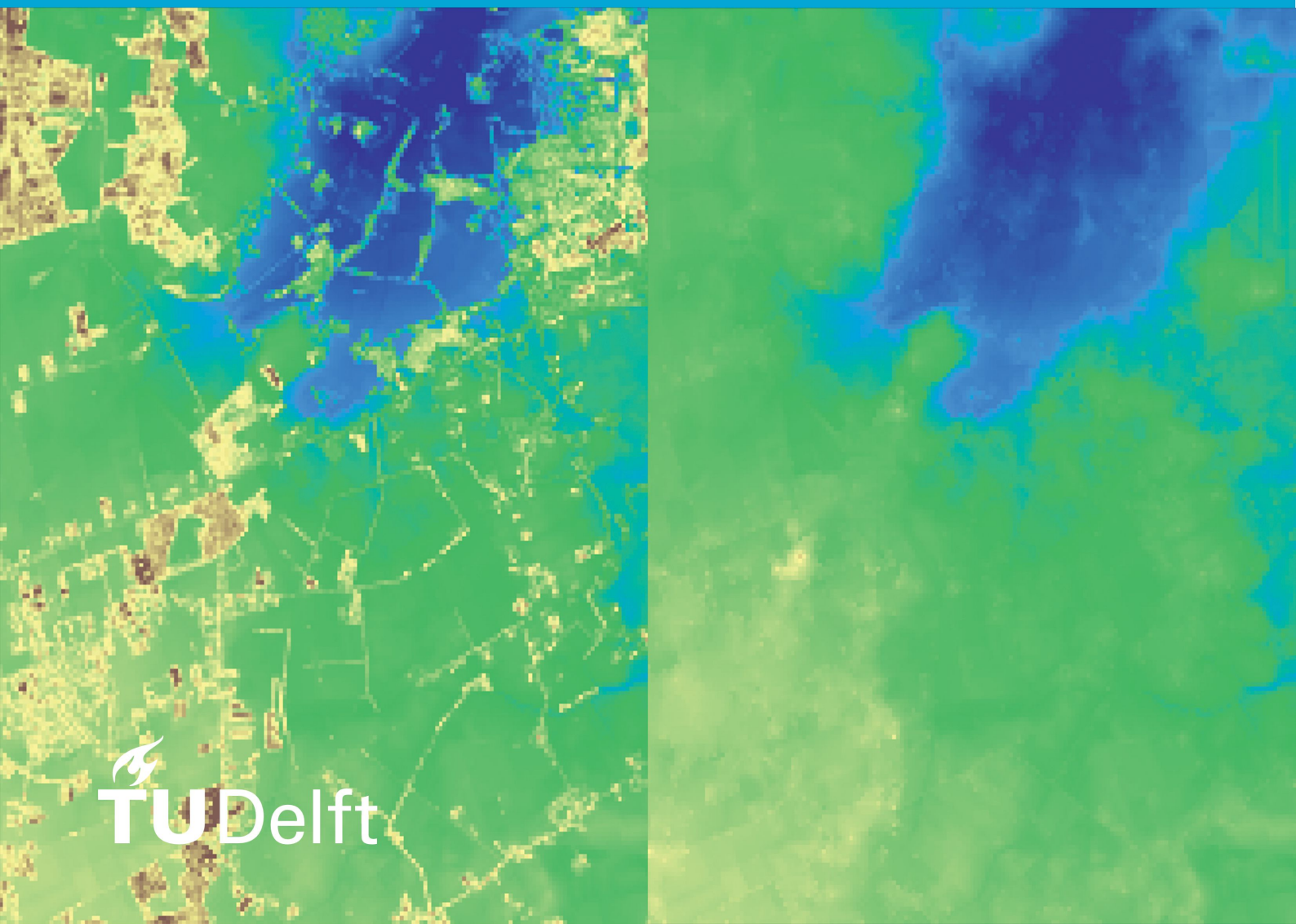
MSc thesis in Geomatics

Correcting Global Elevation Models for Canopy and Infrastructure Using a Residual U-Net

Haoyang Dong

2023



TU Delft

MSc thesis in Geomatics

Correcting Global Elevation Models for Canopy and Infrastructure Using a Residual U-Net

Haoyang Dong

June 2023

A thesis submitted to the Delft University of Technology in partial fulfillment of the requirements for the degree of Master of Science in Geomatics

The work in this thesis was carried out in the:

3D geoinformation group
Delft University of Technology

Supervisors:	ir. Maarten Pronk
	Dr. Hugo Ledoux
Co-reader:	Dr. Liangliang Nan

Abstract

Digital Surface Models (DSMs) are commonly employed to investigate topographical characteristics and processes; however, the presence of canopy and infrastructure in urban and forested areas can lead to height biases and inaccuracies. In this study, I aim to correct such biases by applying a deep learning approach known as Residual U-Net to remove the selected pixels and generate Digital Terrain Models (DTMs) that accurately represent the Earth's surface without canopy and infrastructure influence.

The Residual U-Net model was trained and tested on a dataset of DSM and DTM pairs, which were acquired from resampled AHN4. The model was evaluated on its ability to predict DTMs from DSMs, and its performance was compared with other existing methods. Additionally, the model was tested on different resolutions and the Copernicus DEM to assess its adaptability and generalization capabilities.

The results indicate that the Residual U-Net model outperforms conventional techniques, effectively reducing the influence of canopy and infrastructure, and resulting in DTMs with enhanced precision. The study also explores the errors in detail and identifies the model's error causes, highlighting its limitations and areas for potential improvement.

This study concludes by demonstrating the efficacy of applying deep learning techniques, such as Residual U-Net, to correct global elevation models for canopy and infrastructure. The results indicate that the model is a promising tool for topographical investigation in both urban and woodland situations, offering a versatile solution for generating accurate DTMs from DSMs.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisors. Their unwavering support and guidance throughout this journey have been invaluable.

This study experienced three withdrawals, effectively doubling its intended duration. This unusual trajectory can be attributed to a number of challenges I faced, not least of which was political depression. Once again, I must extend my sincere appreciation to my supervisors for their patience and enduring support during these testing times.

In respect to the incidents that led to the aforementioned withdrawals, I choose to withhold a detailed account due to the public nature of this thesis. Given the potential risks and consequences, including legal repercussions, it seems prudent not to elaborate further. However, the strains of these ordeals may be inferred from my expressed concerns.

I owe an enormous debt of gratitude to my partner, Yowai. Her presence has been a source of comfort and solace, fostering within me a sense of inner peace. Without her support, progress would have been a challenge too great to overcome.

My heartfelt thanks extend to my dear friends. Over the course of this study, relationships evolved; friendships were let go, and new ones were formed. In particular, I am grateful to the Mastodon community, where I met most of these friends who have significantly enriched my life.

I am also appreciative of the Babylon cinema in Berlin. Although regular visits were not feasible, the mere existence of this space offered profound emotional relief.

Lastly, I wish to acknowledge my own perseverance and determination. I truly believe that I have grown and continued to progress towards my ideals throughout this journey. I am thankful to myself for persisting in the face of adversity and for never ceasing to aspire for more.

Contents

1. INTRODUCTION **1**
- 1.1. Background & Motivation . . . 1
- 1.2. Study Area & Dataset . . . 2
- 1.3. Research Objective . . . 2
  - 1.3.1. Main Goal . . . 2
  - 1.3.2. Sub-questions . . . 3
- 1.4. Thesis Outline . . . 3

2. THEORETICAL BACKGROUND AND RELATED WORK **5**
- 2.1. DEM, DSM, and DTM . . . 5
  - 2.1.1. Actueel Hoogtebestand Nederland 4 (ANH4) . . . 6
- 2.2. DEM Analysis . . . 7
  - 2.2.1. Slope . . . 7
  - 2.2.2. Roughness . . . 7
  - 2.2.3. Aspect . . . 7
  - 2.2.4. Topographic Roughness Index (TRI) . . . 8
  - 2.2.5. Topographic Position Index (TPI) . . . 8
- 2.3. DTM Extraction & Machine Learning . . . 8
  - 2.3.1. Traditional Regression Analysis . . . 8
  - 2.3.2. Machine learning . . . 9
  - 2.3.3. Regression, Classification, and Segmentation . . . 9
  - 2.3.4. Emerging Approaches for Elevation Error Correction . . . 11
  - 2.3.5. U-Net . . . 12
  - 2.3.6. Residual Networks (ResNets) . . . 13
  - 2.3.7. Residual U-Net . . . 15

3. METHODOLOGY **17**
- 3.1. Residual U-Net . . . 17
  - 3.1.1. Architecture . . . 17
- 3.2. Data Processing . . . 19
  - 3.2.1. Data Acquisition . . . 19
  - 3.2.2. DEM Preprocessing . . . 20
- 3.3. Generating DTMs & Evaluation . . . 24
  - 3.3.1. Stitching . . . 25
  - 3.3.2. Root-mean-square Error (RMSE) . . . 26
- 3.4. Differences Between My Method and FABDEM . . . 27

4. IMPLEMENTATION & EXPERIMENTS **29**
- 4.1. Code & Data Specifications . . . 29
- 4.2. Residual U-Net . . . 29
  - 4.2.1. Implementation Details . . . 29
  - 4.2.2. Combinations of Inputs . . . 30

4.2.3. Normalization . . . 31
4.2.4. Threshold Set for Generating Ground Truth . . . 31
4.3. Data Processing . . . 32
4.3.1. Resampling . . . 32
4.3.2. Inverse Distance Weighting (IDW) Interpolation . . . 32
4.3.3. DEM Anaylize . . . 32
4.3.4. Data Augmentation . . . 33

5. RESULTS **35**
5.1. Segmentation Accuracy . . . 35
5.2. Comparison with FABDEM . . . 36
5.3. Evaluating Model Performance in Miami and Elevated Miami . . . 41
5.4. Visual Comparisons of Other Areas . . . 43
5.4.1. Dense Urban Area (Shibuya and Shinjuku in Tokyo) . . . 43
5.4.2. Vast Forested Region (Amazon Basin) . . . 44
5.4.3. Mountainous Area (Alps in the western region of Austria) . . . 45
5.5. Other Resolutions . . . 47

6. DISCUSSION & CONCLUSION **49**
6.1. Single Input Analysis with Fixed Window Size . . . 49
6.2. Limitations & Method Flaws . . . 50
6.2.1. Artifacts . . . 50
6.3. The Improvement of Slicing with Overlaps . . . 52
6.4. Input Sensitivity . . . 53
6.5. Future Work . . . 53
6.6. Conclusion . . . 53

A. Reproducibility self-assessment **55**
A.1. Marks for each of the criteria . . . 55
A.2. Self-reflection . . . 55

B. Supporting Datasets and Links **57**
B.1. Github Repository . . . 57
B.2. Datasets . . . 57

List of Figures

2.1. An illustration depicting the differences between DTM and DSM in the context of trees and buildings. [Polidori and El Hage, 2020] 6
2.2. Schematic representation of FABDEM [Hawker et al., 2022] 12
2.3. The architecture of a U-Net. [Ronneberger et al., 2015] 13
2.4. A building block of the U-Net. [He et al., 2015] 14
2.5. A identity block of the U-Net. [Shehab et al., 2021] 14
2.6. A convolutional block of the U-Net. [Shehab et al., 2021] 14

3.1. Residual U-Net architecture (rotated) 18
3.2. An example of a patch of DSM with a notable abundance of no-value pixels: (a) Satellite Image, (b) AHN4 DSM (resampled to 30 m resolution), (c) The DSM after gap filling. Notice that there are artifacts in the gap-filled DSM. . . 21
3.3. Distribution of the patches. The regions in orange indicate the areas used for training the model, whereas the regions in green denote the areas used for testing the model. 21
3.4. Generation of inputs and labels for the model from DSMs and DTMs. 23
3.5. An illustration shows the relationship between the pixels and the labels. 23
3.6. Examples of a 23x23 array sliced into 5x5 segments, with overlapping areas highlighted in red: (a) Slicing without inner overlaps; (b) Slicing with a 1-pixel width overlap. 25
3.7. The process of generating a DTM from a DSM using the trained model. 26

4.1. An example of the combination: (a) Filled DSM at 30 m resolution, (b) Slope map, (c) Roughness map. 31

5.1. The worst-performing patch: (a) Satellite imagery, (b) DSM (resampled), (c) DTM (resampled), (d) Ground Truth computed from the DSM and the DTM, (e) Prediction from the model, and (f) positive-negative map (subtracting the ground truth from the prediction maps) 36
5.2. The best-performing patch: (a) Satellite imagery, (b) DSM (resampled), (c) DTM (resampled), (d) Ground Truth computed from the DSM and the DTM, (e) Prediction from the model, and (f) positive-negative map (subtracting the ground truth from the prediction maps) 37
5.3. Comparison of DTMs: (a) Satellite Image, (b) AHN4 DSM, (c) COP30, (d) AHN4 DTM (reference), (e) Generated DTM by the model, and (f) FABDEM. . 38
5.4. Plots of the best performance area: (a) Satellite image, (b) COP30, (c) Prediction by the method, (d) generated DTM, (e) FABDEM, (f) AHN4 DTM (reference). 39
5.5. Plots of the worst performance area: (a) Satellite image, (b) COP30, (c) Prediction by the model, (d) generated DTM, (e) FABDEM, (f) AHN4 DTM (reference). The blurred area in the satellite image is a military Airport in Rotterdam. 40

5.6. Plots of the Miami area used in the study: (a) Satellite image, (b) COP 30, (c) FABDEM, (d) Reference DTM. 41
5.7. Plots of the Miami area used in the study: (a) COP30, (b) Prediction map by the model, (c) Generated DTM, (d) FABDEM. 42
5.8. Plots of the Miami area used in the study: (a) elevated COP30, (b) Prediction map by the model, (c) Generated DTM, (d) elevated FABDEM. 43
5.9. Plots of the dense urban area (Shibuya and Shinjuku in Tokyo): (a) the satellite imagery, (b) COP30, (c)prediction, (d) FABDEM, (e) the DTM generated from COP30 by the model. 44
5.10. Plots of the vast forested region (Amazon Basin): (a) the satellite imagery, (b) COP30, (c) prediction, (d) FABDEM, (e) the DTM generated from COP30 by the model. 45
5.11. Plots of the mountainous area (Alps in the western region of Austria): (a) the satellite imagery, (b) COP30, (c) prediction, (d) FABDEM, (e) the DTM generated from COP30 by the model. 46
5.12. DSMs and the DTMs at different resolutions. 48

6.1. An example of The DSM layer of an input of a DSM from Amazon Basin. . . . 49
6.2. The other 2 layers of the input. 50
6.3. Plots of the Amazon Basin : (a) Prediction, (b) Generated DTM. 51
6.4. Plots of the Alps area : (a) Prediction, (b) Generated DTM. 51
6.5. Comparison of prediction maps for the dense urban area when sliced with or without overlaps, highlighting the artifacts or errors caused by no-overlap slicing in (a) with red boxes. 52

A.1. Reproducibility criteria to be assessed. 55

List of Tables

3.1. Comparison with FABDEM 27

4.1. Number and shape of inputs in different stages 29

5.1. Accuracy analysis 35
5.2. (RMSE) of Different DTMs Compared to AHN4 DTMs.Note: ″*″ denotes the RMSE values for the best-performance and worst-performance areas. 38
5.3. RMSE for Miami DEMs 42
5.4. Mean accuracy and RMSE for different input resolutions 47

6.1. Comparison of mean accuracy and RMSE for slicing method with and without overlaps 52

List of Algorithms

3.1. Contraction 19
3.2. Expansion 20
3.3. Slicing 24
3.4. Stitching 25

1. INTRODUCTION

A digital elevation model (DEM) is a crucial component of geographic information systems (GIS) and an invaluable resource for a vast array of scientific, engineering, and planning applications[Lakshmi and Yarrakula, 2018]. It captures the three-dimensional surface morphology of the Earth's surface, including terrain heights and depressions, slopes, and aspects, in a digital format. DEMs are a great resource for geological applications such as soil erosion, reservoir planning, and flood prediction due to the wealth of topographic and geomorphological information they provide.

Many local and global DEM datasets are now available for public use since the availability and accessibility of DEM data have vastly grown in recent years [Polidori and El Hage, 2020]. However, The vast majority of available global-scale DEMs are digital surface models (DSMs) rather than digital terrain models (DTMs).

1.1. Background & Motivation

Numerous geoscience applications necessitate an understanding of the true terrain surface, not the surface disguised by plant or man-made buildings, and so require accurate elevation data. The existence of these traits in DSMs results in inaccurate elevation data and impedes their usage in a variety of applications. Direct use of DSMs in certain applications will result in mistakes; for instance, the threat of floods may be underestimated when utilising DSMs in urban or forested settings [Kulp and Strauss, 2016]. In order for the DEM to be technically regarded as a DTM, it is necessary to employ post-processing techniques to eliminate height bias from trees and structures. This is why adjusting global elevation models for the presence of canopy and infrastructure is a significant and ongoing research topic.

Despite the increased interest in the conversion of global digital surface models (DSMs) to digital terrain models (DTMs) over the past few years, the methodologies proposed in previous studies have drawbacks that limit their effectiveness. Hawker et al. [2022]; Kulp and Strauss [2018] have attempted to address this issue; however, their algorithms require additional inputs, including population density and vegetation density. This might lead to constraints on the number of model inputs, resulting in inferior results and the appearance of artefacts. In spite of this, recent advances in deep-learning algorithms for computer vision have indicated the promise in the extraction of DTMs, as evidenced by works by Meadows and Wilson [2021]; Kazimi et al. [2020]; Gevaert et al. [2018]. However, these methods often require many bands of remote sensing imagery or point clouds as inputs, which may not always be accessible or attainable.

Deep learning algorithms have demonstrated excellent performance in several computer vision applications over the past decade, including object categorization, detection, and semantic segmentation. Despite the vast disparities between remote sensing data and conven-

tional images, academics have made a concentrated effort to use these improved technologies in geoscience studies. Using both remote sensing pictures and DEM data, Zhou et al. [2022] creates a way to identify locations damaged by landslides automatically. In comparison, Dang et al. [2020] suggests a deep learning strategy for categorising nine categories of coastal wetlands.

These advancements have created new opportunities for investigating the translation of Digital Surface Models (DSMs) into Digital Terrain Models (DTMs). This thesis capitalises on this trend and focuses on the application of deep learning techniques to accomplish this change. I use the resampled AHN4 (Actueel Hoogtebestand Nederland 4)[1] DSMs as inputs, while the outputs are DTMs generated through the use of the Residual U-Net model. The quality of the method is evaluated by comparing the accuracy to other related works.

1.2. Study Area & Dataset

This research focuses on the Netherlands, a country characterized by its low-lying topography and intricate hydrological systems. Accurate elevation data is particularly important in this region due to its vulnerability to flooding and sea-level rise, as well as its extensive infrastructure and densely populated urban areas. The study area is also known for its diverse landscape, which includes urban, rural, and natural environments, further emphasizing the need for accurate and reliable DTMs.

The dataset used in this research is the Actueel Hoogtebestand Nederland 4, a high-resolution digital elevation model produced by the Dutch government. The AHN4 dataset covers the entire country of the Netherlands and is generated using LiDAR technology, providing a detailed representation of the Earth's surface, including both natural and man-made features. The AHN4 dataset is widely used in various applications such as flood risk assessment, infrastructure planning, and environmental management.

In this study, I utilize the resampled AHN4 DSMs as inputs for the deep learning model, aiming to convert them into accurate DTMs. By focusing on a specific region and dataset, I can better assess the effectiveness of my proposed methodology and its potential for broader application to other regions and datasets.

1.3. Research Objective

1.3.1. Main Goal

Develop and evaluate a Residual U-Net model for reliable DTM extraction from DSMs by removing pixels associated with the canopies and infrastructures.

[1] https://www.ahn.nl/ahn-4

1.3.2. Sub-questions

1. How does the performance of the Residual U-Net compare to other deep learning methods for DTM extraction in terms of accuracy and RMSE after interpolation?
2. How do the model's results compare to global DEMs, and what are the quantitative and visual differences when applied to other areas and datasets?
3. How does the resolution influence the performance of the model?
4. What are the advantages and limitations of the Residual U-Net approach for DTM extraction, and how can these challenges be addressed in future work?

1.4. Thesis Outline

This thesis is organized as follows:

- **Chapter 1: Introduction** provides the background, motivation, and objectives of this study. It also introduces the study area and dataset.
- **Chapter 2: Theoretical Background and Related Work** elaborates on the technical aspects, including the concepts of DEM, DSM, DTM, and AHN4, and various DEM analysis approaches. This chapter also details the machine learning methodologies employed and discusses emerging approaches for elevation error correction, including FABDEM.
- **Chapter 3: Methodology** delineates the implementation of Residual U-Net and the data processing steps undertaken. It further details the generation and evaluation of DTMs and contrasts my approach with FABDEM.
- **Chapter 4: Implementation and Experiments** describes the coding and data specifications, the details of the Residual U-Net implementation, and various data processing steps. This chapter also elaborates on the different combinations of inputs, IDW interpolation, normalization techniques, and the thresholds set for generating Ground Truth.
- **Chapter 5: Results** presents the accuracy of segmentation, comparison with FABDEM, quantitative evaluation and visual comparisons of other areas, and analysis of different resolutions.
- **Chapter 6: Discussion and Conclusion** analyzes single input data with a fixed window size, discusses the effects of overlaps, outlines the limitations and flaws of the method, and delves into input sensitivity. It also suggests areas of future work and summarizes the conclusions drawn from the research.

