U-Net Based Deep Learning Approach for Land Classification in Aerial Imagery

Iman Yavari

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science
in
Information Technology

Eastern Mediterranean University
August 2023
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Prof. Dr. Ali Hakan Ulusoy
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science in Information Technology.

Asst. Prof. Dr. Ece Çelik
Director, School of Computing and Technology

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Information Technology.

Prof. Dr. Ahmet Rizaner
Supervisor

Examining Committee

1. Prof. Dr. Ahmet Rizaner

2. Prof. Dr. Ali Hakan Ulusoy

3. Assoc. Prof. Dr. Kamil Yurtkan

ABSTRACT

With the invention of aerial imagery and satellite photography, reading the content of the images and having a better and easier understanding of their analysis has been challenging. Although high-quality aerial images include all the details in the image, how can scientists be sure of what they observe within the image? Also, in some cases where the images have low quality, the weather of the region is cloudy, or even the region is covered (lakes covered by plants, roads covered by trees, and so on), and in some other similar challenging cases, we have difficulty understanding the content of the images easily.

Considering this problem, we are looking for an approach that uses neural networks to analyse and read the content of aerial images and more precisely detect the type of land within the images. By utilizing convolutional neural networks with the U-Net model, we propose an automated and reliable solution for perceiving land types in aerial imagery. We evaluated the performance of our system using performance metrics such as accuracy, precision, and F1 scores for each land type. The results showed that our system achieved high accuracy and precision for specific land types. We believe that our system can help those who need to analyse aerial imagery better understand the content of the images and make more informed decisions.

Keywords: land type classification, unet classification, land type detection

ÖZ

Hava görüntüleri ve uydu fotoğrafçılığının icadıyla, görüntülerin içeriğini okumak ve analiz etmek daha zor hale gelmiştir. Yüksek kaliteli hava görüntüleri görüntüdeki tüm detayları içermesine rağmen, bilim adamları görüntünün içinde ne gözlemlediklerinden emin olamamaktadırlar. Ayrıca, görüntüler düşük kalitede olduğunda, bölgenin havası bulutlu olduğunda veya bölge kaplı olduğunda (bitkilerle kaplı göller, ağaçlarla kaplı yollar vb.) görüntülerin içeriğini kolayca anlamak zordur.

Bu sorunu göz önünde bulundurarak, hava görüntülerinin içeriğini analiz etmek, okumak ve görüntülerin içindeki arazi türünü daha kesin olarak tespit etmek için sinir ağları yardımıyla bir yaklaşım arıyoruz. Bu çalışmada, U-Net modeli ile evrişimli sinir ağlarını kullanarak, havadan görüntülerde arazi tiplerini algılamak için otomatik ve güvenilir bir çözüm önerilmektedir. Sistemimizin performansını, her arazi tipi için doğruluk, hassasiyet ve F1 puanları gibi performans metrikleri kullanarak değerlendirdik. Sonuçlar, sistemimizin belirli arazi tipleri için yüksek doğruluk ve hassasiyet elde ettiğini gösterdi. Bu çalışmada önerdiğimiz sistemimizin, hava görüntülerini analiz etmesi gereken kişilerin görüntülerin içeriğini daha iyi anlamasına ve daha bilinçli kararlar vermesine yardımcı olabileceğine inanıyoruz.

Anahtar Kelimeler: arazi tipi sınıflandırması, unet sınıflandırması, arazi tipi tespiti

ACKNOWLEDGMENT

I would like to express my gratitude to my supervisor, Prof. Dr. Ahmet Rizaner, for granting me the opportunity to work under his supervision and for his unwavering support throughout the process of completing my thesis. The members of my dissertation committee have also played a significant role in guiding me and imparting valuable knowledge about scientific research. Furthermore, I would like to extend my special thanks to my mentors and teachers from my master's program, who have taught me invaluable lessons that I cannot fully acknowledge here. Their patience, motivation, and enthusiasm have been truly remarkable, and I am deeply grateful for their contribution to creating a conducive environment for my thesis work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ADAM	Adaptive Moment Estimation
AI	Artificial Intelligence
CIoU	Complete Intersection over Union
CNN	Convolutional Neural Network
FN	False Negative
FP	False Positive
GB	Giga Byte
GPU	Graphical Processor Unit
IoU	Intersection over Union
LiDAR	Light Detection And Ranging
PPV	Positive Predictive Value
R-CNN	Region-based Convolutional Neural Networks
ReLU	Rectified Linear Unit
ROI	Region Of Interest
TN	True Negative
TP	True Positive
UAV	Unmanned Aerial Vehicle
YOLO	You Look Only Once

Chapter 1

INTRODUCTION

Aerial imagery is a valuable tool for understanding the landscape, but it can be challenging to analyse manually. This chapter discusses the challenges of automating and analysing aerial images for land type detection and introduces the research questions and aims of this thesis.

1.1 Aerial Imagery and Detecting the Type of Land Inside the Image

Nowadays, thanks to the latest technological advances in aerial photography and satellite imagery, we can access more quality images. However, one of the challenges with these pictures is analysing them. For example, consider receiving high-resolution aerial images from satellites or drones to determine how many residential areas are included in this region or to detect water resources or wetlands in a specific area before making a decision about the location. How the images are analysed is important. The traditional approach would require some human resources to manually analyse the data, which would take time and almost certainly result in numerous errors. As scientists or experts who like a fast and proper tool to make their own decisions, we believe that this might be handy. Additionally, managers, the government, and politicians will be able to analyse an aerial image without spending too much time or hiring too many experts.

