. 2018, no. 34, pp. 151–162, Apr. 2018, doi: 10.21005/pif.2018.34.C-03.

[173] J. Pluto-Kossakowska and M. Cuprjak, “INDICATORS METHOD OF AESTHETICS ANALYSIS USING SPATIAL DATASETS,” *piF*, vol. 2023, no. 55, pp. 179–204, Sep. 2023, doi: 10.21005/pif.2023.55.C-03.

[174] A. Lefebvre, C. Sannier, and T. Corpetti, “Monitoring Urban Areas with Sentinel-2A Data: Application to the Update of the Copernicus High Resolution Layer Imperviousness Degree,” *Remote Sensing*, vol. 8, no. 7, Art. no. 7, Jul. 2016, doi: 10.3390/rs8070606.

[175] M. Buchhorn, M. Lesiv, N.-E. Tsendbazar, M. Herold, L. Bertels, and B. Smets, “Copernicus Global Land Cover Layers—Collection 2,” *Remote Sensing*, vol. 12, no. 6, Art. no. 6, Jan. 2020, doi: 10.3390/rs12061044.

[176] European Commission. Joint Research Centre., *Mapping population density in functional urban areas: a method to downscale population statistics to urban atlas polygons.* LU: Publications Office, 2016. Accessed: Jul. 12, 2024. [Online]. Available: https://data.europa.eu/doi/10.2791/06831

[177] M. Ohki and M. Shimada, “Large-Area Land Use and Land Cover Classification With Quad, Compact, and Dual Polarization SAR Data by PALSAR-2,” *IEEE Trans. Geosci. Remote Sensing*, vol. 56, no. 9, pp. 5550–5557, Sep. 2018, doi: 10.1109/TGRS.2018.2819694.

[178] T. Esch, M. Thiel, A. Schenk, A. Roth, A. Muller, and S. Dech, “Delineation of Urban Footprints From TerraSAR-X Data by Analyzing Speckle Characteristics and Intensity Information,” *IEEE Trans. Geosci. Remote Sensing*, vol. 48, no. 2, pp. 905–916, Feb. 2010, doi: 10.1109/TGRS.2009.2037144.

[179] M. Thiel, T. Esch, and A. Schenk, “OBJECT-ORIENTED DETECTION OF URBAN AREAS FROM TERRASAR-X DATA,” 2008.

[180] F. Dell’Acqua and P. Gamba, “Texture-based characterization of urban environments on satellite SAR images,” *IEEE Trans. Geosci. Remote Sensing*, vol. 41, no. 1, pp. 153–159, Jan. 2003, doi: 10.1109/TGRS.2002.807754.

[181] W. Zhai, H. Shen, C. Huang, and W. Pei, “Fusion of polarimetric and texture information for urban building extraction from fully polarimetric SAR imagery,” *Remote Sensing Letters*, vol. 7, no. 1, pp. 31–40, Jan. 2016, doi: 10.1080/2150704X.2015.1101179.

[182] M. Gong, Y. Li, L. Jiao, M. Jia, and L. Su, “SAR change detection based on intensity and texture changes,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 93, pp. 123–135, 2014, doi: https://doi.org/10.1016/j.isprsjprs.2014.04.010.

[183] K. Ji and Y. Wu, “Scattering Mechanism Extraction by a Modified Cloude-Pottier Decomposition for Dual Polarization SAR,” *Remote Sensing*, vol. 7, no. 6, pp. 7447–7470, Jun. 2015, doi: 10.3390/rs70607447.

[184] N. Bhogapurapu *et al.*, “Dual-polarimetric descriptors from Sentinel-1 GRD SAR data for crop growth assessment,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 178, pp. 20–35, Aug. 2021, doi: 10.1016/j.isprsjprs.2021.05.013.

[185] C. G. Candido and J. A. Principe, “DUAL-POLARIMETRIC DECOMPOSITION OF SENTINEL-1 SAR IMAGE AND MACHINE LEARNING MODEL FOR OIL SPILL DETECTION: CASE OF MINDORO OIL SPILL,” *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLVIII-4-W8-2023, pp. 101–106, Apr. 2024, doi: 10.5194/isprs-archives-XLVIII-4-W8-2023-101-2024.

[186] J. B. D. Jesus, T. M. Kuplich, Í. D. D. C. Barreto, and D. C. Gama, “Dual polarimetric decomposition in Sentinel-1 images to estimate aboveground biomass of arboreal caatinga,” *Remote Sensing Applications: Society and Environment*, vol. 29, p. 100897, Jan. 2023, doi: 10.1016/j.rsase.2022.100897.