2. THEORETICAL BACKGROUND AND RELATED WORK

In this chapter, I present an overview of the existing literature in the fields of global elevation models, canopy and infrastructure correction, and deep learning techniques such as Residual Networks and U-Net. This review provides the necessary context for understanding the motivation and contributions of the research on correcting global elevation models. By examining the existing literature, we can gain insights into the most effective approaches and methodologies, enabling us to design a more robust and innovative research framework for addressing the challenges in correcting global elevation models.

2.1. DEM, DSM, and DTM

The Digital Elevation Model (DEM) is a prevalent term in Geographic Information Systems (GIS) employed to represent the topographic relief of the Earth's surface. Various data models have been proposed for storing elevation data, including regularly spaced tessellation (grids), Triangular Irregular Network (TIN), and others. Among these, the regularly spaced tessellation model has emerged as the most widely used method for processing DEM data [Mesa-Mingorance and Ariza-López, 2020]. The definition of a Digital Elevation Model (DEM), as "a regular gridded matrix representation of the continuous variation of relief over space" suggested by Burrough and McDonnell [1986], provides a comprehensive interpretation of the concept and will be adopted as the working definition in the present study when referring to DEMs.

Digital Elevation Models (DEMs) are an essential tool for research in geoscience and have broad applications across many fields, including Hydrology, Geomorphology, Civil Engineering, Biology, and studies of geologic, geomorphic, atmospheric, hydrologic, and ecological processes [Mesa-Mingorance and Ariza-López, 2020; Okolie and Smit, 2022]. As a result of this increasing demand, there is a growing need for high-quality wide-area DEMs and global DEMs (GDEM).

Over the course of the past century, the techniques used for mapping elevation have undergone significant evolution, with the mapping industry transitioning from traditional ground surveying methods to advanced passive and active sensing methodologies [Polidori and El Hage, 2020; Okolie and Smit, 2022]. In recent years, a range of modern techniques for acquiring Digital Elevation Models (DEMs) have emerged, including airborne LiDAR [Liu, 2008], spaceborne IfSAR [van Zyl, 2001], aerial photography [James and Robson, 2012], etc. The development of data capture technologies, combined with advanced processing methods such as automatic image matching and aerotriangulation [Polidori and El Hage, 2020], have enabled the creation of global Digital Elevation Models (GDEM) with increased efficiency and accuracy [Mesa-Mingorance and Ariza-López, 2020].

Digital Elevation Models (DEMs) are often divided into two models based on the differing definitions of nominal surfaces, which are the physical surfaces being modeled. The two frequently considered models are the Digital Terrain Model (DTM) and the Digital Surface Model (DSM). In the case of a DTM, the nominal surface is the terrain itself, while in the case of a DSM, the nominal surface is the surface above trees, buildings, and other natural or artificial objects [Polidori and El Hage, 2020]. Figure 2.1 illustrates a conceptual representation of a DSM and DTM. Many of the current DEM manufacturing techniques, including photogrammetry and short-wavelength radar technologies, are designed to generate DSMs [Hawker et al., 2022].

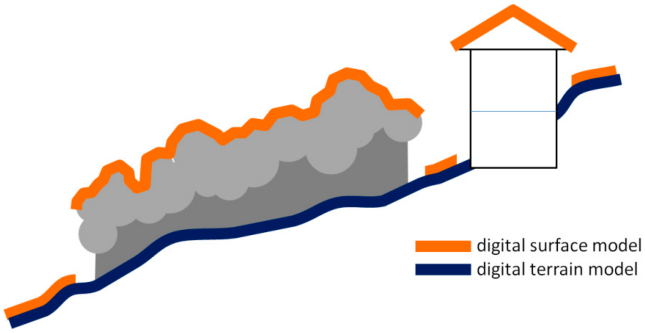


Figure 2.1.: An illustration depicting the differences between DTM and DSM in the context of trees and buildings. [Polidori and El Hage, 2020]

2.1.1. Actueel Hoogtebestand Nederland 4 (ANH4)

ANH4 is an updated digital elevation model of the Netherlands created using laser altimetry from aircraft. The data collection process is divided into seven steps: data collection, processing, filtering, quality control, end product creation, merging with existing data, and distribution [AHN, 2020a]. AHN4 is part of a multi-year program, a collaboration between Dutch water authorities, provinces, and Rijkswaterstaat to create a digital elevation model of the Netherlands [AHN, 2023].

The AHN4 project began in 2020 and is now completed in 2023. It provides more accurate and up-to-date elevation data for the country. The AHN4 has a systematic error of 5 cm and a stochastic error of 5 cm, with a minimum of 99.7% of points having a height accuracy of 20 cm. The point density for AHN4 is around 10-14 points per square meter, with even higher density (20-24 points per square meter) in the area around Schiphol airport [AHN, 2020b].

The data collected is classified into different categories: ground, buildings (limited), infrastructure (limited), water, and others. The classification process is conducted through a semi-manual approach. During the subsequent checking phase, point distribution and filtering techniques are employed [AHN, 2020a]. The final data is delivered in various formats, including classified laser data, 0.5m ground raster (DTM), 0.5m raw raster (DSM - no water class), 5m ground raster (DTM), and 5m raw raster (DSM – no water class) [AHN, 2023, 2020b]. These raster files in GeoTiff format can be easily used in various GIS software packages. The study utilizes raster files as the initial dataset.

2.2. DEM Analysis

This section introduces several parameters that can be directly measured from Digital Elevation Models (DEMs). These parameters provide valuable insights into the characteristics of the terrain and play a crucial role in various geospatial analyses. By leveraging the information encoded in DEMs, researchers can extract meaningful metrics that describe key aspects of the landscape, such as slope, aspect, roughness, topographic roughness index (TRI), and topographic position index (TPI). These parameters offer quantitative measures of the terrain's characteristics, enabling a better understanding of its morphology, surface variations, and topographic features. By utilizing these directly measurable parameters, researchers can gain valuable insights into the landscape's properties and support informed decision-making in a wide range of disciplines, including geology, hydrology, ecology, and land management.

2.2.1. Slope

Slope, a crucial parameter derived from digital elevation models (DEMs), refers to the change in elevation along the steepest incline within a given analysis window, regardless of its orientation. The magnitude of slope values is influenced by the size of the analysis window, where smaller window sizes accentuate steeper slopes over shorter distances [Wilson et al., 2007; Saleem et al., 2019]. Additionally, slope can be classified into gradient and aspect components, wherein the gradient represents the maximum rate of change in altitude, while the aspect denotes the compass direction associated with this maximum rate of change [Saleem et al., 2019].

2.2.2. Roughness

The roughness value is calculated as the difference between the maximum and minimum bathymetry values within a user-defined rectangular neighbourhood centred around a specific pixel. This measure quantifies the variation in terrain and is particularly relevant for studies focusing on habitat mapping and characterization [Wilson et al., 2007; Smith, 2014]. By analyzing the roughness values, researchers can identify distinct patterns and features in the landscape [Wilson et al., 2007].

2.2.3. Aspect

Aspect represents the direction of the maximum slope at each point[Saleem et al., 2019]. It is determined by calculating the tangent of the angle between a horizontal plane and the surface normal vector at a specific location on the DEM. This can be represented with the equation $aspect = arctan(e/d)$, where e and d represent the partial derivatives of the DTM surface in the x and y directions Wilson et al. [2007]. The aspect value provides crucial information about the orientation of terrain features, such as ridges and valleys, and plays a vital role in various geospatial analyses, including the assessment of landslide susceptibility and risk [Saleem et al., 2019; Wilson et al., 2007].

2.2.4. Topographic Roughness Index (TRI)

TRI is a morphometric measure to assess the ruggedness of the terrain [Saleem et al., 2019]. It provides a quantification of the heterogeneous nature of a land surface, distinguishing between smooth and rugged areas. TRI measures the differences between the central pixel and its neighbouring pixels, taking the absolute values of these differences and averaging the results. It highlights small-scale variations in the terrain and tends to capture the same features as the slope calculations [Wilson et al., 2007].

2.2.5. Topographic Position Index (TPI)

TPI serves to identify the position of a point in relation to its surrounding terrain by subtracting the average elevation of the neighbouring pixels from the elevation of the central pixel [Saleem et al., 2019; Wilson et al., 2007]. It quantifies the deviation of a point's elevation from the average elevation of its surroundings within a specified search radius. Positive TPI values signify that the central pixel is situated at a higher elevation than its neighbouring area, while negative values denote a lower position [Wilson et al., 2007].

2.3. DTM Extraction & Machine Learning

2.3.1. Traditional Regression Analysis

Traditional approaches to the correction of elevation errors predominantly rely on regression analysis, typically employing a limited set of variables. These methodologies often draw upon vegetation cover indices to mitigate the bias introduced by tree canopies. However, their applicability to the removal of biases caused by built structures, such as buildings, is significantly restricted [Baugh et al., 2013; Su et al., 2015; Kulp and Strauss, 2018].

Improvements in elevation error correction could be achieved by incorporating a broader set of variables that exhibit correlations with these errors. Yet, as the dimensionality of the problem increases with the introduction of additional variables, traditional parametric regression techniques encounter limitations. This is primarily attributed to the "curse of dimensionality" [Köppen, 2000], a phenomenon that refers to the challenges encountered when handling high-dimensional data. These challenges include the computational intractability of conducting systematic searches through high-dimensional spaces, the difficulties in accurately approximating high-dimensional functions, and the complexities associated with integrating such functions [Donoho and others, 2000].

Machine learning, as an empirically effective approach for regression and classification tasks in nonlinear systems [Lary et al., 2016], may provide a solution to circumvent the curse of dimensionality.

2.3.2. Machine learning

Machine learning (ML) is a multidisciplinary field that encompasses concepts from various domains such as computer science, statistics, cognitive science, engineering, etc.[Ghahramani, 2004; Soofi and Awan, 2017] It focuses on enabling computers to simulate human learning processes, acquiring new knowledge and skills, identifying existing information, and continuously enhancing their performance and achievements [Wang et al., 2009]. It involves developing algorithms and models that enable computers to learn from data, detect patterns, make predictions, and improve their performance through experience [Wang et al., 2009; Ray, 2019].

Machine learning (ML) serves as an empirical approach whose fundamental principle involves constructing a comprehensive training dataset comprising diverse examples that span a wide range of system parameter values [Lary et al., 2016]. ML finds effectiveness in the empirical regression and classification of nonlinear systems, regardless of whether they involve supervised or unsupervised learning [Lary et al., 2016; Liu et al., 2019; Prasad et al., 2006].

Over the years, machine learning (ML) has witnessed the development of various algorithms and models, which have been extensively applied in diverse domains [Lary et al., 2016; Liu et al., 2019; Prasad et al., 2006]. However, in this thesis, I will focus solely on the models and algorithms that are relevant to my research. I will introduce and discuss these specific models and algorithms, highlighting their applications in addressing the research objectives and solving pertinent problems. Furthermore, I will analyze the strengths and limitations of these approaches, providing readers with a comprehensive understanding of their potential and effectiveness within the context of my study.

By leveraging its capacity to model complex, nonlinear relationships between variables, deep learning can enhance the robustness and accuracy of elevation error correction methodologies [Poggio et al., 2017]. Consequently, it offers a promising avenue for improving the precision and reliability of digital elevation models in geospatial analysis and applications.

Before delving into the details of the models employed in DTM extraction, it is essential to comprehend the distinctions between regression, classification, and segmentation, as well as gain familiarity with some commonly used models. This foundational knowledge is crucial for comprehending the intricate methods and models utilized in the process of DTM extraction.

2.3.3. Regression, Classification, and Segmentation

ML can be broadly classified into two categories: supervised and unsupervised learning. Unsupervised learning involves drawing conclusions from unlabeled datasets where the desired output is not provided. On the other hand, supervised learning techniques aim to establish the relationship between input attributes (independent variables) and a target attribute (dependent variable) [Ghahramani, 2004; Maimon and Rokach, 2005; Soofi and Awan, 2017]. Supervised techniques can be further categorized into two main types: classification and regression. In regression, the output variable takes continuous values, whereas in classification, the output variable represents class labels [Soofi and Awan, 2017].

Linear Regression

Linear regression, a fundamental and widely used machine learning algorithm, serves as a mathematical method for predictive analysis [Maulud and Abdulazeez, 2020]. It enables the projection of continuous or mathematical variables. The concept of linear regression was first introduced by Sir Francis Galton in 1894.

Linear regression is a statistical technique employed to quantify and assess the relationship between variables of interest and finds broad applications in mathematical research methodologies, particularly when assessing the predicted effects and modelling them against multiple input variables [Maulud and Abdulazeez, 2020; Abdulqader et al., 2020]. It serves as a data evaluation and modelling technique that establishes linear relationships between dependent and independent variables. Through analysis and learning from training data, this method models the relationships between the dependent variables and the independent variables [Maulud and Abdulazeez, 2020; Lim, 2019]. Multiple regression methods are commonly utilized, including Simple Linear Regression [Zou et al., 2003], Multiple Linear Regression [Eberly, 2007, p. 165–187], Decision Tree Regression [Xu et al., 2005], Random Forest Regression [Segal, 2004], etc.

Classification

Classification problems involve categorizing examples into two or more classes, which can be represented by real-valued or discrete input variables. In classification models, it is common to predict the probability of an example belonging to each output class, resulting in continuous values. These probabilities can be interpreted as the likelihood or confidence of an example falling into each class. The predicted probabilities can then be converted into class values by selecting the label with the highest probability [Langford and Schapire, 2005; Speiser et al., 2019].

Various classification techniques have been developed and widely used to tackle different classification problems. Three well-known and widely used classification techniques are Decision Tree Classification [Twa et al., 2005], K-Nearest Neighbor [Cover and Hart, 1967], and Support Vector Machines (SVM) [Jayadeva et al., 2007].

Semantic Segmentation

Semantic segmentation, also known as pixel-based classification, is a crucial task in computer vision where the goal is to assign a specific class to each pixel in an image, ensuring that each pixel is labelled with the corresponding object or region. [Darwish et al., 2003; Garcia-Garcia et al., 2017].

Semantic segmentation is not an independent field of study but rather a progression from classification to localization or detection, enabling more refined inference. Classification serves as the starting point, predicting the object in an image or providing a ranked list of potential objects. The subsequent step, localization or detection, goes beyond classification by not only assigning classes but also providing spatial information such as centroids or bounding boxes. This natural progression leads to semantic segmentation, which aims to make dense predictions by associating each pixel with the corresponding object or region it belongs to [Garcia-Garcia et al., 2017].

The task of semantic segmentation has garnered considerable attention in recent years, not only within the computer vision community but also in other disciplines, such as biomedical imaging and remote sensing. The automated annotation of images plays a crucial role in these domains. As a result, specialized techniques have emerged in various fields to cater to the specific requirements and challenges unique to each discipline. These task-specific peculiarities necessitate the development of tailored approaches that may differ from those commonly used in the computer vision community and vice versa [Diakogiannis et al., 2020].

In the realm of semantic segmentation, there are several commonly used techniques, including Fully Convolutional Networks (FCNs) [Long et al., 2015] and U-Net [Ronneberger et al., 2015]. These architectures have demonstrated their effectiveness in pixel-wise classification tasks. Furthermore, various variations of these methods have been proposed, such as V-Net [Milletari et al., 2016], Residual U-Net (ResUNet) [Diakogiannis et al., 2020], and others, which seek to enhance the performance and address specific challenges in semantic segmentation [Siddique et al., 2021].

2.3.4. Emerging Approaches for Elevation Error Correction

Various approaches have been developed to address elevation errors in Digital Elevation Models (DEMs) by harnessing machine learning techniques and utilizing diverse data sources. One notable technique of linear regression in this domain is the Forest and Building Digital Elevation Model (FABDEM) [Hawker et al., 2022].

Alternative methods, such as the progressive morphological filter, have been proposed for the removal of canopy and infrastructure pixels [Gevaert et al., 2018; Kazimi et al., 2020]. Recent research has also demonstrated the efficacy of convolutional neural networks (CNNs) in this context, highlighting the potential of deep learning techniques [Ji et al., 2020]. For instance, Hu et al. [2016] successfully applied a deep CNN to classify points into ground and non-ground classes, showcasing the utility of this approach.

The residual U-Net framework has gained significant recognition as a fundamental technique in semantic segmentation. Li et al. [2019] presented a novel approach that exploits an improved difference image and leverages the capabilities of the residual U-Net network. Their method aims to enhance the accuracy and effectiveness of change detection, specifically in urban environments. Diakogiannis et al. [2020] introduced a robust framework that has demonstrated high-performance results. Their proposed approach incorporates essential elements, including the residual U-Net architecture, residual connections, atrous convolutions, pyramid scene parsing pooling, and multi-tasking inference. This framework has shown promise in advancing the field of semantic segmentation for aerial imagery analysis. Furthermore, Amini Amirkolaee et al. [2022] utilized a combination of the residual U-Net and a multi-scale fusion strategy to extract Digital Terrain Models (DTMs) from Digital Surface Models (DSMs).

While many existing methods focus on aerial images or employ multiple channels of LiDAR data, there is a scarcity of studies that directly employ single DSM as input. This research gap highlights the value and significance of your thesis, as it contributes to the field by addressing this specific challenge.

FABDEM

FABDEM leverages various data sources, including the Copernicus DEM (COP30, COP-DEM30), forest height data, land cover information, and world settlement footprint data, among others, to detect and correct elevation errors attributed to forests and buildings. The methodology involves pre-classifying different zones, namely urban, forest, and boreal forests, with specific datasets tailored for each zone. Forest Correction utilizes forest height data, as depicted in Figure 2.2, while urban areas are characterized using footprint data. Random forest regression models are employed to eliminate the height contributions of these features, thereby enhancing the accuracy of the original COPDEM30 dataset. The effectiveness of this approach has been demonstrated in various applications, including flood simulation and hydrological studies, where high-quality global terrain information is essential. Notably, the model was also trained using a unique set of reference elevation data from 12 countries, including the Netherlands, and exhibited lower root-mean-square errors (RMSE) compared to other DEMs, such as COPDEM30, MERIT DEM, and SRTM-based DEMs [Hawker et al., 2022].

The structure of FABDEM is illustrated in Figure 2.2, which provides a visual schematic of its inner workings.

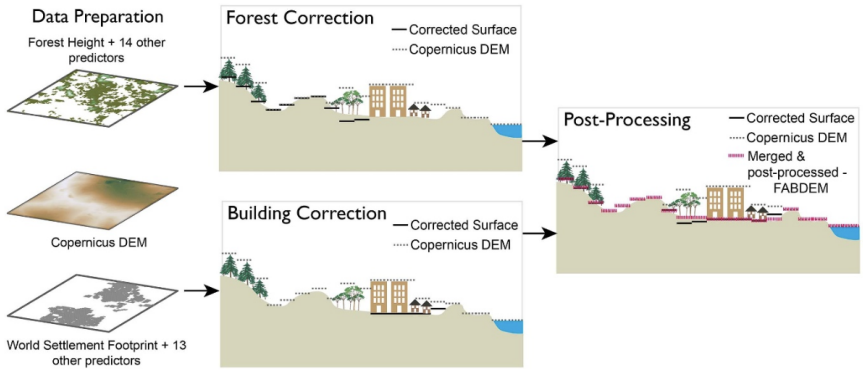


Figure 2.2.: Schematic representation of FABDEM [Hawker et al., 2022]

In the context of FABDEM, it is important to highlight the process of eliminating elevation biases associated with forests and buildings in the COPDEM30 dataset prior to merging it into the digital terrain model (DTM). Following this step, FABDEM employs pit-filling techniques to address depressions and subsequently applies a bilateral filter to reduce speckle noise. While these post-processing steps contribute to a smoother output, it should be noted that they may introduce blurring effects (see 5.2) compared to the method proposed in Chapter 3.

2.3.5. U-Net

The U-Net architecture, designed primarily for image segmentation, has gained significant attention in recent years due to its effectiveness in various applications, including water body extraction, sea-land segmentation, and road extraction [Feng et al., 2019; Shamsolmoali et al., 2019; Chen et al., 2021].

The U-Net architecture is composed of two primary components: the contracting path, also known as the encoder or analysis path, and the expansion path, also referred to as the decoder or synthesis path. The contracting path resembles a standard convolutional network and includes consecutive 3x3 convolutions, ReLU activation, and max-pooling layers. In contrast, the expansion path employs up-convolutions and concatenates features from the contracting path. This distinctive feature of U-Net enables the upsampling of the feature map using 2x2 up-convolutions. It facilitates the learning of localized classification information and enhances the output resolution. The final stage involves a convolutional layer to refine the output and generate a fully segmented image. The symmetric structure of U-Net is responsible for its U-like shape. Unlike most convolutional networks that classify the entire image into a single label, U-Net focuses on pixel-level segmentation [Ronneberger et al., 2015].

Figure 2.3 provides an illustrative depiction of the U-Net architecture.