[187] Pluto-Kossakowska, Joanna and Wędzikowska, Paulina, “Texture analysis of optical and radar imagery to assess grey infrastructure sealing and density,” presented at the National Conference on Photointerpretation and Remote Sensing, Warsaw, Poland, 2022. [Online]. Available: https://www.xxvokfit.pw.edu.pl/

[188] G. Urban Atlas, “Urban Atlas — Copernicus Land Monitoring Service.” Accessed: Jun. 28, 2024. [Online]. Available: https://land.copernicus.eu/en/products/urban-atlas

[189] Eckardt , Robert, Urbazaev, M., Salepci, N., Schmullius, Ch., Woodhouse, i., and Stewart, Ch., “MOOC on SAR: Echoes in Space,” eo science for society. Accessed: Jun. 28, 2024. [Online]. Available: https://eo4society.esa.int/resources/echoes-in-space/

[190] M. Hall-Beyer, “GLCM Texture: A Tutorial v. 3.0 March 2017,” Mar. 2017, Accessed: Jun. 28, 2024. [Online]. Available: http://hdl.handle.net/1880/51900

[191] R. M. Haralick, K. Shanmugam, and I. Dinstein, “Textural Features for Image Classification,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973, doi: 10.1109/TSMC.1973.4309314.

[192] A. Braun, “SAR-based landcover classification with Sentinel-1 GRD products,” ESA, Oct. 2020. [Online]. Available: https://step.esa.int/docs/tutorials/S1TBX%20Landcover%20classification%20with%20Sentinel-1%20GRD.pdf

[193] S. Sarkar, T. Halder, V. Poddar, R. K. Gayen, A. M. Ray, and D. Chakravarty, “A Novel Approach for Urban Unsupervised Segmentation Classification in SAR Polarimetry,” in *2021 2nd International Conference on Range Technology (ICORT)*, Chandipur, Balasore, India: IEEE, Aug. 2021, pp. 1–5. doi: 10.1109/ICORT52730.2021.9581380.

[194] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” May 18, 2015, *arXiv*: arXiv:1505.04597. doi: 10.48550/arXiv.1505.04597.

[195] H. Zhang *et al.*, “ResNeSt: Split-Attention Networks,” Dec. 30, 2020, *arXiv*: arXiv:2004.08955. Accessed: Jun. 28, 2024. [Online]. Available: http://arxiv.org/abs/2004.08955

[196] Y. Bai, B. Adriano, E. Mas, and S. Koshimura, “Building Damage Assessment in the 2015 Gorkha, Nepal, Earthquake Using Only Post-Event Dual Polarization Synthetic Aperture Radar Imagery,” *Earthquake Spectra*, vol. 33, no. 1_suppl, pp. 185–195, Dec. 2017, doi: 10.1193/121516eqs232m.

[197] Esri, “World Imagery,” 2023.

[198] Z. Zhu, C. E. Woodcock, J. Rogan, and J. Kellndorfer, “Assessment of spectral, polarimetric, temporal, and spatial dimensions for urban and peri-urban land cover classification using Landsat and SAR data,” *Remote Sensing of Environment*, vol. 117, pp. 72–82, Feb. 2012, doi: 10.1016/j.rse.2011.07.020.

[199] C. Corbane, J.-F. Faure, N. Baghdadi, N. Villeneuve, and M. Petit, “Rapid Urban Mapping Using SAR/Optical Imagery Synergy,” *Sensors*, vol. 8, no. 11, pp. 7125–7143, Nov. 2008, doi: 10.3390/s8117125.

[200] C. Corbane and F. Sabo, “European Settlement Map from Copernicus Very High Resolution data for reference year 2015, Public Release 2019.” European Commission, Joint Research Centre (JRC), Oct. 01, 2019. doi: 10.2905/8BD2B792-CC33-4C11-AFD1-B8DD60B44F3B.

[201] M. Datcu, Z. Huang, A. Anghel, J. Zhao, and R. Cacoveanu, “Explainable, Physics Aware, Trustworthy AI Paradigm Shift for Synthetic Aperture Radar,” Jan. 09, 2023, *arXiv*: arXiv:2301.03589. Accessed: Mar. 21, 2023. [Online]. Available: http://arxiv.org/abs/2301.03589