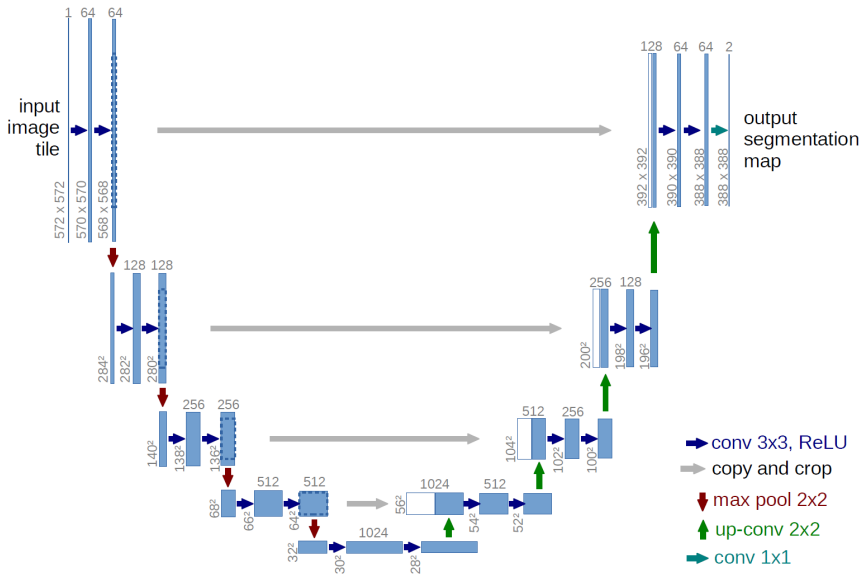


Figure 2.3.: The architecture of a U-Net. [Ronneberger et al., 2015]

2.3.6. Residual Networks (ResNets)

The vanishing gradient problem, characterized by the diminishing gradient norm of earlier layers, poses a challenge during the training process of deep convolutional neural networks (CNNs) [Shehab et al., 2021; Glorot and Bengio, 2010; Bengio et al., 1994]. To mitigate this issue, Residual Network (ResNet) learning methods have been proposed by He et al. [2015]; Ioffe and Szegedy [2015]. ResNet addresses the problem by introducing residual layers, where the output of each layer is convolved with its input, thereby serving as the input for the subsequent layer [Shehab et al., 2021; He et al., 2015]. This residual mapping plays a central role in the residual learning process. Figure 2.4 illustrates the building block of the ResNet, showcasing the convolution of the residual layer's output with its input.

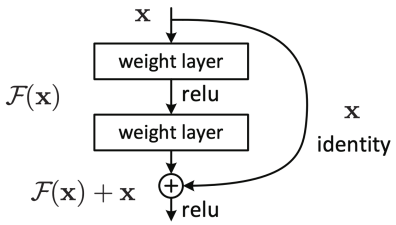


Figure 2.4.: A building block of the U-Net. [He et al., 2015]

In their work, He et al. [2015] propose an improvement to gradient propagation by introducing additional identity connections, which directly backpropagate gradients to earlier layers, thereby simplifying the training process. The ResNet model is composed of two primary blocks: identity blocks and convolutional blocks. Figure 2.5 illustrates the structure of the Identity block, which consists of three components. Firstly, a 2D convolutional layer with a specific filter size and stride is employed. Secondly, batch normalization is applied for channel axis normalization. Lastly, the Rectified Linear Unit (ReLU) activation function is utilized. The second and third components are similar to the first component, except for their distinct filter sizes. The output of the identity block is generated by merging the shortcut connection and the input using the ReLU activation function [Shehab et al., 2021; He et al., 2015; Ioffe and Szegedy, 2015].

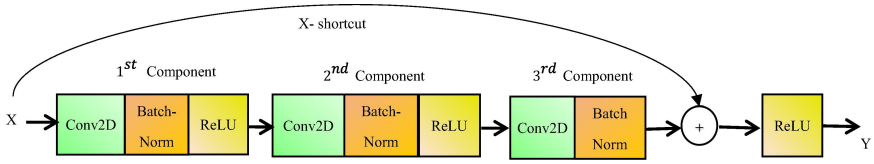


Figure 2.5.: A identity block of the U-Net. [Shehab et al., 2021]

The convolutional block (shown in Figure 2.6) is utilized when the input and output dimensions do not align. In this case, the shortcut connections incorporate linear projection to adjust the dimensions between the input and output. The Convolutional block shares similarities with the Identity block but includes an additional 2D convolutional layer in the shortcut connection. By modifying the shortcut, the input is aligned with the main path, effectively addressing the vanishing gradient problem. This modified shortcut is then combined with the output of the main path, resulting in the final output [Shehab et al., 2021; He et al., 2015; Ioffe and Szegedy, 2015].

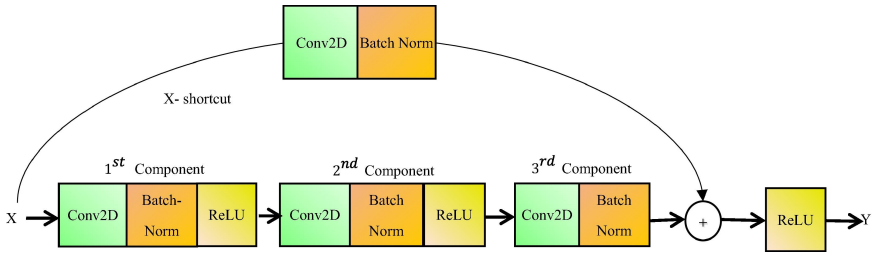


Figure 2.6.: A convolutional block of the U-Net. [Shehab et al., 2021]

The utilization of ResNet and its blocks offers a significant advantage in tackling the vanishing gradient problem, as they enable the smooth propagation of gradients through deep networks. The introduction of shortcut connections ensures that the higher layers can achieve at least the same performance as the lower layers, if not better, enhancing the overall effectiveness and training of the network [Shehab et al., 2021].

2.3.7. Residual U-Net

The Residual U-Net retains the U-shaped structure of the original U-Net and incorporates residual blocks in both the encoder and decoder parts. Within each block, the input is split into two paths: one path processes the information through a series of convolutions, while the other path serves as an identity connection. The outputs from these two paths are then combined to generate the final output of the block.

This modification enables the Residual U-Net to maintain the advantageous segmentation performance of the original U-Net architecture while enhancing the depth of the network to capture more complex features and improve training optimization. As a result, the Residual U-Net is a highly effective model for various remote sensing tasks [Diakogiannis et al., 2020; Ghorbanzadeh et al., 2021; Zhang et al., 2018], including the task at hand.

3. METHODOLOGY

3.1. Residual U-Net

3.1.1. Architecture

The Residual U-Net is a deep learning architecture designed to perform semantic segmentation tasks with high precision. It combines the strengths of the U-Net, which is known for its U-shaped structure [Ronneberger et al., 2015], and the ResNet-18, a popular deep residual network [He et al., 2015]. The architecture consists of two main parts: the downscaling path (Contraction) and the upscaling path (Expansion).

A detailed illustration of the Residual U-Net architecture is shown in Figure 3.1 (rotated).

During the implementation process, to accelerate the training process and mitigate the limitations imposed by insufficient data, a pre-trained ResNet-18 model provided by PyTorch[1] is utilized. This model is designed to accept inputs in the form of 3-layered image-like arrays. The subsequent algorithms and methods adhere to this input structure.

Downscaling Path (Contraction)

The downscaling path is based on the ResNet-18 architecture proposed by He et al. [2015]. It starts with an input of dimensions (64, 64, 3) and passes through a series of Convolution-ReLU-Batch Norm blocks. After the first block, the shape becomes (32, 32, 64). Then, it proceeds through four additional levels, each containing two basic blocks mentioned in Section 2.3.6 with halved height, halved width, and doubled depth. At the end of this path, the output has dimensions (2, 2, 512). The algorithm depicting the process is displayed in Algorithm 3.1.

Upscaling Path (Expansion)

The upscaling path, also known as the Expansion part, follows a U-Net architecture [Ronneberger et al., 2015]. It consists of five levels of upscaling, each doubling the height and width of the input. The output from the last four levels of the downscaling path (ResNet-18) is combined with the corresponding levels in the upscaling path to preserve spatial information. After the last Convolution layer, the output dimensions become (64, 64, 2) since there are two classes to predict. The process is illustrated in Algorithm 3.2.

[1] https://pytorch.org/vision/main/models/generated/torchvision.models.resnet18.html

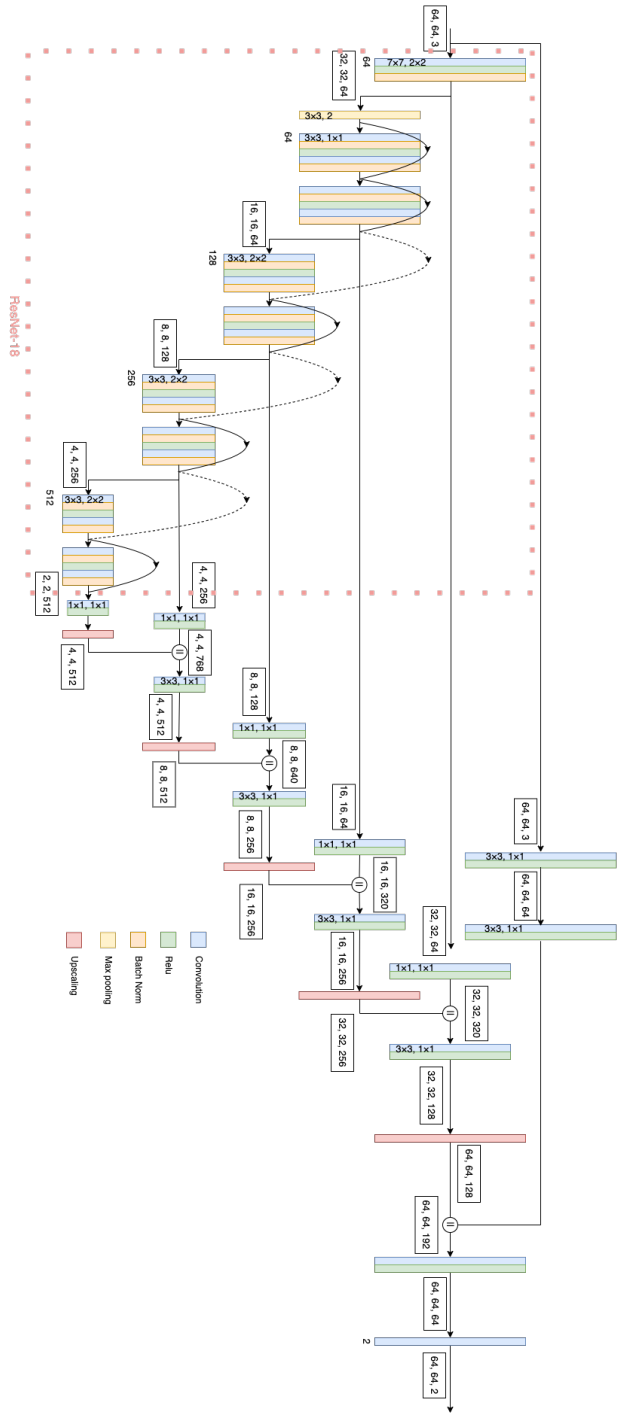


Figure 3.1.: Residual U-Net architecture (rotated)

Algorithm 3.1: CONTRACTION

Input: Preprocessed array-like data with shape (64, 64, 3)

Basic Block: *Conv* 3×3, *Batch Norm*, *ReLU*, *Conv* 3×3, *Batch Norm*

Level 1(*(64, 64, 3)*)**:**
  Conv 7×7, *ReLU*, *Batch Norm*;
  return (32, 32, 64);

Level 2(*(32, 32, 64)*)**:**
  Max Pooling 3×3 ;
  Basic Block ;
  Basic Block ;
  return (16, 16, 64);

Level 3(*(16, 16, 64)*)**:**
  Basic Block ;
  Basic Block ;
  return (8, 8, 128);

Level 4(*(8, 8, 128)*)**:**
  Basic Block ;
  Basic Block ;
  return (4, 4, 256);

Level 5(*(4, 4, 256)*)**:**
  Basic Block ;
  Basic Block ;
  return (2, 2, 512);

3.2. Data Processing

In this section, I discuss the data acquisition, preprocessing, and preparation for model training. I also describe the process of generating a DTM from a DSM using the trained model.

3.2.1. Data Acquisition

Please note that at the time this project is being established (early in 2022), AHN4 is not yet fully published.[2] I downloaded the DSMs and DTMs of 5m resolution from the ArcGIS [3]. Each patch of DSM or DTM had dimensions of 1000 × 1250, representing an area of 5000 m × 6250 m.

Patches that contained a proportion of no-value pixels exceeding 1/8 were excluded from the analysis. This decision was made due to incomplete data generation in certain areas and their proximity to coastal regions, where the AHN4 lacks a water class designation. DSM patches with a considerable number of no-data pixels have the potential to introduce

[2] According to the official website (https://www.ahn.nl/ahn-4), the Eastern Region (Regio Oost) is scheduled for completion in 2022.

[3] https://www.arcgis.com/home/item.html?id=77da2e9eeea8427aab2ac83b79097b1a

Algorithm 3.2: EXPANSION

Output: Prediction of 2 classes with shape (64, 64, 2)

Level 5*((2, 2, 512))*:
  Conv 1×1, *ReLU* ;
  Upscaling ×2;
  return (4, 4, 512);

Level 4*((4, 4, 512))*:
  Stack with the return from Level 4 of *Contraction* process;
  Conv 3×3, *ReLU* ;
  Upscaling ×2;
  return (8, 8, 512);

Level 3*((8, 8, 512))*:
  Stack with the return from Level 3 of *Contraction* process;
  Conv 3×3, *ReLU* ;
  Upscaling ×2;
  return (16, 16, 256);

Level 2*((16, 16, 256))*:
  Stack with the return from Level 2 of *Contraction* process;
  Conv 3×3, *ReLU* ;
  Upscaling ×2;
  return (32, 32, 256);

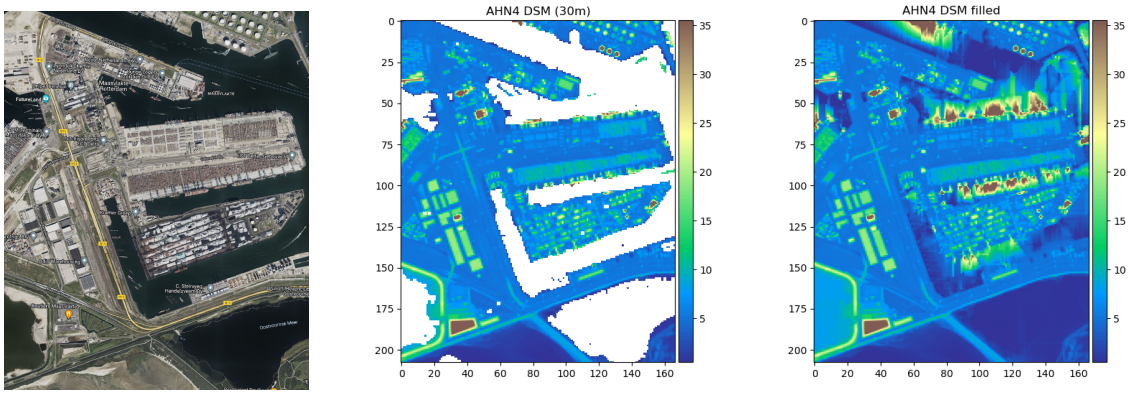
Level 1*((32, 32, 256))*:
  Stack with the return from Level 1 of *Contraction* process;
  Conv 3×3, *ReLU* ;
  Upscaling ×2;
  Stack with the processed **Input** from *Contraction* process;
  Conv 1×1, *ReLU* ;
  Conv 1×1;
  return (64, 64, 2);

artifacts after undergoing preprocessing (refer to Section 3.2.2), consequently disrupting the performance of the model. A specific example demonstrating a DSM patch with an abundance of no-value pixels is illustrated in Figure 3.2.

The division of the patches into training and testing regions was performed in order to evaluate the performance of the model on unseen data. Out of the 505 total patches, a random sample of 50 patches was selected for testing, ensuring that the model never encountered these patches during training. The distribution of the patches can be seen in Figure 3.3.

3.2.2. DEM Preprocessing

As discussed in Section 1.2 and 2.1.1, this thesis utilizes well-classified AHN4 DTMs and DSMs as the input data. To ensure the representativeness and facilitate comparison with global DEMs, the AHN4 data is resampled to a 30m resolution (1 arc-second), which is a



(a) Satellite Image (b) AHN4 DSM @ 30 m resolution (c) Gap-filled DSM

Figure 3.2.: An example of a patch of DSM with a notable abundance of no-value pixels: (a) Satellite Image, (b) AHN4 DSM (resampled to 30 m resolution), (c) The DSM after gap filling. Notice that there are artifacts in the gap-filled DSM.

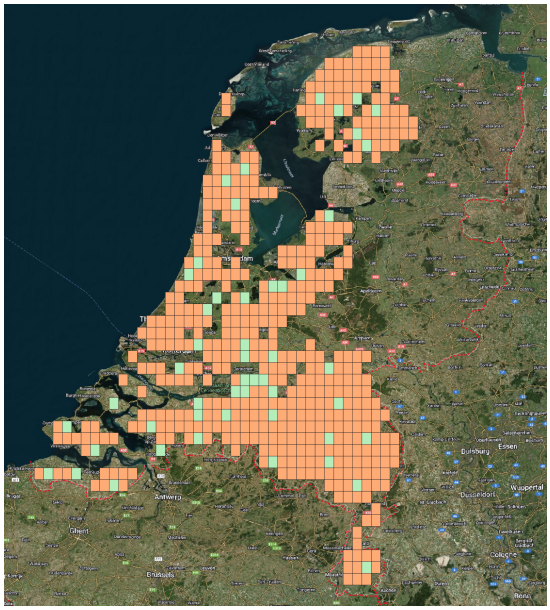


Figure 3.3.: Distribution of the patches. The regions in orange indicate the areas used for training the model, whereas the regions in green denote the areas used for testing the model.

typical resolution for global DEMs. This also allows for improved transferability and better alignment with global DEMs.

The resampling process employs the average method, considering the higher resolution of the original AHN4 DEMs. Further information regarding the implementation methods can be found in Section 4.3.

Subsequently, the ground truth data is generated based on the DSMs and DTMs. A detailed

explanation of this procedure is provided below in the Grond Truth section.

To handle no-data values in the DSMs, Inverse Distance Weighting (IDW) interpolation is applied (refer to Section 4.3.2). Inverse Distance Weighting (IDW) interpolation is a mathematical method used to estimate unknown values based on known data points. It operates under the assumption that closer values have a stronger relationship compared to those that are further apart. IDW works by assigning weights to neighbouring points based on their distance to the target location. These weights are inversely proportional to the distance, meaning closer points have a higher weight in the interpolation process.

The filled DTMs are then used to compute slope maps and roughness maps. These filled DTMs, along with the slope maps and roughness maps, are combined into 3-layer arrays. A comprehensive discussion of the implementation details and methods can be found in Section 4.3.

To prepare the data for training, both the arrays and the ground truth maps are divided into 64x64 slices, with a comprehensive explanation of this procedure in Section 3.2.2. These sliced arrays and ground truth maps are employed as inputs and labels for the training process, respectively. Figure 3.4 presents a visual representation of this data preprocessing process.

In addition to these processes, data normalization is applied prior to feeding the data into the model. The detailed description of the normalization technique can be found in Section 4.2.3. Normalization plays a crucial role in standardizing the input data and ensuring that the features are on a similar scale. By applying normalization, the range and distribution of the data are adjusted, which can improve the performance and convergence of the model during training.

The detailed numerical information of inputs can be found in Section 4.1.

Ground Truth

As the main goal is to remove pixels representing the canopy and infrastructure, a labelling process was conducted to assign appropriate values. Pixels requiring removal were labelled as 1, while those that should be retained were labelled as 0. This labelling procedure involved a comparison between selected pixels in the Digital Surface Model (DSM) and corresponding pixels in the Digital Terrain Model (DTM). The labelling criteria were as follows:

1. Terrain pixels (labelled as 0),
2. Pixels representing trees and buildings (labelled as 1).

If the corresponding pixels either contained no values or had values below a designated threshold (commonly set at 0.5 in practice), they were assigned a label of 1; otherwise, a label of 0 was assigned. Figure 3.5 illustrates the labelling process.

Slicing

In this study, the term ”slicing” refers to the procedure of partitioning an image-like array into smaller segments, each characterized by a specific width and height. To maximize data utilization and preserve all relevant information for subsequent reassembly, Algorithm 3.3 has been employed. To address potential discontinuities at the boundaries of the sliced

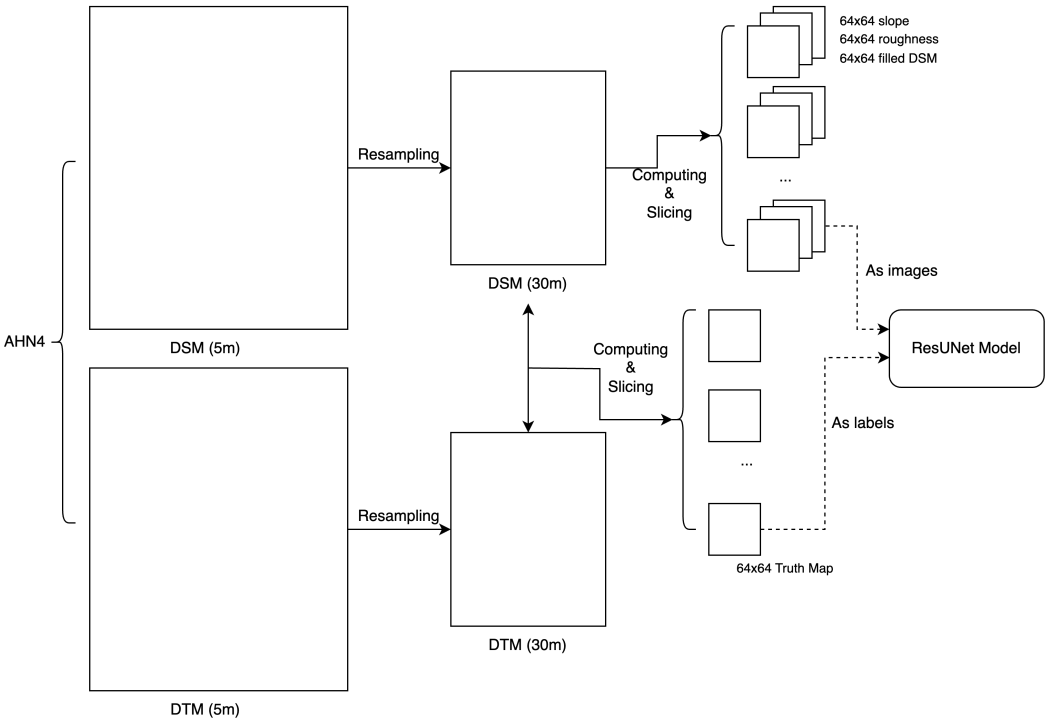


Figure 3.4.: Generation of inputs and labels for the model from DSMs and DTMs.

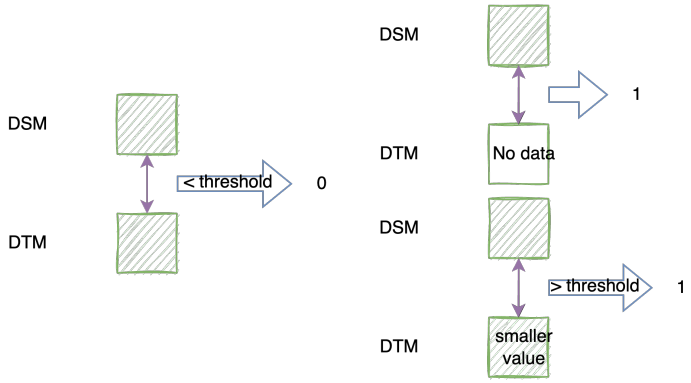


Figure 3.5.: An illustration shows the relationship between the pixels and the labels.

segments, the slicing method incorporates overlaps between adjacent segments, which is inspired by the overlap-tile strategy employed by U-Net [Ronneberger et al., 2015].

During the experimental phase, I have set the overlap ratio to 15% following the approach adopted by U-Net [Ronneberger et al., 2015]. This overlap ratio corresponds to a 9-pixel overlap when the segment size is fixed at 64 pixels. By incorporating this overlap, It can

be ensured that neighbouring segments share common information, facilitating smoother transitions and reducing potential artifacts during the reassembly process. This overlap strategy enhances the overall coherency and integrity of the reassembled image.

Algorithm 3.3: SLICING (h, w)

Input: An image-like array A with width H and height W, overlap ratio r
Output: A dictionary whose keys are locations and values are arrays with height h and width w

$dictionary \leftarrow \{\}$
$Step_h \leftarrow h - \lfloor h \cdot r \rfloor$
$Step_w \leftarrow w - \lfloor w \cdot r \rfloor$
$Last_h \leftarrow H - h$
$Last_w \leftarrow W - w$
$hOffsets \leftarrow list(\text{from 0 to } Last_h, \text{ in steps of } Step_h)$
$wOffsets \leftarrow list(\text{from 0 to } Last_w, \text{ in steps of } Step_w)$
if $length(hOffsets) = 0$ ***or*** $hOffsets[-1] \neq Last_h$ **then**
    append $Last_h$ **to** $hOffsets$
if $length(wOffsets) = 0$ ***or*** $wOffsets[-1] \neq Last_w$ **then**
    append $Last_h$ **to** $wOffsets$
for $hOffset$ ***in*** $hOffsets$ **do**
    for $wOffset$ ***in*** $wOffsets$ **do**
        $array \leftarrow A[hOffset : hOffset + h, wOffset : wOffset + w]$
        $key \leftarrow (hOffset : hOffset + h, wOffset : wOffset + w)$
        $dictionary[key] \leftarrow array$
return $dictionary$

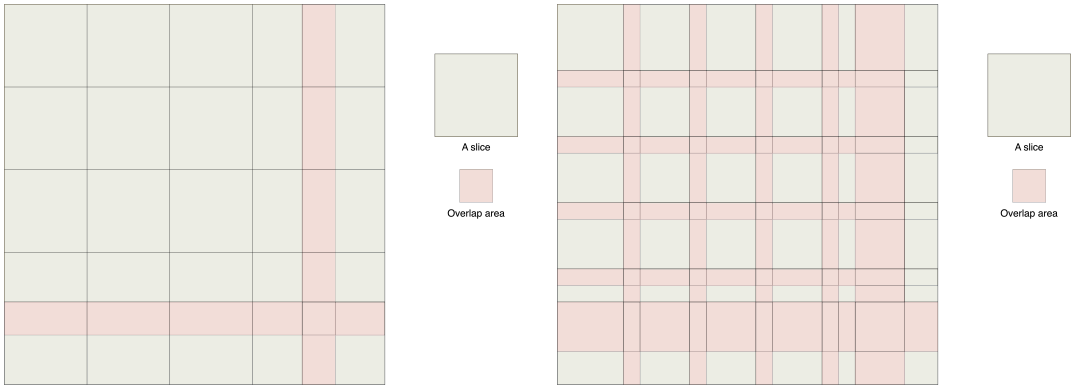
It is worth highlighting that when the overlap ratio is set to 0%, overlaps only occur at the right and bottom edges of the segments, thereby preserving the exact boundaries of the original image. This configuration is particularly valuable in scenarios where maintaining the precise image boundaries is of utmost importance, ensuring accurate representation and analysis.

To illustrate the impact of incorporating inner overlaps in the segments, Figure 3.6 provides visual examples showcasing the differences between segments with and without such overlaps.

3.3. Generating DTMs & Evaluation

The input Digital Surface Model (DSM) undergoes a similar processing methodology as outlined in Section 3.2.2, encompassing computations and slicing. These processed inputs are subsequently fed into the trained model, generating corresponding prediction maps. To assemble a comprehensive prediction map that matches the original shape of the input DSM, the 64x64 individual prediction maps are stitched together using a method that will be discussed in Section 3.3.1.

Further processing involves the identification of pixels in the input DSM that correspond to a value of ”1” in the prediction map. These identified pixels are then set to a no-value state in



(a) Slicing without (inner) overlaps (b) Slicing with overlaps

Figure 3.6.: Examples of a 23x23 array sliced into 5x5 segments, with overlapping areas highlighted in red: (a) Slicing without inner overlaps; (b) Slicing with a 1-pixel width overlap.

the input DSM. Subsequently, interpolation is applied to the partially emptied DSM, resulting in the final Digital Terrain Model (DTM). The entire process, from the initial processing of the DSM to the creation of the final DTM, is depicted in Figure 3.7.

3.3.1. Stitching

The process of stitching serves as the complementary procedure to slicing, as outlined in Section 3.2.2. Its objective is to integrate the individual slices in a specific order to create a complete array. To accomplish this, Algorithm 3.4 is employed to assemble the results obtained from the slicing process and form a comprehensive array.

It is noteworthy that in the context of this study, slicing and stitching techniques are applied to distinct types of arrays. Specifically, the slicing operation is utilized on the input image-like arrays, whereas stitching is implemented for the reassembly of prediction maps. Thus, each process serves a unique function within the overall methodology.

Algorithm 3.4: STITCHING

Input: List of prediction slices *list*, Dictionary of slice positions *dictionay*
Output: Stitched prediction map P

$h \leftarrow$ get_end_height(*dictionay*)
$w \leftarrow$ get_end_width(*dictionay*)
$P \leftarrow$ zeros(h, w)
$P \leftarrow P + 0.5$
foreach *key in dictionary* **do**
    $h_0, h_1 \leftarrow$ get_height_bounds(*dictionay*, *key*)
    $w_0, w_1 \leftarrow$ get_width_bounds(*dictionay*, *key*)
    $P[h_0 : h_1, w_0 : w_1] \leftarrow$ closest_to_0.5(*list*[*key*], $P[h_0 : h_1, w_0 : w_1]$)
return P

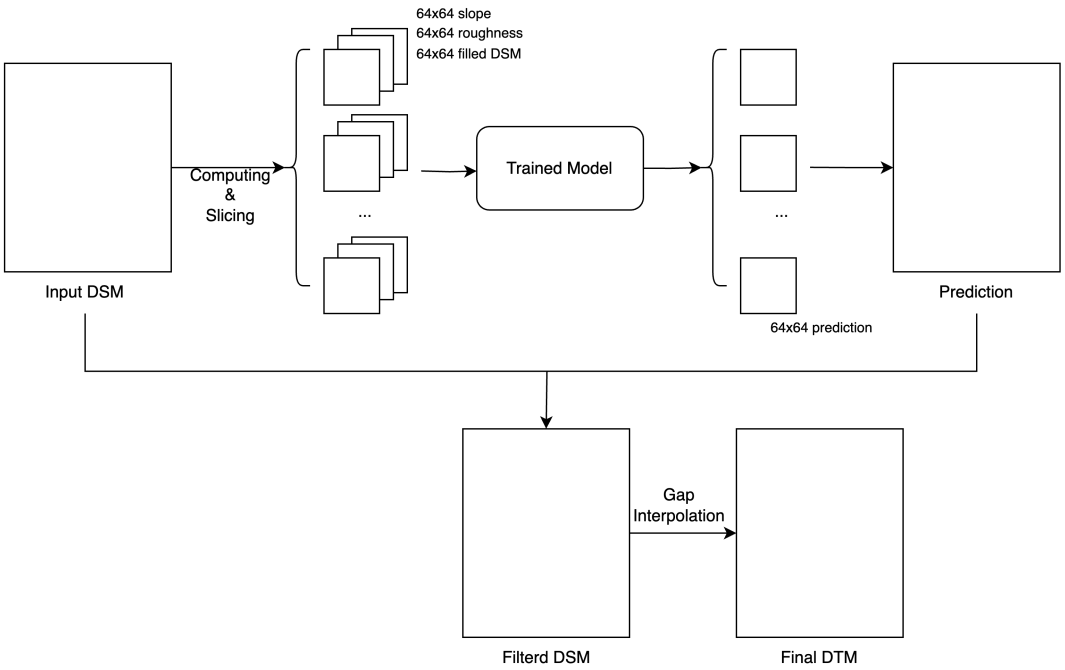


Figure 3.7.: The process of generating a DTM from a DSM using the trained model.

For binary prediction maps, it compares the absolute differences of the slice's pixels and 0.5 with the absolute differences of the merged image's pixels and 0.5. It chooses the pixel value (from either the slice or the merged image) that has the smaller difference from 0.5. It is worth noting that the method is equivalent to averaging the prediction values for the same pixel when overlaps occur. By averaging the predictions, the model takes into account the information from both overlapping segments, resulting in a more comprehensive and representative estimation for that particular pixel. This approach helps to mitigate any potential discrepancies or inconsistencies that may arise at the boundaries of the segments and improves the overall accuracy and continuity of the predictions.

This approach also ensures that a no-data pixel (with value 0.5) in the slice does not overwrite a valid pixel (with value 0 or 1) in the merged image.

3.3.2. Root-mean-square Error (RMSE)

The root-mean-square error (RMSE) is a frequently used measure of the differences between values predicted by a model or an estimator and the values observed. It's a standard way to measure the error of a model in predicting quantitative data. Formally, the RMSE of an estimator measures the square root of the average of the squares of the errors—that is, the square root of the mean square error.

The RMSE serves to aggregate the magnitudes of the errors in predictions for various data points into a single measure of predictive power. RMSE is a good measure of accuracy, but only when comparing prediction errors of different models or model configurations for a

particular variable and not between variables of different scales. RMSE is widely used in valuating DEMs [Mesa-Mingorance and Ariza-López, 2020; Okolie and Smit, 2022; Hawker et al., 2022; Gonga-Saholiariliva et al., 2011; Hawker et al., 2018].

The equation for RMSE is given as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3.1}$$

In this equation, n represents the total number of observations, y_i represents the observed values, and $\hat{y}_i$ represents the predicted values. The term inside the sum calculates the square of the difference between the observed and predicted values for each observation, and the sum of these squared differences is then averaged over all observations. The square root of this average gives the RMSE. The RMSE is always non-negative, and a value of 0 (almost never achieved in practice) would indicate a perfect fit to the data. In general, a lower RMSE is better than a higher one.

3.4. Differences Between My Method and FABDEM

To further illustrate the differences between my method and FABDEM, a comparison table is presented in Table 3.1.

	My Method	**FABDEM**
Inputs	Resampled AHN4 DEMs	COPDEM30, AHN3, forest heights, ICESAT2 canopy heights, Travel Times, Night Lights, WorldPop Constrained, GHS Urban Centre Database, World Settlement Footprint
Outputs	DTMs	DTMs
Machine Learning Method	Semantic Segmentation, Residual U-Net	Linear Regression, Random Forest
Postprocessing	IDW interpolation	Pit-filling and bilateral filter

Table 3.1.: Comparison with FABDEM

4. IMPLEMENTATION & EXPERIMENTS

4.1. Code & Data Specifications

In the interest of reproducibility and transparency, all code employed in this study have been made publicly available. The code includes data preprocessing, model training, and performance evaluation. The complete set of code, along with the necessary instructions for their use, have been uploaded to a GitHub repository. The link to this repository is provided in the appendices of this paper. It is our hope that this open-source approach will facilitate further research in this domain and potentially accelerate advancements in the field.

The data for training, validation, and testing in this model follows a ratio of 7:2:1. A detailed tabulated representation of the input data (training and validation) used in the preprocessing phase, including numbers, sizes, heights, and widths, can be found in Table 4.1.

Input data	**Number**	**Shape (height, width)**
Orignal AHN4 DSM	455	(1250, 1000)
DSM @ 30 m resolution	455	(208, 167)
Image-like array after slicing	5460	(64, 64)
Image-like array after augmentation	43680	(64, 64)

Table 4.1.: Number and shape of inputs in different stages

4.2. Residual U-Net

4.2.1. Implementation Details

In this study, I employed the Residual U-Net model, which was trained using the Adam optimizer [Kingma and Ba, 2017] with an initial learning rate of 1×10^{-4}. This optimizer was specifically selected for its ability to adjust the model parameters that necessitated gradient updates. To dynamically adapt the learning rate during the training process, I incorporated a learning rate scheduler. This scheduler was a step type with a step size of 8 and a gamma value of 0.1. This configuration served to decrease the learning rate by a factor of 0.1 every 8 epochs, a strategy that has been shown to foster convergence and stability throughout the training phase.

The model was implemented utilizing the PyTorch deep learning framework [Paszke et al., 2019], which was chosen due to its flexibility and efficiency. I employed the rectified linear

unit (ReLU) as the activation function throughout the entire model architecture. ReLU was selected as it is known to help alleviate the vanishing gradient problem, a common issue in deep learning, and as a result, it can hasten the convergence during the training process.

The selected loss function was a combination of binary cross-entropy with logits and the Dice loss. This hybrid loss function was utilized in order to allow the model to strike a balance between pixel-wise classification accuracy and region overlap performance, both of which are crucial for the task at hand. The model was trained over a total of 30 epochs, utilizing the aforementioned optimizer, learning rate scheduler, and loss function.

The selection of these specific training parameters and implementation details was not arbitrary. Instead, they were carefully chosen with the objective of optimizing the model's performance with respect to convergence speed, generalization ability, and computational efficiency, thus ensuring robust and reliable results.

4.2.2. Combinations of Inputs

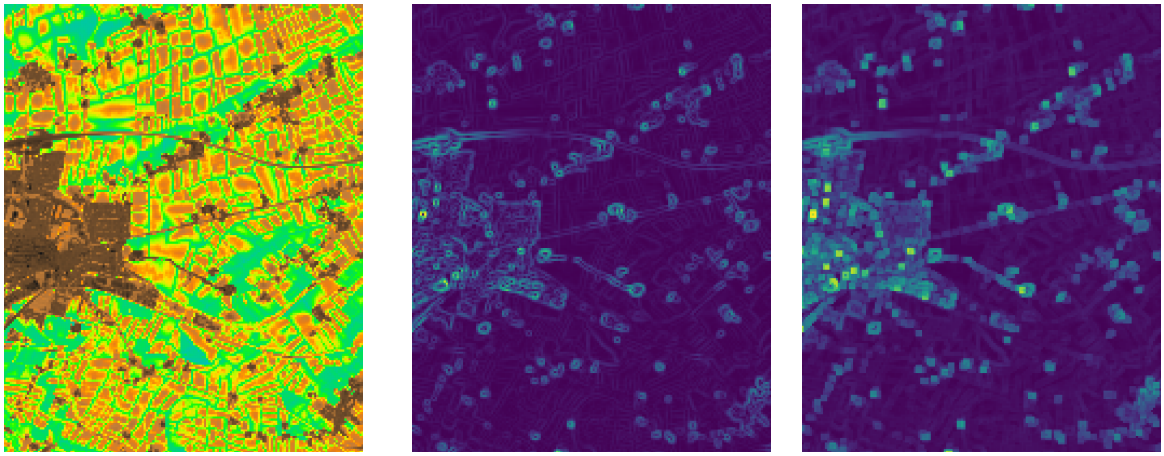
As delineated in Chapter 3, the model necessitates inputs that are arrays with a depth of three, implying the incorporation of three distinct layers. An array of analytical map products can be directly computed from Digital Elevation Models (DEMs), including slope maps, aspect maps, Terrain Ruggedness Index (TRI) maps, Topographic Position Index (TPI) maps, and roughness maps.

A prerequisite for the model input is a filled Digital Surface Model (DSM). Existing algorithms for DTM extraction from DSMs frequently utilize slope calculations derived from a pre-existing DTM Gevaert et al. [2018]; Debella-Gilo [2016]. In the realm of 2D raster methods, slope is often harnessed to define segmentation boundaries Gevaert et al. [2018]; Hingee et al. [2016]; Tomljenovic et al. [2016]. Therefore, I have elected to utilize the slope map as the second layer in the model.

Given this fixed second layer, the third layer can be any one of the other map products, resulting in four potential combinations of map products for model input:

1. Filled DSM, Slope map, Aspect map
2. Filled DSM, Slope map, TRI map
3. Filled DSM, Slope map, TPI map
4. Filled DSM, Slope map, Roughness map

Given the constraints of time, a comprehensive evaluation of all possible combinations for optimal performance was not feasible. The study primarily investigated the first and fourth combinations, out of which the fourth combination - comprising of the Filled DSM, Slope map, and Roughness map - exhibited a marginal improvement in performance. Figure 4.1 provides an illustrative example of this combination. It should be noted that in subsequent training and testing procedures, the Slope map was positioned as the third layer as opposed to the second. The sequence of these layers could potentially influence performance, an aspect that warrants further exploration in future work.



(a) Filled DSM at 30 m resolution (b) Slope map (c) Roughness map

Figure 4.1.: An example of the combination: (a) Filled DSM at 30 m resolution, (b) Slope map, (c) Roughness map.

4.2.3. Normalization

As discussed in Section 3.2.2, the process of normalization plays a crucial role in standardizing the input data to ensure that the features are on a similar scale. The model utilizes a pre-trained ResNet-18 for the downscaling path (refer to Section 3.1.1). The pre-trained model comes equipped with its initial parameters for normalization, tailored to real-world images. The values for an image array typically fall within the range of (0, 255). However, the elevation values can significantly exceed this range, while the values in the slope map and the roughness map tend to be more confined. Hence, it is important to note that the pre-trained model's normalization scheme may not be suitable for the specific requirements of this study. Therefore, in order to align with the dataset used in this study, the normalization parameters, including the means and standard deviations, are computed from the training data. These computed parameters are then used to normalize the input data for the model, ensuring compatibility and optimal performance.

The normalization formula is given as follows:

$$\text{output} = \frac{\text{input} - \text{mean}}{\text{std}} \tag{4.1}$$

4.2.4. Threshold Set for Generating Ground Truth

As previously delineated in Section 3.2.2, some labels are generated by determining whether the difference between corresponding pixels from a Digital Surface Model (DSM) and a Digital Terrain Model (DTM) exceeds a specific threshold. This threshold embodies the physical distinction between a DSM and a DTM in this study.

The value of the threshold, therefore, has a direct influence on the generated labels. An overly large or small threshold value can compromise model performance by introducing

inconsistencies that may confuse the model during training. In the initial stages of this research, a threshold value of 2.0 was employed. This was later revised to 0.5, resulting in noticeable improvements in model performance.

As indicated in Section 4.2.2, however, a comprehensive evaluation of all possible threshold values was not feasible due to time and computational constraints. Future research could potentially explore a wider range of threshold values to further optimize model performance.

4.3. Data Processing

4.3.1. Resampling

In this study, the resampling of Digital Elevation Models (DEMs) is performed using the gdalwarp function from the Geospatial Data Abstraction Library (GDAL)[1]. GDAL is a powerful and widely used open-source library designed for reading, writing, and manipulating geospatial raster data.

The gdalwarp utility is a versatile tool extensively utilized for image mosaicing, reprojection, and warping purposes. It offers the capability to reproject images into various supported projections and can also apply Ground Control Points (GCPs) if available in the raw image data containing control information.

Considering that the original DEMs have a 5 m resolution and the objective is to target lower resolutions, I have opted for the average resampling technique. This method computes the weighted average of all non-NODATA contributing pixels, resulting in the desired lower-resolution representation.

4.3.2. Inverse Distance Weighting (IDW) Interpolation

In this study, the "fillnodata" function from the rasterio package [2] is utilized to handle gaps in raster data through Inverse Distance Weighting (IDW) interpolation. This algorithm [3] performs interpolation for all designated nodata pixels, which are identified by zeros in the mask. To determine the interpolated values, a conic search is conducted in four directions from each pixel. The search aims to locate neighboring values that are used in the inverse distance weighting process, resulting in the estimation of missing values in the raster dataset. The search distance is set to 100 by default.

4.3.3. DEM Anaylize

The analytical map products, as previously discussed in the section "Combinations of Inputs," were computed utilizing functions provided by the gdaldem tool from the Geospatial Data Abstraction Library (GDAL).

[1] https://gdal.org/programs/gdaldem.html
[2] https://rasterio.readthedocs.io/en/stable/
[3] From the document on https://rasterio.readthedocs.io/en/stable/api/rasterio.fill.html

The gdaldem tool provides a suite of functionalities that enable the generation of various derivative products from digital elevation models (DEMs), including slope, aspect, and terrain ruggedness index (TRI) maps, among others. In this study, I leveraged the capabilities of gdaldem to compute the required analytical map products which served as integral components of the input data combinations.

These computational procedures are essential for enhancing the feature diversity of the input data, consequently improving the ability of the model to capture complex topographical characteristics and patterns. The application of these analytical map products, therefore, plays a pivotal role in bolstering the performance and reliability of the ensuing machine learning tasks.

4.3.4. Data Augmentation

In an effort to augment the available data and enhance the robustness of the model, a data augmentation process was implemented within the scope of this study. The data augmentation techniques used involved expanding the original input dataset to include not only the original image-like arrays but also their 90-degree, 180-degree, and 270-degree rotations, along with their mirrored counterparts and the 90-degree, 180-degree, and 270-degree rotations of these mirrored versions. By applying these data augmentation techniques, the size of the input dataset was effectively increased eight-fold, thereby providing a more comprehensive training set for the model.

5. RESULTS

5.1. Segmentation Accuracy

The accuracy of the Residual U-Net model was assessed by comparing the predicted DTM label maps, as discussed in Section 3.3, with the corresponding ground truth maps, as presented in Section 3.2.2.

The results of the accuracy assessment are summarized in Table 5.1:

	Accuracy
Worst-performance patch	86.03%
Best-performance patch	98.93%
Average of 50 test patches	94.21%

Table 5.1.: Accuracy analysis

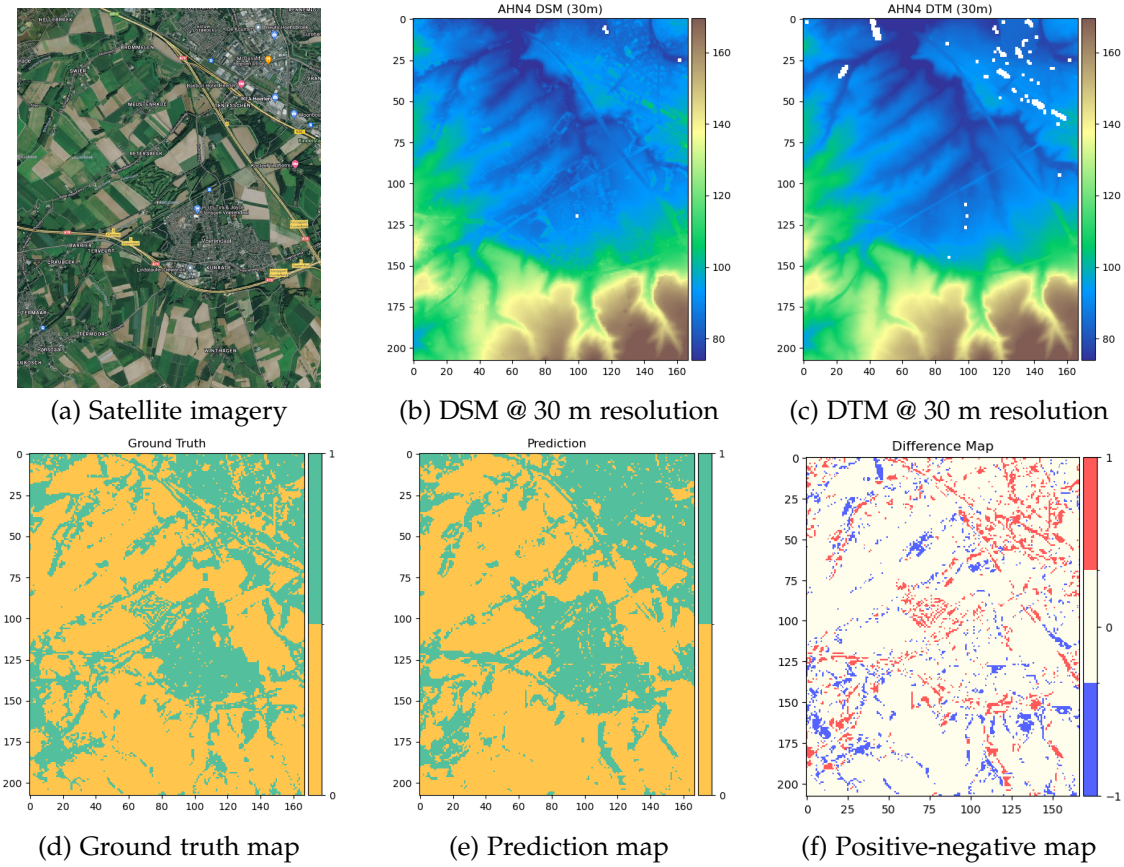
The Residual U-Net model achieved an overall mean accuracy of 94.11%, with individual patch accuracies ranging from 98.84% to 85.92%. The minor variation between the different patches highlights the model's effectiveness in accurately classifying terrain, vegetation, and man-made structures across the test dataset. This observation is further supported by the visual comparison of the ground truth and prediction maps in Figures 5.1 and 5.2, which exhibit striking similarities.

To gain further insights into the differences between the predictions and ground truth, as well as to identify potential segmentation patterns, I generated positive-negative maps by subtracting the ground truth from the prediction maps. These visual representations can be found in Figures 5.1 and 5.2.

In the positive-negative map, pixels with a value of 1 indicate instances where the prediction map erroneously classified the pixel as 1 while the ground truth map labelled it as 0. Conversely, pixels with a value of -1 represent cases where the prediction map incorrectly identified the pixel as 0 while the ground truth map classified it as 1. A value of 0 indicates correct classification.

Figure 5.2 suggests that the model demonstrates superior performance in flat plains with sparse housing. However, in Figure 5.1, the model faces difficulties in areas with rapidly varying elevations and in mountainous regions. These findings indicate that while the model is effective in certain landscapes, there is still room for improvement in its ability to accurately predict DTMs across diverse terrains.

Analyzing the visual results, it is evident that the positive and negative pixels are distributed randomly, suggesting that the model does not exhibit a bias towards either removing or retaining more pixels. The similar proportions of 1's and -1's indicate that the model does



(a) Satellite imagery (b) DSM @ 30 m resolution (c) DTM @ 30 m resolution

(d) Ground truth map (e) Prediction map (f) Positive-negative map

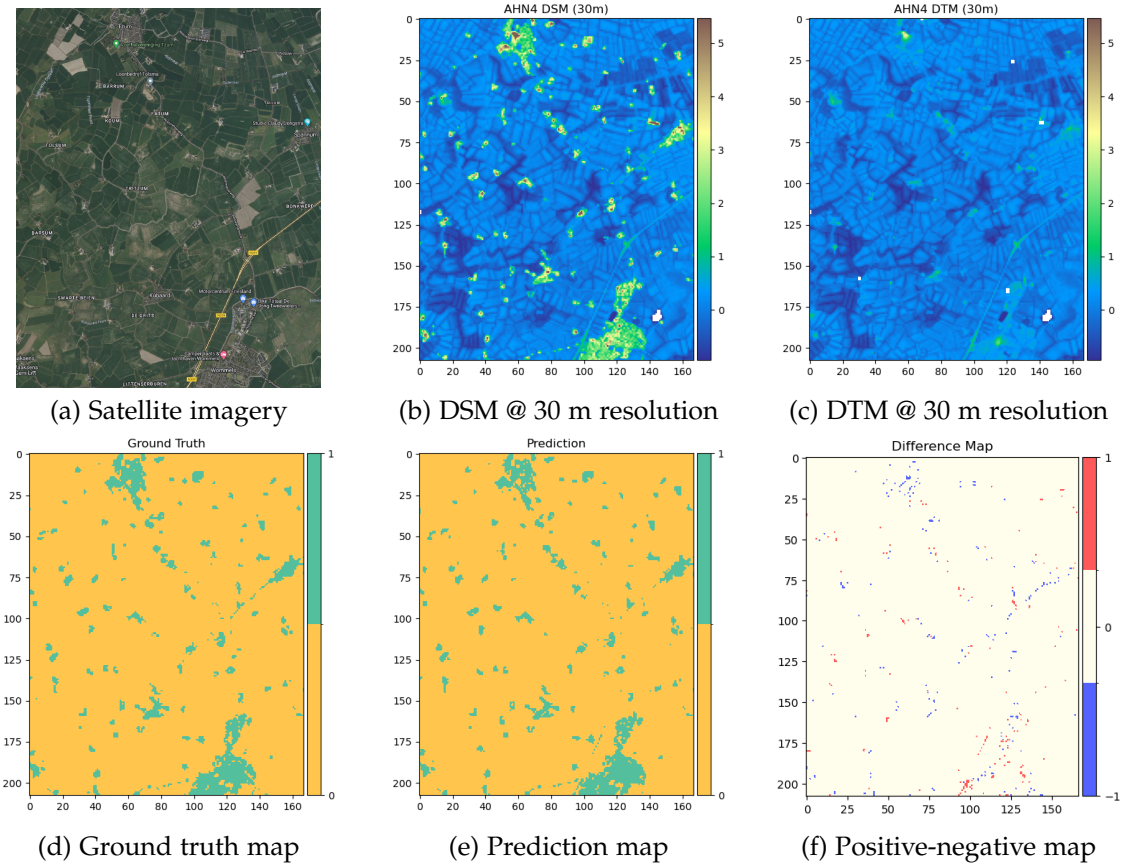
Figure 5.1.: The worst-performing patch: (a) Satellite imagery, (b) DSM (resampled), (c) DTM (resampled), (d) Ground Truth computed from the DSM and the DTM, (e) Prediction from the model, and (f) positive-negative map (subtracting the ground truth from the prediction maps)

not display an "aggressive" behavior. However, notable observations can be made regarding the model's tendency to remove roads, which are visually identifiable as red dotted lines in the positive-negative maps depicted in Figures 5.1 and 5.2.

In summary, the model exhibits a high level of accuracy in the segmentation task, effectively differentiating between diverse non-terrain and terrain features. While certain landscapes present challenges, the Residual U-Net demonstrates its capability in producing precise elevation data across an extensive range of surface types.

5.2. Comparison with FABDEM

As mentioned in Section 2.3.4, FABDEM primarily utilizes COP30 for training, along with several local reference elevation datasets Hawker et al. [2022]. According to its supplemen-



(a) Satellite imagery (b) DSM @ 30 m resolution (c) DTM @ 30 m resolution

(d) Ground truth map (e) Prediction map (f) Positive-negative map

Figure 5.2.: The best-performing patch: (a) Satellite imagery, (b) DSM (resampled), (c) DTM (resampled), (d) Ground Truth computed from the DSM and the DTM, (e) Prediction from the model, and (f) positive-negative map (subtracting the ground truth from the prediction maps)

tary documentation [1], FABDEM employs AHN3 as an additional reference elevation dataset for areas in the Netherlands, making it a suitable example for comparison.

To better evaluate the method and model, I computed the RMSEs (see Section 3.3.2) for three sets: 1) the DTMs generated by the trained model from AHN4 DSMs against the AHN4 DTMs; 2) FABDEM against the AHN4 DTMs; 3) the DTMs generated by the trained model from COP30 DSMs against the AHN4 DTMs.

All the RMSEs are computed using the same reference (AHN4 DTMs), and FABDEM and COP30 are remapped to the areas of the patches for testing using QGIS [2].

Analyzing Table 5.2, I observe that the DTMs generated by the trained model from AHN4 DSMs exhibit the highest performance, while those derived from COP30 yield the least favorable results. FABDEM, on the other hand, lies in an intermediate position, with DTMs

[1] Supplementary data can be found at https://iopscience-iop-org.tudelft.idm.oclc.org/article/10.1088/1748-9326/ac4d4f/data

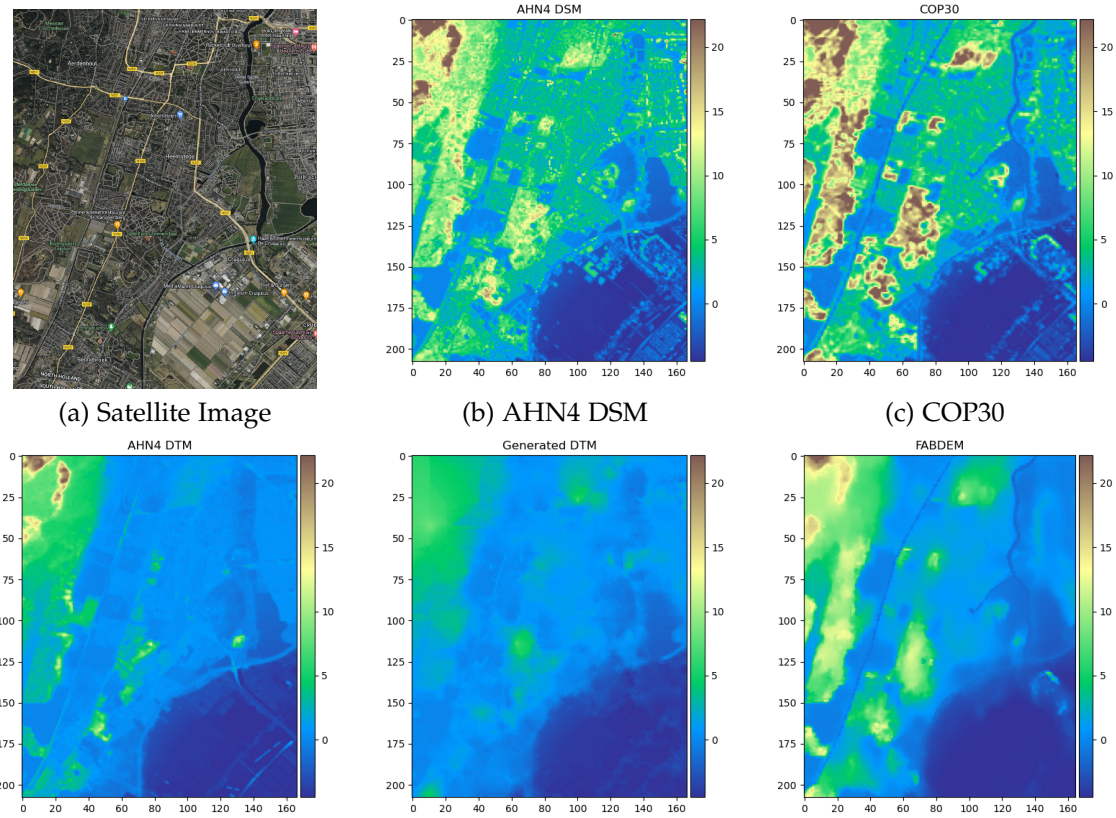
[2] https://www.qgis.org/en/site/

DTM set	RMSE	best RMSE*	worst RMSE*
DTMs generated from AHN4 DSMs	0.59	0.13	1.67
FABDEM	0.93	0.37	2.96
DTMs generated from COP30 DSMs	1.18	0.40	3.76

Table 5.2.: (RMSE) of Different DTMs Compared to AHN4 DTMs.Note: "*" denotes the RMSE values for the best-performance and worst-performance areas.

generated from COP30 demonstrating similar performance in terms of average and best RMSE values.

Furthermore, the analysis reveals that both FABDEM and the model encounter challenges in the same area of the test dataset, resulting in the poorest performance (with an RSME of 1.71 for the DTM generated from AHN4 DSM and 2.96 for FABDEM) across all patches. Figure 5.3 provides a visual representation of the generated DTMs and FABDEM, alongside the corresponding satellite image, AHN4 DSM, COP30, and AHN4 DTM (serving as the reference) for this particular area.



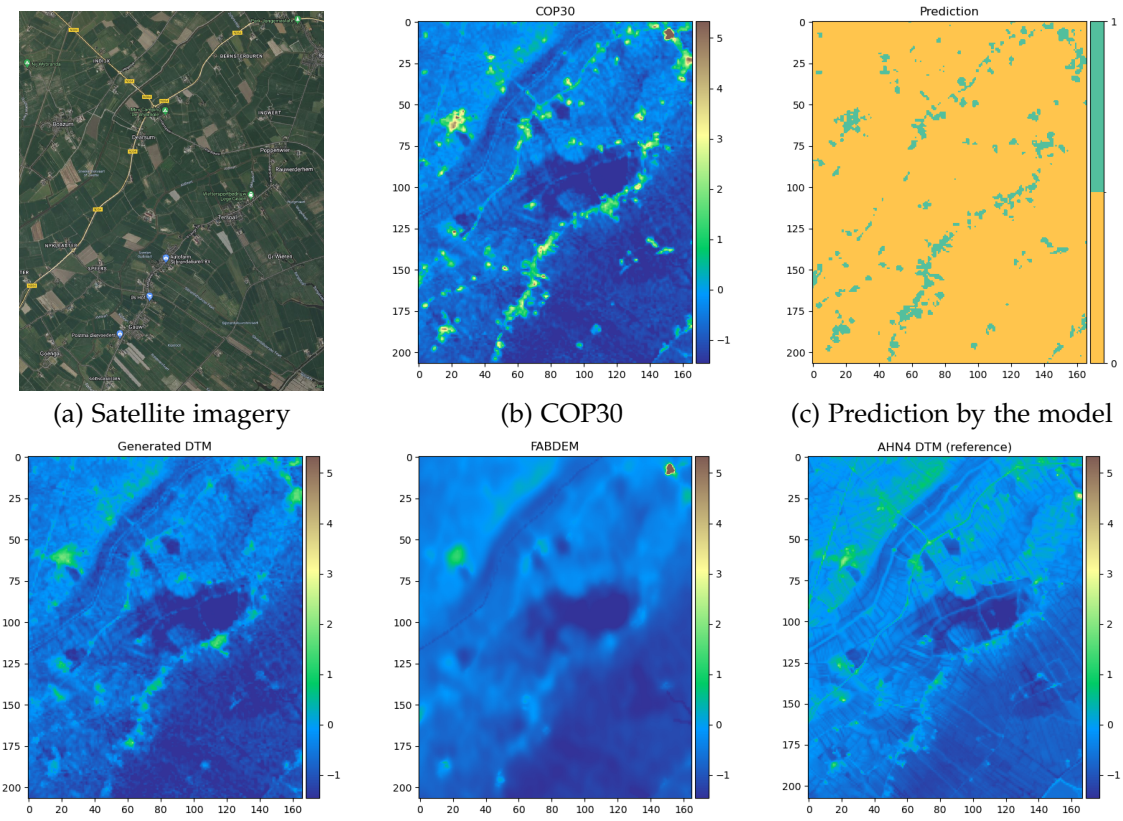
(a) Satellite Image (b) AHN4 DSM (c) COP30

(d) AHN4 DTM (Reference) (e) Generated DTM (f) FABDEM

Figure 5.3.: Comparison of DTMs: (a) Satellite Image, (b) AHN4 DSM, (c) COP30, (d) AHN4 DTM (reference), (e) Generated DTM by the model, and (f) FABDEM.

From the figure, it can be observed that the area represents a complex scenario, encompassing urban environments, agricultural lands, forests, and mountainous regions. When comparing the generated DTM to the FABDEM, it is evident that the DTM more closely resembles the reference DTM, which is also consistent with the differences in RMSE values. Owing to the methodological differences (removing a specific elevation bias versus removing selected pixels and interpolating from neighboring pixels), FABDEM tends to retain more elevated values, predominantly those corresponding to trees. On the other hand, the method smooths out more elevated points, as demonstrated by the complete removal of the small hill in the top left corner of the image.

When comparing the DTMs generated from AHN4 DSMs and FABDEM DTMs with those generated from COP30 DSMs, there is a significant variation in performance across different regions. With an overall RMSE of 1.18, the highest RMSE reaches 3.76, while the lowest RMSE is 0.40 for specific areas.

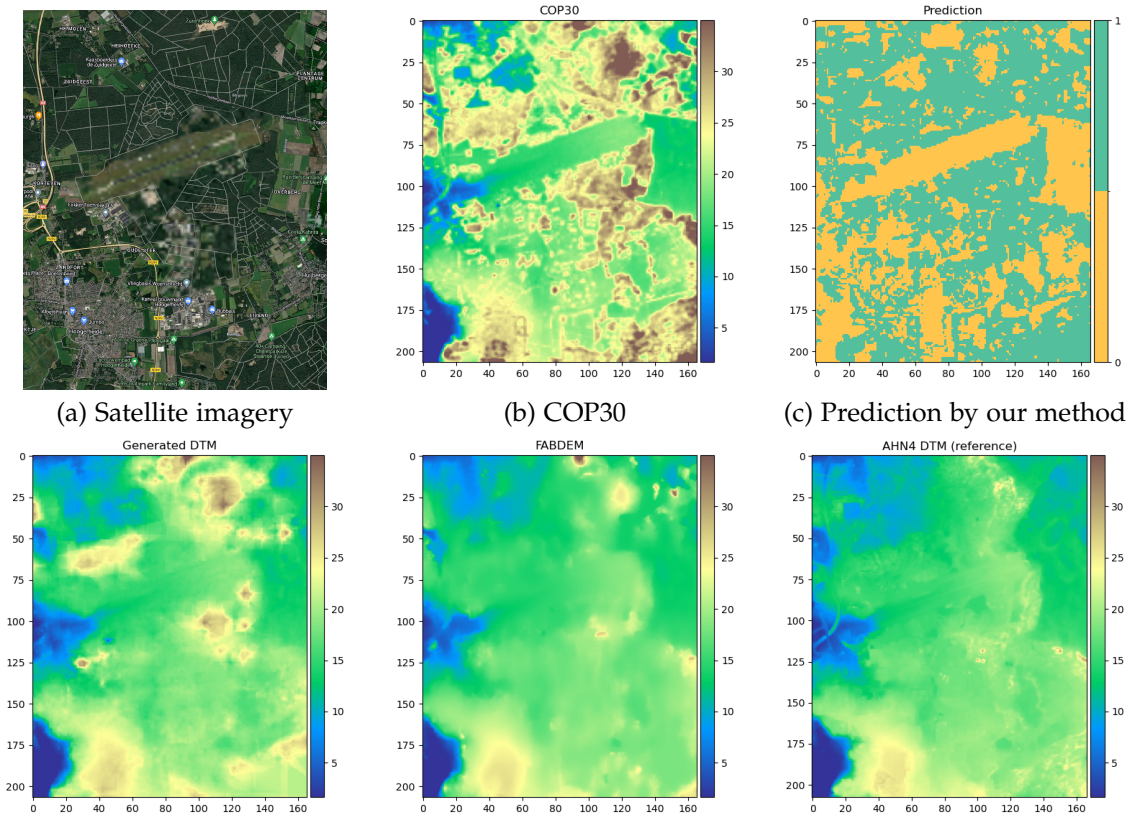


(a) Satellite imagery (b) COP30 (c) Prediction by the model

(d) generated DTM (e) FABDEM (f) AHN4 DTM (reference)

Figure 5.4.: Plots of the best performance area: (a) Satellite image, (b) COP30, (c) Prediction by the method, (d) generated DTM, (e) FABDEM, (f) AHN4 DTM (reference).

The observed variability in model performance is predictable, given that the model was not trained with COP30 data, leading to outcomes that are not as precise as those obtained with AHN4. To illustrate this, Figure 5.4 and Figure 5.5 present the satellite imagery, COP30 DSMs, model predictions, reference DTMs, generated DTMs, and FABDEM DTMs for re-



(a) Satellite imagery (b) COP30 (c) Prediction by our method

(d) generated DTM (e) FABDEM (f) AHN4 DTM (reference)

Figure 5.5.: Plots of the worst performance area: (a) Satellite image, (b) COP30, (c) Prediction by the model, (d) generated DTM, (e) FABDEM, (f) AHN4 DTM (reference). The blurred area in the satellite image is a military Airport in Rotterdam.

gions demonstrating the highest and lowest RMSE values, which underlines the disparity in model efficacy.

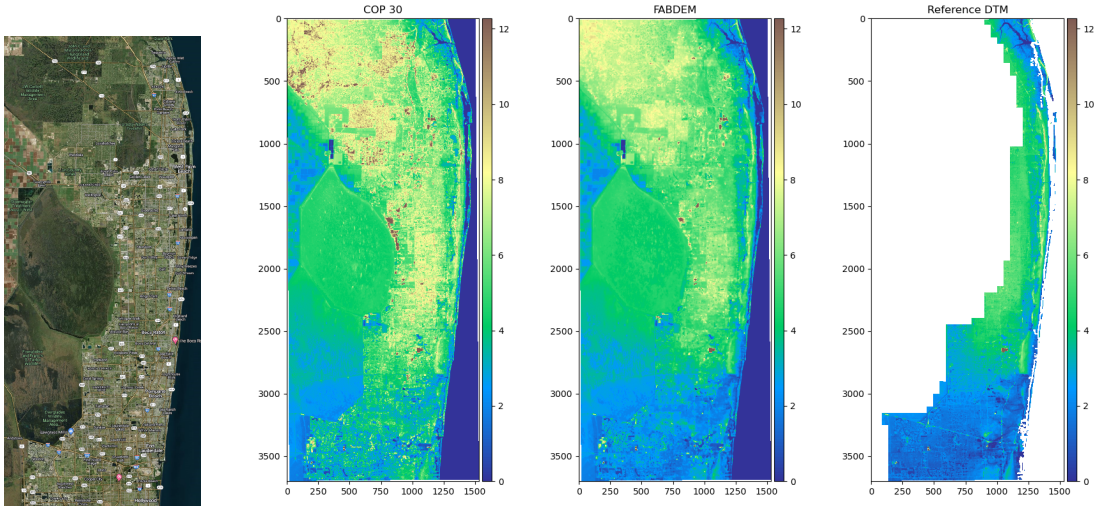
Upon analyzing the prediction images, the model exhibits a notable ability to distinguish between buildings, trees, and the underlying terrain within the COP30 dataset. However, in Figure 5.5, it becomes apparent that the model takes a relatively conservative approach when classifying the canopies in COP30. Rather than completely removing the entire forest, some pixels remain, resulting in artifacts resembling fake hills at the upper right corner and along the middle right edge. This phenomenon contributes to a significant increase in the Root Mean Square Error (RMSE) values.

Another observation is evident in Figures 5.4 and 5.5, where FABDEM displays distinct blurring effects. FABDEM selectively retains only certain "important" pixels with finer details, such as the small "hills" at the upper right corner in Figure 5.4 and along the upper edge in Figure 5.5. However, these features are also artifacts. Nonetheless, the area affected by these artifacts is relatively smaller compared to the method. This discrepancy primarily stems from the inherent differences in the respective methodologies. FABDEM directly addresses elevation bias (as discussed in Section 2.3.4), whereas the approach involves pixel

removal and interpolation based on neighboring values. Consequently, the method may inadvertently create larger hills when incomplete removal of trees from a forest occurs.

5.3. Evaluating Model Performance in Miami and Elevated Miami

To evaluate the generalization capability and transferability of the model, it is necessary to test its performance in an area outside of the Netherlands. Miami, with its similar elevation characteristics and a combination of sparse buildings and forests, serves as a suitable choice for this purpose. For a more comprehensive comparison with FABDEM, the corresponding area of Copernicus DEM for Miami is utilized as input data, while the reference DTM is sourced from the NOAA Sea Level Rise Viewer DEM dataset [3]. All the DEMs have been aligned and remapped to a local coordinate reference system (CRS) with a uniform resolution of 30 meters. Figure 5.6 presents the satellite image along with the respective DEMs.



(a) Satellite imagery (b) COP 30 (c) FABDEM (d) Reference DTM

Figure 5.6.: Plots of the Miami area used in the study: (a) Satellite image, (b) COP 30, (c) FABDEM, (d) Reference DTM.

To further assess the model's performance, an elevated version of Miami is created, where the elevation is artificially increased by 100 meters compared to the original city. The root mean square error (RMSE) of the generated DTMs by the model for both Miami and the elevated version, as well as the RMSE of FABDEM, are provided in Table 5.3.

The model achieved a remarkable reduction in the RMSE from 1.43 (COP30) to 0.74, demonstrating its ability to effectively extracting a DTM from a DSM. This performance is comparable to that of the FABDEM. Even when confronted with the elevated Miami area, the

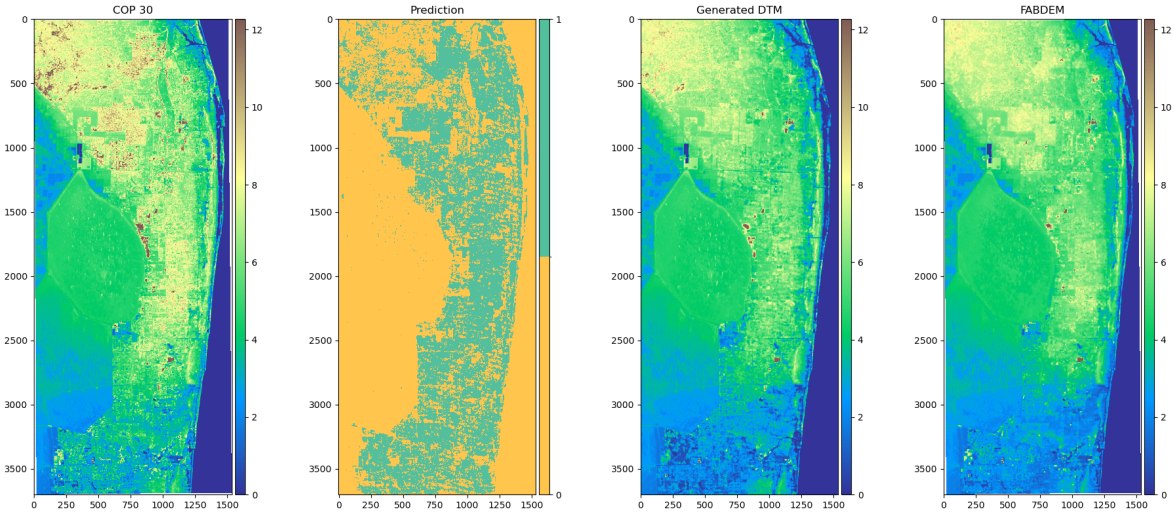
[3] https://coast.noaa.gov/htdata/raster2/elevation/SLR_viewer_DEM_6230/

	RMSE
Generated DTM of Miami	0.74
Generated DTM of elevated Miami	0.87
FABDEM	0.67
COP30	1.43

Table 5.3.: RMSE for Miami DEMs

model's performance only slightly diminishes, indicating its robustness and generalization capability in handling different terrain conditions.

Figure 5.7 and Figure 5.8 illustrate the prediction maps and the generated DTMs for both Miami and elevated Miami, enabling a visual comparison of the model's performance in the area.

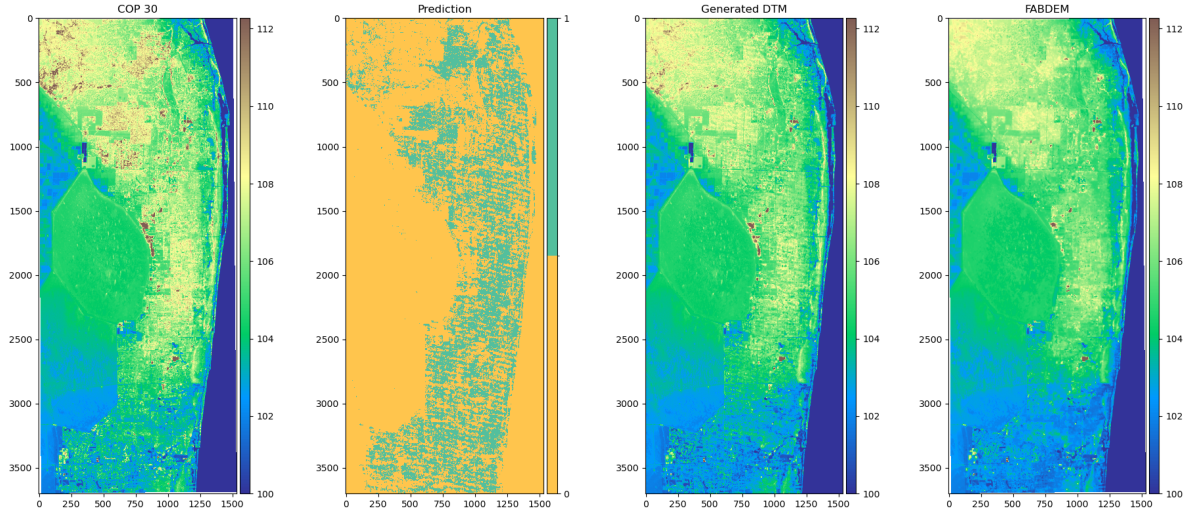


(a) COP30 (b) Prediction map (c) Generated DTM (d) FABDEM

Figure 5.7.: Plots of the Miami area used in the study: (a) COP30, (b) Prediction map by the model, (c) Generated DTM, (d) FABDEM.

When comparing the prediction map and generated DTM of Miami with the COP30 DSM and the FABDEM, it is evident that the model effectively eliminates a significant portion of pixels representing buildings, man-made structures, and canopies while retaining important pixels corresponding to hills and other terrain features. This retention of key pixels aligns with the observations made in the FABDEM, contributing to the similar RMSE values observed between the two. The generated DTM further showcases the model's exceptional generalization capability and transferability to areas with terrain characteristics similar to those found in the Netherlands.

When confronted with the elevated Miami area, the model exhibited misjudgments in numerous pixels that should have been classified as 1, as opposed to its performance in the original Miami scenario. Consequently, this led to the presence of artifacts and an increase in the RMSE value. These results indicate that the model is sensitive to variations in elevation



(a) elevated COP30 (b) Prediction map (c) Generated DTM (d) elevated FABDEM

Figure 5.8.: Plots of the Miami area used in the study: (a) elevated COP30, (b) Prediction map by the model, (c) Generated DTM, (d) elevated FABDEM.

and encounters difficulties when processing regions with higher elevations. Further investigation into this sensitivity and its impact on the model's performance will be discussed in the subsequent sections.

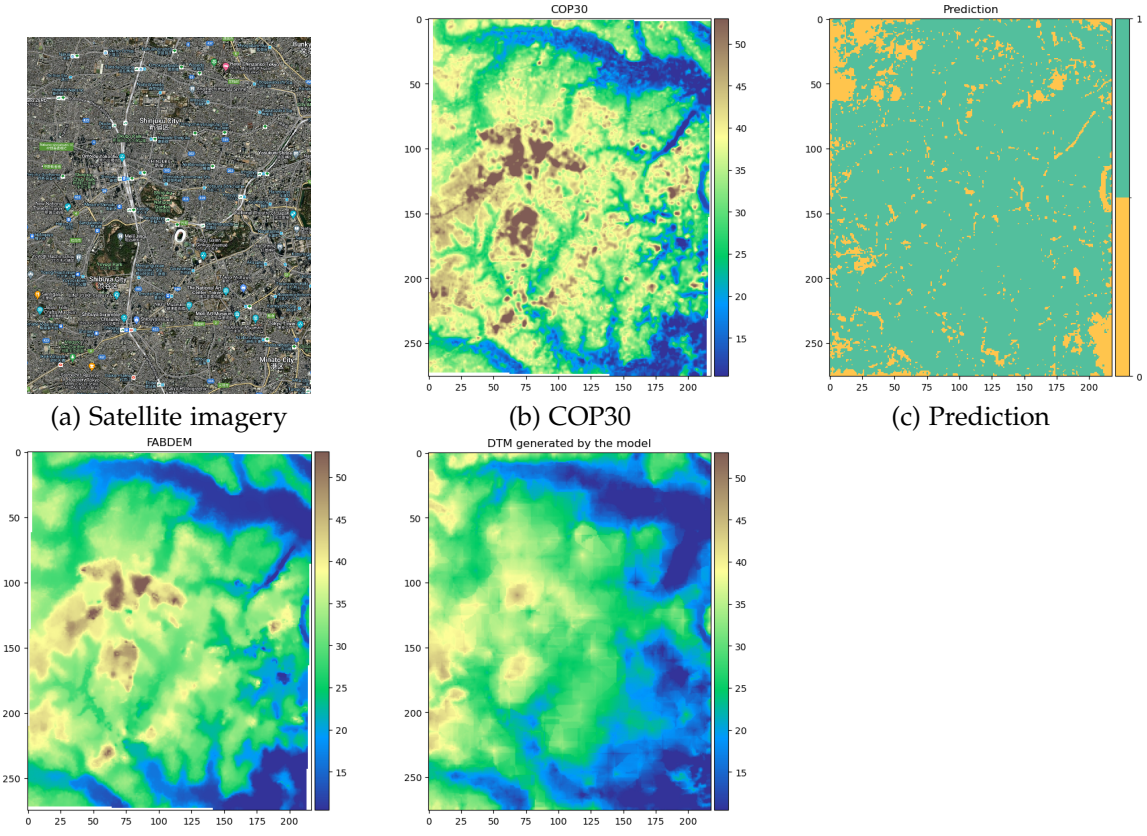
5.4. Visual Comparisons of Other Areas

To evaluate the model's transferability and further investigate the features captured by the model, I tested three distinct areas around the world: 1. Dense Urban Area (Shibuya and Shinjuku in Tokyo), 2. Vast Forested Region (Amazon Basin), and 3. Mountainous Area (Alps). All global DEMs were derived from COP30 and remapped to the local Coordinate Reference System (CRS). Please notice that the selected areas were initially cropped using the World Geodetic System 1984 (WGS84) Coordinate Reference System (CRS). However, when remapped into local CRSs, these areas experienced distortions that resulted in missing portions of the corners in the subsequent figures.

5.4.1. Dense Urban Area (Shibuya and Shinjuku in Tokyo)

Shibuya and Shinjuku in Tokyo are characterized by their dense urban landscapes, featuring numerous high-rise buildings, commercial areas, and bustling streets. Despite the urban density, these districts also contain parks and green spaces with trees, providing a mix of natural and built environments. The land surface in these areas is generally flat or gently sloping, with limited elevation changes. The combination of these features makes Shibuya and Shinjuku interesting case studies for evaluating the performance of elevation models in complex urban settings.

For the dense urban area of Shibuya and Shinjuku in Tokyo, Figure 5.9 showcases the satellite imagery, COP30, FABDEM, and the DTM generated from COP30 by the model, providing a detailed representation of the complex built environment and the model's performance in this challenging setting.



(a) Satellite imagery (b) COP30 (c) Prediction

(d) FABDEM (e) DTM generated from COP30

Figure 5.9.: Plots of the dense urban area (Shibuya and Shinjuku in Tokyo): (a) the satellite imagery, (b) COP30, (c)prediction, (d) FABDEM, (e) the DTM generated from COP30 by the model.

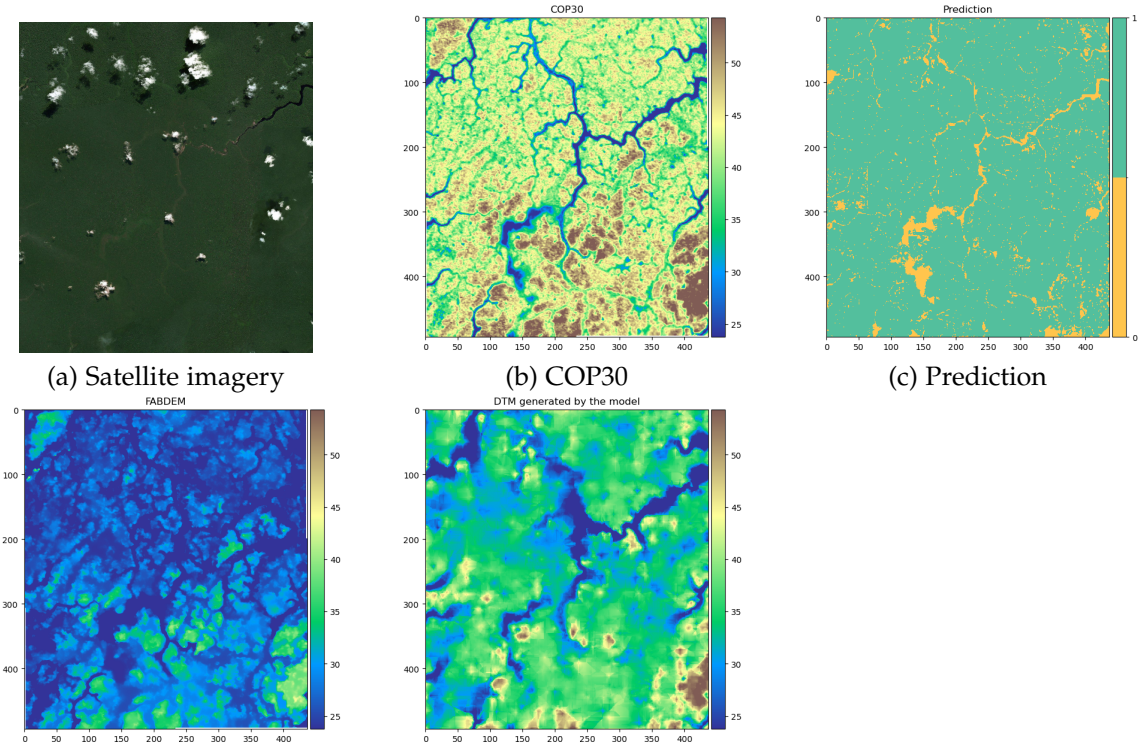
As depicted in the prediction map shown in Figure 5.9, the model correctly identified the majority of pixels as 1, as expected given that most of these pixels represent buildings. Additionally, the model successfully removed trees present in Yoyogi Park, located near the center of the selected area. However, it is noteworthy that the model misjudged the corners, primarily due to the missing corner portions mentioned earlier.

5.4.2. Vast Forested Region (Amazon Basin)

The Amazon Basin, located in South America, is a vast forested region characterized by its extensive tropical rainforests, diverse flora and fauna, and a dense network of rivers, including the mighty Amazon River. The terrain in the Amazon Basin is predominantly flat or gently undulating, with occasional areas of higher elevation, such as the Andean

foothills. The dense vegetation and complex hydrological system make the Amazon Basin a challenging area for assessing the performance of elevation models, particularly in terms of capturing the underlying terrain beneath the forest canopy.

For the vast forested region of the Amazon Basin, Figure 5.10 presents the satellite imagery, COP30, FABDEM, and the DTM generated from COP30 by the model, highlighting the intricate vegetation patterns and the effectiveness of the approach in accurately representing the terrain beneath the dense forest canopy.



(a) Satellite imagery (b) COP30 (c) Prediction

(d) FABDEM (e) DTM generated from COP30

Figure 5.10.: Plots of the vast forested region (Amazon Basin): (a) the satellite imagery, (b) COP30, (c) prediction, (d) FABDEM, (e) the DTM generated from COP30 by the model.

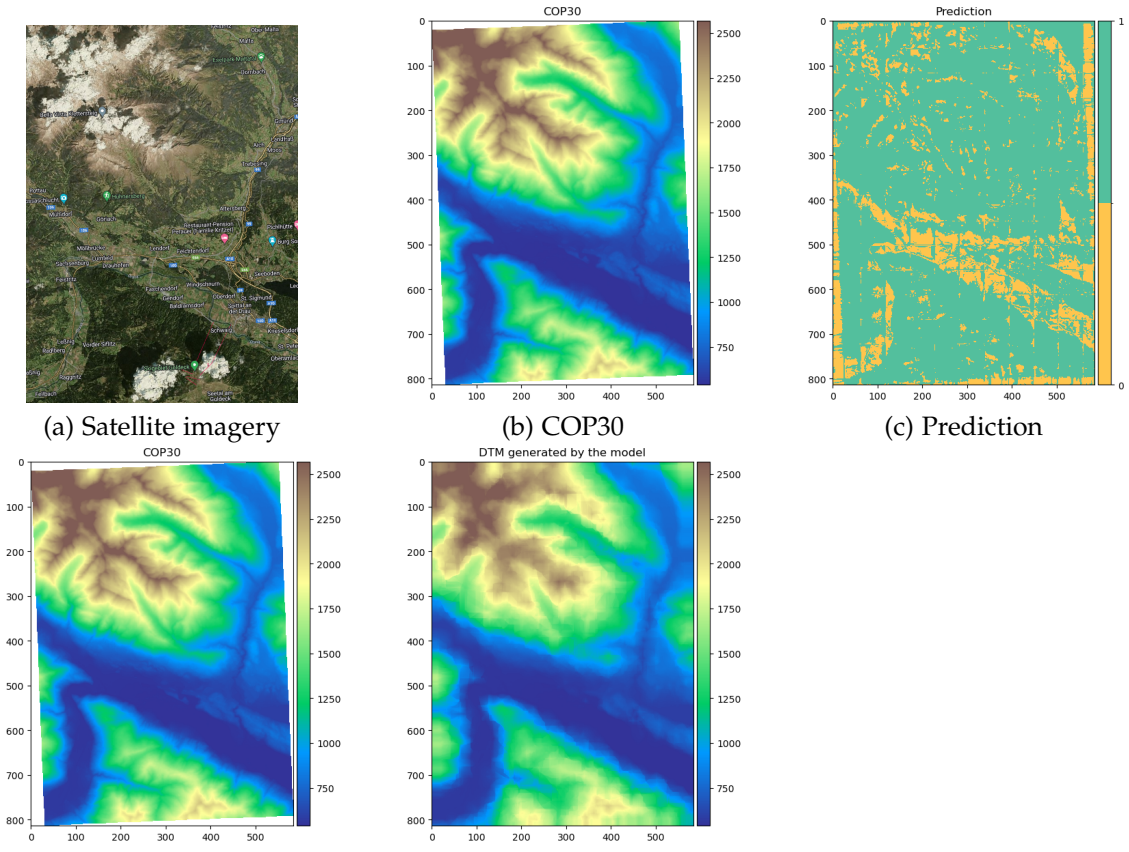
Examining the prediction map presented in Figure 5.10, the model demonstrates the ability to successfully identify a significant number of trees within the Amazon Basin. However, as discussed in Section 5.2, the presence of remaining pixels has a substantial impact. Consequently, the generated Digital Terrain Model (DTM) exhibits significant differences compared to FABDEM and contains higher elevation areas.

5.4.3. Mountainous Area (Alps in the western region of Austria)

The Alps, specifically the western region of Austria, are a mountainous area known for their rugged landscapes, steep slopes, and high elevation changes. The Alps consist of a mix of vegetation types, including forests, alpine meadows, and exposed rock or snow-covered peaks. In addition to the natural landscape, the region also features human settlements and

infrastructure, such as roads, railways, and mountain resorts. The complex topography and diverse land cover of the western Austrian Alps present a demanding test case for evaluating the accuracy and effectiveness of elevation models in capturing the intricate terrain variations.

For the mountainous area of the western Austrian Alps, Figure 5.11 illustrates the satellite imagery, COP30, prediction, FABDEM, and the DTM generated from COP30 by the model.



(a) Satellite imagery (b) COP30 (c) Prediction

(d) FABDEM (e) DTM generated from COP30

Figure 5.11.: Plots of the mountainous area (Alps in the western region of Austria): (a) the satellite imagery, (b) COP30, (c) prediction, (d) FABDEM, (e) the DTM generated from COP30 by the model.

Among the three selected areas, the Alps region stands out as the largest in terms of geographical extent and exhibits the widest range of elevations. Consequently, this area poses the most challenging task for the model. Upon closer examination, it is evident that the ridge section located in the upper left corner of the generated DTM appears significantly blurred in Figure 5.11. This observation suggests that the model encounters difficulties in accurately handling steep slopes like the one observed in this particular area.

5.5. Other Resolutions

Most global DEMs are available at resolutions of 30 or 90 meters (1 or 3 arc-seconds) (Hawker et al. [2018]). Some newer DEMs, such as the USGS 3DEP, have achieved a resolution of 10 meters (1/3 arc-second).

Figure 5.12 displays the DSMs and the DTMs of the same region at different resolutions.

In this section, I evaluate the performance of the model, which was trained using 30-meter resolution DSMs, when applied to DSMs with 10-meter and 90-meter resolutions. To achieve this, I resampled the 50 test patches to 10-meter and 90-meter resolutions (see Section 4.3) and subsequently computed the accuracy and the RMSE. The results are presented in Table 5.4.

It is important to note that the width of the 90-meter DSMs is smaller than the input size required by the model. Therefore, for this test, I have zero-padded their width to 64.

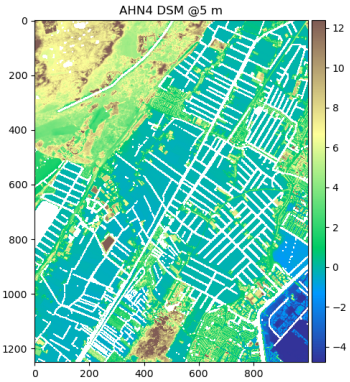
Input resolution	Accuracy	RMSE
30 m	94.1%	0.59
10 m	77.5%	2.16
90 m	68.8%	1.85

Table 5.4.: Mean accuracy and RMSE for different input resolutions

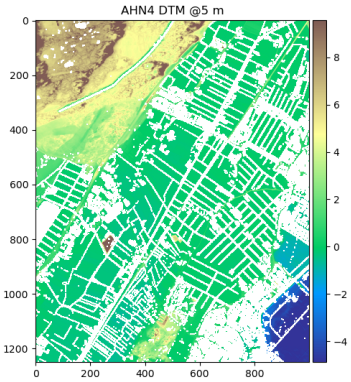
Upon examining Table 5.4, I observe that the results indicate that the model's accuracy varies depending on the input data resolution. As expected, the model performs best on the 30m resolution data. Both the 90 m resolution test and the 10 m resolution test exhibit inferior performance compared to the model's training resolution of 30 m. When applied to the higher-resolution 10m data, the model's accuracy might be slightly affected due to the increased level of detail in the input data. In contrast, the model's performance on the lower-resolution 90m data could be limited by the lack of detail and the reduced quality of the input data.

More specifically, in terms of accuracy, the 10 m test outperforms the 90 m test. I attribute this result to the higher level of detail preserved in the 10 m resolution, which assists the model in making more accurate judgments. Conversely, in terms of RMSE, the 90 m test demonstrates better performance. I attribute this outcome to the smoothing effect of the 90 m resolution, which reduces the number of pixels and consequently mitigates the impact of incorrectly classified pixels.

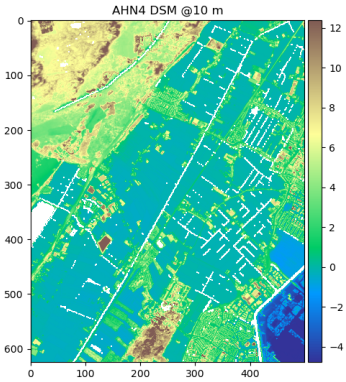
(a) Satellite Image



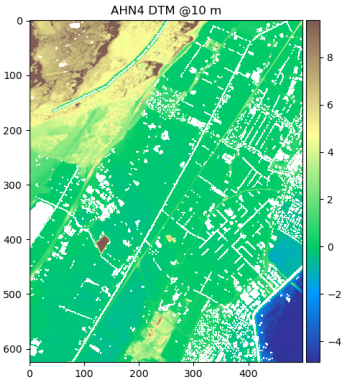
(b) AHN4 DSM at 5 m resolution



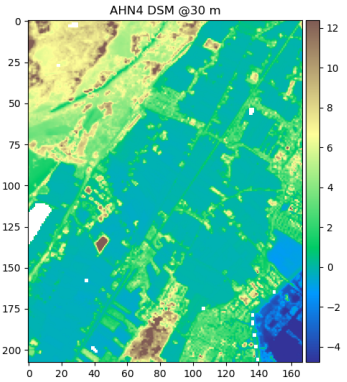
(c) AHN4 DTM at 5 m resolution



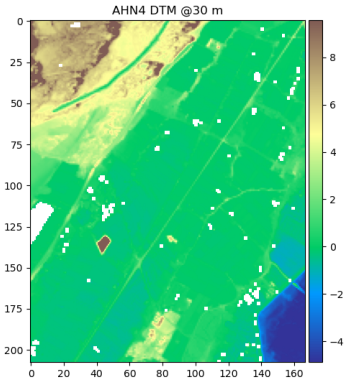
(d) resampled DSM (10 m)



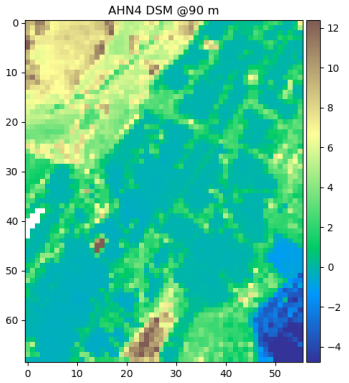
(e) resampled DTM (10 m)



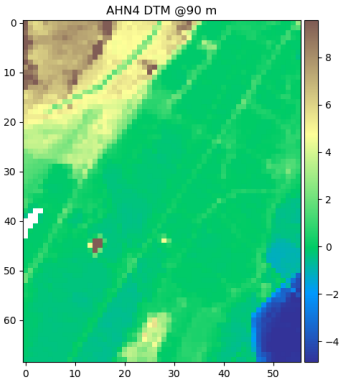
(f) resampled DSM (30 m)



(d) resampled DTM (30 m)



(e) resampled DSM (90 m)



(f) resampled DTM (90 m)

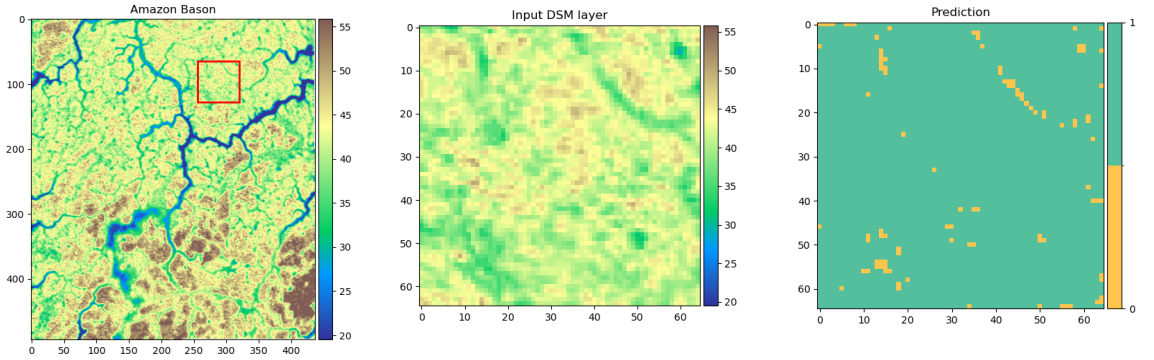
Figure 5.12.: DSMs and the DTMs at different resolutions.

6. DISCUSSION & CONCLUSION

6.1. Single Input Analysis with Fixed Window Size

In accordance with the methodology delineated in Section 3.2.2, the model received inputs that were sliced into a fixed shape of (64, 64, 3). A potential issue with this fixed window size approach may arise when the sliced area is predominantly occupied by a single class, such as infrastructure or canopy. In such instances, the model might face difficulties in distinguishing terrain pixels due to the lack of class variation.

Finding a square area with a side length of 64*30 m entirely filled with buildings or even covered by a single structure may prove to be challenging. Conversely, it is relatively straightforward to select a similarly sized area densely populated with trees, especially in regions like the Amazon Basin, as highlighted in Section 5.4.2. For illustrative purposes, I have selected a specific area (64:129, 256:321) from the Amazon Basin, as marked by the red border in subfigure (a) in Figure 6.1. This region provides an apt demonstration of the aforementioned case. Figure 6.1 exhibits the DSM of the selected region in the Amazon Basin, its corresponding sliced DSM, and the model's prediction for the input.

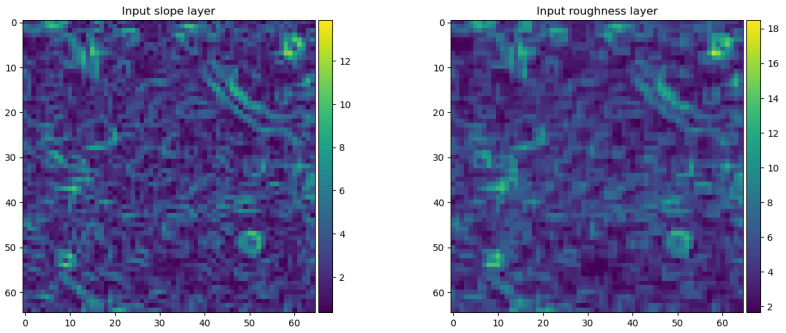


(a) A DSM of Amazon Basin (b) The DSM layer of an input (c) The prediction by the model

Figure 6.1.: An example of The DSM layer of an input of a DSM from Amazon Basin.

A close examination of the prediction in Figure 6.1 reveals that despite the input DSM being predominantly filled with a single class and exhibiting minimal elevation variation, the model successfully identifies the canopy, marking the majority of the pixels as 1. This performance could be attributed to the additional layers of input, namely the slope map and the roughness map, both of which are depicted in Figure 6.2.

Upon inspecting Figure 6.2, it becomes evident that the slope map and the roughness map convey more intricate details, potentially enabling the model to extract essential features and



(a) The slope layer of the input (b) The roughness layer of the input

Figure 6.2.: The other 2 layers of the input.

consequently classify the pixels accurately. The rationale behind choosing the slope map and the roughness map as the two additional layers is elucidated in Section 4.2.2. However, until now, there is no empirical evidence supporting their essentiality or usefulness to the model. The case presented here serves as a supportive argument, offering tangible proof of their value to the model's performance.

The analysis also reveals a scenario representing an extreme case where the vast majority of the pixels are removed, leaving only a sparse distribution of remaining pixels. This condition could potentially lead to the formation of artifacts, subsequently contributing to an elevated Root Mean Square Error (RMSE). It should be noted that these results are significantly influenced by the applied interpolation methods.

6.2. Limitations & Method Flaws

Despite its strengths, the Residual U-Net model also has some limitations. One of the primary challenges is the dependency on the quality and resolution of the input data. The model's performance can be affected when applied to datasets with different resolutions than the training data. Furthermore, the model may struggle in areas with complex terrain, dense vegetation, or a mix of both. Another potential limitation is the need for a large annotated dataset for training, which can be time-consuming and labour-intensive to create. Lastly, although the model performs well on the tested resolutions, it may not generalize well to other resolutions without additional fine-tuning or retraining.

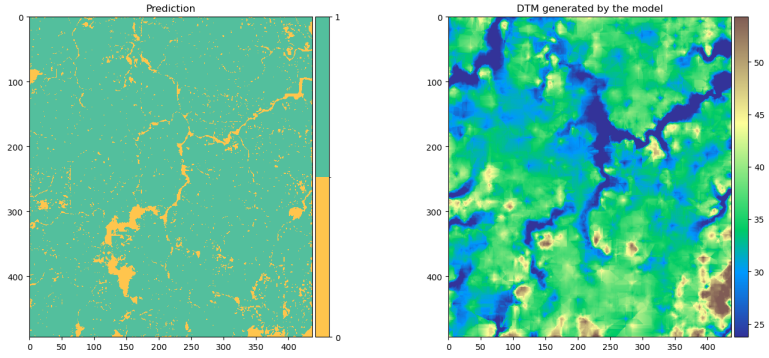
6.2.1. Artifacts

Artifacts, predominantly arising from the interpolation process, can be broadly categorized into two distinct scenarios:

1. Misjudged pixels situated within an input
2. Misjudged pixels located on the edge of an input

The interpolation methodology adopted in this study renders the resultant DTM particularly sensitive to unclassified or missing pixels. Consequently, if a set of pixels with high elevation are inaccurately predicted as terrain pixels while the surrounding pixels are correctly predicted as building or tree pixels, the latter will adopt the value of the former during interpolation. This could potentially lead to the formation of artificial hills, thereby resulting in a higher RMSE in the outcomes.

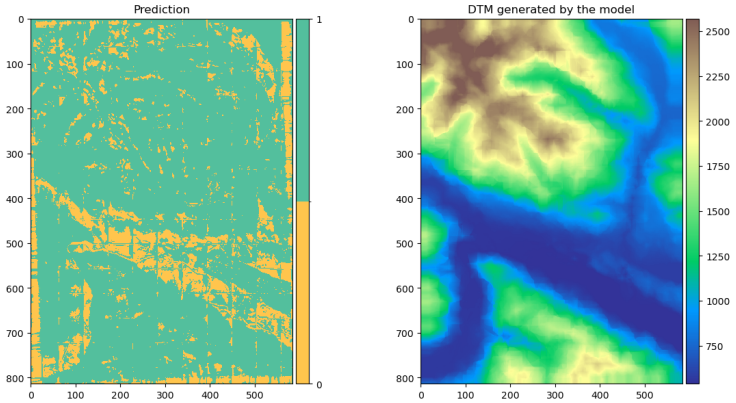
Figure 6.3 presents the prediction and final DTM for the Amazon Basin as an example.



(a) Prediction of the Amazon Basin (b) Generated DTM

Figure 6.3.: Plots of the Amazon Basin : (a) Prediction, (b) Generated DTM.

As discussed in Section 3.2.2, the model ingests inputs of the dimensions (64, 64, 3). I employ slicing on DSMs exceeding these dimensions, generating 64x64 sized pieces. The output mirrors these dimensions, and the individual pieces are stitched together to produce the final prediction. This procedure may induce interruptions and discontinuity, as demonstrated in Figure 6.4. This figure reveals the presence of discontinuity at the borders of the sliced pieces, leading to the creation of artificial hills in the final DTM of the Alpine region.



(a) Prediction of the Alps area (b) Generated DTM

Figure 6.4.: Plots of the Alps area : (a) Prediction, (b) Generated DTM.

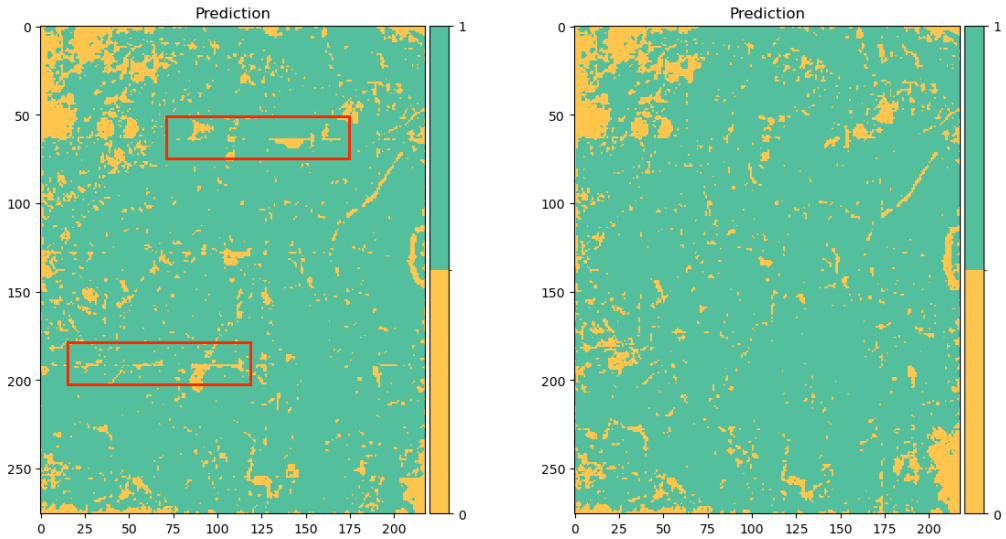
6.3. The Improvement of Slicing with Overlaps

Slicing with overlaps introduces the sharing of common information among neighbouring segments, resulting in smoother transitions and a reduction in potential artifacts during the reassembly process. In the experimental evaluation, although the observed improvement is relatively modest, with an increase in accuracy of 0.1% and a decrease in RMSE of 0.1 compared to the approach without overlaps, the inclusion of overlaps demonstrates its potential impact. Detailed results and statistics can be found in Table 6.1.

	Accuracy	**RMSE**
With overlaps	94.2%	0.59
Without overlaps	94.1%	0.60

Table 6.1.: Comparison of mean accuracy and RMSE for slicing method with and without overlaps

The limited difference in performance between the approaches may be attributed to the absence of significant conflicts or discrepancies along the edges of neighbouring segments. However, visual comparisons of prediction maps in different cases depicted in Figure 6.5 showcase the effectiveness of incorporating overlaps in the slicing method, as it noticeably reduces discontinuities and enhances the overall cohesiveness of the segmented image.



(a) Prediction when sliced without overlaps (b) Prediction when sliced with overlaps

Figure 6.5.: Comparison of prediction maps for the dense urban area when sliced with or without overlaps, highlighting the artifacts or errors caused by no-overlap slicing in (a) with red boxes.

As observed in the prediction map of the mountainous area (Figure 5.11 and Figure 6.4), it is evident that utilizing slicing with overlaps does not completely eliminate all conflicts or errors along the edges of the segments. One possible reason for this is that the fixed width of the overlap may not be suitable for all segments, resulting in conflicts in certain cases.

This indicates the need for adaptive overlap widths to better accommodate the variations in segment characteristics and reduce potential conflicts. Further research and experimentation are necessary to optimize the overlap width and ensure its effectiveness across different segments.

6.4. Input Sensitivity

As described in Section 4.2.3, the model is trained using normalized image-like arrays, and the mean and standard deviations are computed from the inputs sourced exclusively from AHN4. However, applying these normalization parameters to DSMs from other areas can potentially confuse the model and lead to undesirable results. This factor may also account for the suboptimal performance observed when handling the Alps area.

6.5. Future Work

Future work could focus on addressing the model's limitations by incorporating additional data sources, such as optical and radar imagery. Moreover, exploring different deep learning architectures and techniques to optimize the model's performance across various resolutions and input data types can further enhance its applicability in the geoscience domain. The results of this research serve as a solid foundation for the development of more advanced and versatile DSM-to-DTM conversion techniques in the future.

6.6. Conclusion

In this thesis, I have presented a novel approach for converting Digital Surface Models (DSMs) to Digital Terrain Models (DTMs) using a Residual U-Net deep learning model. The primary goal of the research was to develop a model capable of producing reliable DTMs from DSMs by effectively removing the pixels of trees and buildings. I trained the Residual U-Net model on the AHN4 dataset, which consists of high-resolution elevation data for the Netherlands, and demonstrated its effectiveness in generating accurate DTMs.

The results of the experiments show that the Residual U-Net model performs well in capturing relevant terrain features and removing above-ground objects from the DSMs. The model's adaptability to different resolutions, as evidenced by its performance on 10m, 30m, and 90m resolution datasets, indicates its potential in various geoscience applications. Moreover, the visual comparison with the Copernicus DEM (30m resolution) highlights the model's ability to generalize and perform well on different datasets.

Despite its strengths, the Residual U-Net model has some limitations, including dependency on input data quality and resolution, difficulty in handling complex terrains and dense vegetation, and challenges associated with creating large annotated datasets for training. Nevertheless, the model's overall performance and adaptability make it a promising solution for DTM extraction.

A. Reproducibility self-assessment

A.1. Marks for each of the criteria

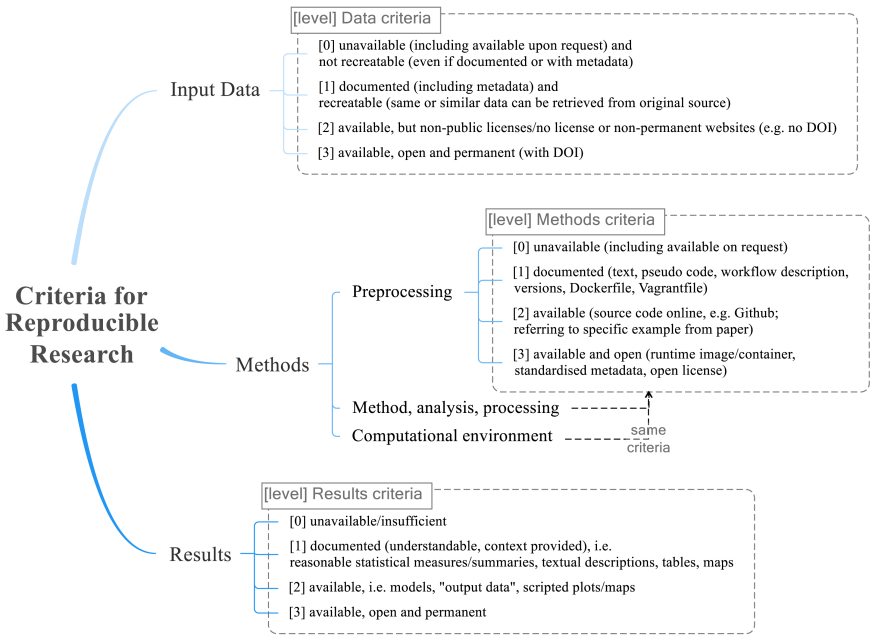


Figure A.1.: Reproducibility criteria to be assessed.

A.2. Self-reflection

1. Input data: **2**
   All the necessary data for the study have been collected and are readily available, although it should be noted that some data sources may not be permanently accessible. A comprehensive list of the data sources utilized in the study can be found in the appendices.
2. Preprocessing: **2**
   The source code for the preprocessing steps has been made available on a dedicated GitHub repository.
3. Methods: **2**
   The source code for the methods employed in the study has also been shared on the GitHub repository.

4. Computational environment: **2**
   The source code on the GitHub repository is compatible with various computational environments. The platform used in the study supports the execution of the code in almost all major environments, providing flexibility for researchers and practitioners to replicate and extend the study.
5. Results: **2**
   In order to facilitate the evaluation and verification of the study's results, the trained model file and selected outputs have been uploaded to the GitHub repository. This allows others to examine the results and compare them with their own findings.

B. Supporting Datasets and Links

B.1. Github Repository

https://github.com/AkiGoat/DSM2DTM-ResUNet.git

B.2. Datasets

1. FABDEM:
   https://data.bris.ac.uk/data/dataset/25wfy0f9ukoge2gs7a5mqpq2j7
2. Copernicus GLO-30 Digital Elevation Model:
   https://portal.opentopography.org/raster?opentopoID=OTSDEM.032021.4326.3
3. AHN4:
   https://www.arcgis.com/home/item.html?id=77da2e9eeea8427aab2ac83b79097b1a

Bibliography

Abdulqader, D., Mohsin Abdulazeez, A., and Zeebaree, D. (2020). Machine Learning Supervised Algorithms of Gene Selection: A Review.

AHN (2020a). Hoe werkt het inwinnen van hoogtegegevens. Publisher: AHN.

AHN (2020b). Kwaliteitsbeschrijving. Publisher: AHN.

AHN (2023). Producten. Publisher: AHN.

Amini Amirkolaee, H., Arefi, H., Ahmadlou, M., and Raikwar, V. (2022). DTM extraction from DSM using a multi-scale DTM fusion strategy based on deep learning. *Remote Sensing of Environment*, 274:113014.

Baugh, C. A., Bates, P. D., Schumann, G., and Trigg, M. A. (2013). SRTM vegetation removal and hydrodynamic modeling accuracy. *Water Resources Research*, 49(9):5276–5289. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wrcr.20412.

Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166. Conference Name: IEEE Transactions on Neural Networks.

Burrough, P. A. and McDonnell, R. A. (1986). Principles of geographical. *Information systems for land resource assessment. Clarendon Press, Oxford*.

Chen, Z., Wang, C., Li, J., Xie, N., Han, Y., and Du, J. (2021). Reconstruction Bias U-Net for Road Extraction From Optical Remote Sensing Images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:2284–2294. Conference Name: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27. Conference Name: IEEE Transactions on Information Theory.

Dang, K. B., Nguyen, M. H., Nguyen, D. A., Phan, T. T. H., Giang, T. L., Pham, H. H., Nguyen, T. N., Tran, T. T. V., and Bui, D. T. (2020). Coastal Wetland Classification with Deep U-Net Convolutional Networks and Sentinel-2 Imagery: A Case Study at the Tien Yen Estuary of Vietnam. *Remote Sensing*, 12(19):3270. Number: 19 Publisher: Multidisciplinary Digital Publishing Institute.

Darwish, A., Leukert, K., and Reinhardt, W. (2003). Image segmentation for the purpose of object-based classification. In *IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No.03CH37477)*, volume 3, pages 2039–2041.

Debella-Gilo, M. (2016). Bare-earth extraction and DTM generation from photogrammetric point clouds including the use of an existing lower-resolution DTM. *International Journal of Remote Sensing*, 37(13):3104–3124. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01431161.2016.1194543.

Diakogiannis, F. I., Waldner, F., Caccetta, P., and Wu, C. (2020). ResUNet-a: A deep learning framework for semantic segmentation of remotely sensed data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 162:94–114.

Donoho, D. L. and others (2000). High-dimensional data analysis: The curses and blessings of dimensionality. *AMS math challenges lecture*, 1(2000):32. Publisher: Citeseer.

Eberly, L. E. (2007). Multiple Linear Regression. In Ambrosius, W. T., editor, *Topics in Biostatistics*, Methods in Molecular Biology™, pages 165–187. Humana Press, Totowa, NJ.

Feng, W., Sui, H., Huang, W., Xu, C., and An, K. (2019). Water Body Extraction From Very High-Resolution Remote Sensing Imagery Using Deep U-Net and a Superpixel-Based Conditional Random Field Model. *IEEE Geoscience and Remote Sensing Letters*, 16(4):618–622. Conference Name: IEEE Geoscience and Remote Sensing Letters.

Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., and Garcia-Rodriguez, J. (2017). A Review on Deep Learning Techniques Applied to Semantic Segmentation. arXiv:1704.06857 [cs].

Gevaert, C. M., Persello, C., Nex, F., and Vosselman, G. (2018). A deep learning approach to DTM extraction from imagery using rule-based training labels. *ISPRS Journal of Photogrammetry and Remote Sensing*, 142:106–123.

Ghahramani, Z. (2004). Unsupervised Learning. In Bousquet, O., von Luxburg, U., and Rätsch, G., editors, *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2 - 14, 2003, Tübingen, Germany, August 4 - 16, 2003, Revised Lectures*, Lecture Notes in Computer Science, pages 72–112. Springer, Berlin, Heidelberg.

Ghorbanzadeh, O., Crivellari, A., Ghamisi, P., Shahabi, H., and Blaschke, T. (2021). A comprehensive transferability evaluation of U-Net and ResU-Net for landslide detection from Sentinel-2 data (case study areas from Taiwan, China, and Japan). *Scientific Reports*, 11(1):14629. Number: 1 Publisher: Nature Publishing Group.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256. JMLR Workshop and Conference Proceedings. ISSN: 1938-7228.

Gonga-Saholiariliva, N., Gunnell, Y., Petit, C., and Mering, C. (2011). Techniques for quantifying the accuracy of gridded elevation models and for mapping uncertainty in digital terrain analysis. *Progress in Physical Geography: Earth and Environment*, 35(6):739–764. Publisher: SAGE Publications Ltd.

Hawker, L., Bates, P., Neal, J., and Rougier, J. (2018). Perspectives on Digital Elevation Model (DEM) Simulation for Flood Modeling in the Absence of a High-Accuracy Open Access Global DEM. *Frontiers in Earth Science*, 6:233.

Hawker, L., Uhe, P., Paulo, L., Sosa, J., Savage, J., Sampson, C., and Neal, J. (2022). A 30 m global map of elevation with forests and buildings removed. *Environmental Research Letters*, 17(2):024016. Publisher: IOP Publishing.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep Residual Learning for Image Recognition. arXiv:1512.03385 [cs].

Hingee, K., Caccetta, P., Caccetta, L., Wu, X., and Devereaux, D. (2016). Digital terrain from a two-step segmentation and outlier-based algorithm. volume 41, pages 233–239. ISSN: 1682-1750.

Hu, X., An, X., and Li, L. (2016). Easy synthesis of highly fluorescent carbon dots from albumin and their photoluminescent mechanism and biological imaging applications. *Materials Science and Engineering: C*, 58:730–736.

Ioffe, S. and Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 448–456. PMLR. ISSN: 1938-7228.

James, M. R. and Robson, S. (2012). Straightforward reconstruction of 3D surfaces and topography with a camera: Accuracy and geoscience application. *Journal of Geophysical Research: Earth Surface*, 117(F3). _eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2011JF002289.

Jayadeva, Khemchandani, R., and Chandra, S. (2007). Twin Support Vector Machines for Pattern Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):905–910. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

Ji, S., Yu, D., Shen, C., Li, W., and Xu, Q. (2020). Landslide detection from an open satellite imagery and digital elevation model dataset using attention boosted convolutional neural networks. *Landslides*, 17(6):1337–1352.

Kazimi, B., Thiemann, F., and Sester, M. (2020). DETECTION OF TERRAIN STRUCTURES IN AIRBORNE LASER SCANNING DATA USING DEEP LEARNING. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume V-2-2020, pages 493–500. Copernicus GmbH. ISSN: 2194-9042.

Kingma, D. P. and Ba, J. (2017). Adam: A Method for Stochastic Optimization. arXiv:1412.6980 [cs].

Kulp, S. and Strauss, B. H. (2016). Global DEM Errors Underpredict Coastal Vulnerability to Sea Level Rise and Flooding. *Frontiers in Earth Science*, 4.

Kulp, S. A. and Strauss, B. H. (2018). CoastalDEM: A global coastal digital elevation model improved from SRTM using a neural network. *Remote Sensing of Environment*, 206:231–239.

Köppen, M. (2000). The curse of dimensionality. In *5th online world conference on soft computing in industrial applications (WSC5)*, volume 1, pages 4–8.

Lakshmi, S. E. and Yarrakula, K. (2018). Review and critical analysis on digital elevation models. *G e o f i z i k a*, 35(2):129–157. Number: 2.

Langford, J. and Schapire, R. (2005). Tutorial on practical prediction theory for classification. *Journal of machine learning research*, 6(3).

Lary, D. J., Alavi, A. H., Gandomi, A. H., and Walker, A. L. (2016). Machine learning in geosciences and remote sensing. *Geoscience Frontiers*, 7(1):3–10.

Li, L., Wang, C., Zhang, H., Zhang, B., and Wu, F. (2019). Urban Building Change Detection in SAR Images Using Combined Differential Image and Residual U-Net Network. *Remote Sensing*, 11(9):1091. Number: 9 Publisher: Multidisciplinary Digital Publishing Institute.

Lim, H.-I. (2019). A Linear Regression Approach to Modeling Software Characteristics for Classifying Similar Software. In *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, volume 1, pages 942–943. ISSN: 0730-3157.

Liu, H., Liu, Z., Liu, S., Liu, Y., Bin, J., Shi, F., and Dong, H. (2019). A Nonlinear Regression Application via Machine Learning Techniques for Geomagnetic Data Reconstruction Processing. *IEEE Transactions on Geoscience and Remote Sensing*, 57(1):128–140. Conference Name: IEEE Transactions on Geoscience and Remote Sensing.

Liu, X. (2008). Airborne LiDAR for DEM generation: some critical issues. *Progress in Physical Geography: Earth and Environment*, 32(1):31–49.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation. pages 3431–3440.

Maimon, O. and Rokach, L. (2005). Introduction to Supervised Methods. In Maimon, O. and Rokach, L., editors, *Data Mining and Knowledge Discovery Handbook*, pages 149–164. Springer US, Boston, MA.

Maulud, D. and Abdulazeez, A. M. (2020). A Review on Linear Regression Comprehensive in Machine Learning. *Journal of Applied Science and Technology Trends*, 1(4):140–147. Number: 4.

Meadows, M. and Wilson, M. (2021). A Comparison of Machine Learning Approaches to Improve Free Topography Data for Flood Modelling. *Remote Sensing*, 13(2):275. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

Mesa-Mingorance, J. L. and Ariza-López, F. J. (2020). Accuracy Assessment of Digital Elevation Models (DEMs): A Critical Review of Practices of the Past Three Decades. *Remote Sensing*, 12(16):2630. Number: 16 Publisher: Multidisciplinary Digital Publishing Institute.

Milletari, F., Navab, N., and Ahmadi, S.-A. (2016). V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571.

Okolie, C. J. and Smit, J. L. (2022). A systematic review and meta-analysis of Digital elevation model (DEM) fusion: pre-processing, methods and applications. *ISPRS Journal of Photogrammetry and Remote Sensing*, 188:1–29.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Poggio, T., Mhaskar, H., Rosasco, L., Miranda, B., and Liao, Q. (2017). Why and when can deep-but not shallow-networks avoid the curse of dimensionality: A review. *International Journal of Automation and Computing*, 14(5):503–519.

Polidori, L. and El Hage, M. (2020). Digital Elevation Model Quality Assessment Methods: A Critical Review. *Remote Sensing*, 12(21):3522. Number: 21 Publisher: Multidisciplinary Digital Publishing Institute.

Prasad, A. M., Iverson, L. R., and Liaw, A. (2006). Newer Classification and Regression Tree Techniques: Bagging and Random Forests for Ecological Prediction. *Ecosystems*, 9(2):181–199.

Ray, S. (2019). A Quick Review of Machine Learning Algorithms. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, pages 35–39.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. In Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F., editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, Lecture Notes in Computer Science, pages 234–241, Cham. Springer International Publishing.

Saleem, N., Huq, M. E., Twumasi, N. Y. D., Javed, A., and Sajjad, A. (2019). Parameters Derived from and/or Used with Digital Elevation Models (DEMs) for Landslide Susceptibility Mapping and Landslide Risk Assessment: A Review. *ISPRS International Journal of Geo-Information*, 8(12):545. Number: 12 Publisher: Multidisciplinary Digital Publishing Institute.

Segal, M. R. (2004). Machine Learning Benchmarks and Random Forest Regression.

Shamsolmoali, P., Zareapoor, M., Wang, R., Zhou, H., and Yang, J. (2019). A Novel Deep Structure U-Net for Sea-Land Segmentation in Remote Sensing Images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(9):3219–3232. Conference Name: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

Shehab, L. H., Fahmy, O. M., Gasser, S. M., and El-Mahallawy, M. S. (2021). An efficient brain tumor image segmentation based on deep residual networks (ResNets). *Journal of King Saud University - Engineering Sciences*, 33(6):404–412.

Siddique, N., Sidike, P., Elkin, C., and Devabhaktuni, V. (2021). U-Net and its variants for medical image segmentation: theory and applications. *IEEE Access*, 9:82031–82057. arXiv:2011.01118 [cs, eess].

Smith, M. W. (2014). Roughness in the Earth Sciences. *Earth-Science Reviews*, 136:202–225.

Soofi, A. and Awan, A. (2017). Classification Techniques in Machine Learning: Applications and Issues. *Journal of Basic & Applied Sciences*, 13:459–465.

Speiser, J. L., Miller, M. E., Tooze, J., and Ip, E. (2019). A comparison of random forest variable selection methods for classification prediction modeling. *Expert Systems with Applications*, 134:93–101.

Su, Y., Guo, Q., Ma, Q., and Li, W. (2015). SRTM DEM Correction in Vegetated Mountain Areas through the Integration of Spaceborne LiDAR, Airborne LiDAR, and Optical Imagery. *Remote Sensing*, 7(9):11202–11225. Number: 9 Publisher: Multidisciplinary Digital Publishing Institute.

Tomljenovic, I., Tiede, D., and Blaschke, T. (2016). A building extraction approach for Airborne Laser Scanner data utilizing the Object Based Image Analysis paradigm. *International Journal of Applied Earth Observation and Geoinformation*, 52:137–148.

Twa, M. D., Parthasarathy, S., Roberts, C., Mahmoud, A. M., Raasch, T. W., and Bullimore, M. A. (2005). Automated Decision Tree Classification of Corneal Shape. *Optometry and Vision Science*, 82(12):1038.

van Zyl, J. J. (2001). The Shuttle Radar Topography Mission (SRTM): a breakthrough in remote sensing of topography. *Acta Astronautica*, 48(5):559–565.

Wang, H., Ma, C., and Zhou, L. (2009). A Brief Review of Machine Learning and Its Application. In *2009 International Conference on Information Engineering and Computer Science*, pages 1–4. ISSN: 2156-7387.

Wilson, M. F. J., O'Connell, B., Brown, C., Guinan, J. C., and Grehan, A. J. (2007). Multiscale Terrain Analysis of Multibeam Bathymetry Data for Habitat Mapping on the Continental Slope. *Marine Geodesy*, 30(1-2):3–35. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01490410701295962.

Xu, M., Watanachaturaporn, P., Varshney, P. K., and Arora, M. K. (2005). Decision tree regression for soft classification of remote sensing data. *Remote Sensing of Environment*, 97(3):322–336.

Zhang, Z., Liu, Q., and Wang, Y. (2018). Road Extraction by Deep Residual U-Net. *IEEE Geoscience and Remote Sensing Letters*, 15(5):749–753. Conference Name: IEEE Geoscience and Remote Sensing Letters.

Zhou, Y., Wang, H., Yang, R., Xie, D., and Liu, J. (2022). Semantic Segmentation Algorithm of Landslide Based on Remote Sensing Image and DEM. In *2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI)*, pages 595–599.

Zou, K. H., Tuncali, K., and Silverman, S. G. (2003). Correlation and Simple Linear Regression. *Radiology*, 227(3):617–628. Publisher: Radiological Society of North America.

Colophon

This document was typeset using LaTeX, using the KOMA-Script class `scrbook`. The main font is Palatino.