1.2 Challenges in Automating and Analysing Aerial Images for Land Type Detection

In this proposal, we are looking for a reliable and precise method for detecting the type of land in aerial or satellite images. Since we now have easy access to Unmanned Aerial Vehicles (UAV), we can access aerial images much easily than before. We are looking for a deep-learning system that is able to detect forests, roads, rivers, rocks, fields, and buildings. In addition to this, going deeper into detail would be another problem, such as detecting the type of field or jungle. Additionally, considering the fact we have so many various types of elements in nature (for example, not always the water in nature is blue), it might be highly challenging to have a precise output. Regarding the mentioned problem, these sub-problems also affect the process:

- Quality and small ratio of image [1]
- Weather (the time images have been captured) [2]
- Various shapes and characteristics of the elements (which make them difficult to analyse)
- Similar elements (like yards, gardens, jungles, etc.)
- Detecting covered lands (like caves or a lake covered by plants, which can be considered green lands)

To address this challenge, we will use neural networks for training a network, image processing tools for processing the images, and finally, a reporting tool to export a final report or analysis of testing data.

1.3 Research Questions

Based on the problem explained above, we will be looking to answer the following

questions:

RQ1. How can we have a machine learning approach that receives the aerial images and exports well-organised analysis?

RQ2. How to design a neural network to receive aerial images as inputs for detecting the type of land?

RQ3. How to design a neural network to interpret land analysis?

RQ4. How is it possible for the system to detect the land types and usages and output the results quickly and precisely?

RQ5. How effective is our neural network compared to other traditional land-type analysis methods?

RQ6. How reliable is the result of the proposed method? How will the results be tested or evaluated?

1.4 Definition of Important Terms

We define some of the important terms in this section. Their details will be discussed in the following chapters.

UAV: It's the abbreviation of Unmanned Aerial Vehicles which means all types of aerial vehicles that can be controlled remotely. Drones and quad pods are two examples of popular aerial vehicles we use nowadays to provide quality aerial images.

CNN: Convolutional Neural Networks are types of architectures in neural networks that are able to learn directly from data. They are mostly used for finding patterns in images, objects in images, image classification or segmentation, and audio classification.

Noise: We use the term noise wherever some unwanted additional elements affect the network. Mostly these effects will negatively alter the expected results.

U-Net: The U-Net is a convolutional neural network model, originally designed for biomedical image segmentation. The U-Net can leverage a symmetric expanding path to precisely localize and be trained with less training data, enabling efficient pixel-wise classification.

1.5 Aims and Challenges

The proposed system can be a handy and efficient tool for scientists and experts to make their own decisions, as it provides a fast and reliable way to detect the type of land in aerial images. Imagine that managers, the government, or politicians will be able to analyze an aerial image without spending too much time and effort, hiring too many human resources, or even having the real-time analysis by a drone.

Our research focuses on developing a method to detect the type of land in aerial images. The proposed method will receive quality images in normal weather conditions (a sunny day) and detect the type of land. This work will not cover details about lands, such as which type of jungle or building. Cloudy weather, low-quality images, unusual element types, and oblique images taken by drones are examples of limitations that will affect our system. The aims and challenges of this research are also listed below as a summary.

1.5.1 Aims

Regarding mentioned research questions, in this thesis our aims of study include the below goals:

- To develop a method to detect the type of land in aerial images.

- To provide a fast and reliable tool for scientists and experts to make their own decisions.
- To enable managers, the government, or politicians to analyze an aerial image without spending too much time and effort, hiring too many human resources, or even having the real-time analysis of a drone.

1.5.2 Challenges

Considering the above mentioned aims of this study, to achieve our goals we experienced some challenges and obstacles which is listed as below in this section:

- Cloudy weather
- Low-quality images
- Unusual element types
- Oblique images taken by drones.

1.6 Organization of The Thesis

The thesis is organized into five chapters. Chapter 1 introduces the topic of aerial imagery analysis and land type detection and discusses the challenges of automating this process. Chapter 2 provides an overview of previous works and research related to aerial imagery analysis and object detection. Chapter 3 provides an overview of the concepts and technologies that will be used in this thesis, including aerial imagery and different machine learning models. Chapter 4 explains the methodology used to ensure the reliable performance of the proposed network, and discusses the progress made to train and test the neural network for classifying land types in aerial images. Chapter 5 summarizes the findings of the thesis and discusses the implications for future research.

Chapter 2

LITERATURE REVIEW

This chapter provides an overview of previous works and research related to aerial imagery analysis and object detection. We will discuss the current solutions and approaches in the fields of CNNs, image processing, object detection and aerial image segmentation.

2.1 Review of Previous Works

The history of image processing started in the 1960s at the Massachusetts Institute of Technology, University of Maryland [3]. The aim of the research was to improve the quality of satellite images to obtain a clearer vision of the people in the images. With the foundation of machine learning algorithms and deep learning techniques like neural networks, scientists and researchers started to detect various objects in aerial images, and they proposed lots of techniques in this regard. In this section, we will discuss and possibly use some of the proposed technologies and methods. The first step for most image processing systems is to optimise the quality of the image and then split it into smaller units so the network can analyse and process it. In this regard, sometimes the quality and ratio of input images are not satisfying due to common reasons, including low-quality cameras on drones or cloudy weather conditions, which lead to importing noise into the network. Maesako and Zhang [4] evaluated obstacles during the object recognition process, such as a small ratio of objects in the image. They proposed a method that increases the ratio of objects and evaluating them again. They proved their method achieved better performance and precision compared to

other methods. When dealing with small objects in UAV-based aerial images, models such as You Look Only Once [5,6] (YOLO v5) object detection algorithm and YOLOv4 outperform other algorithms. They noticed that considering the perspective of positioning accuracy and detection rate, the YOLOv4 model is slightly higher than YOLOv3 and much higher than YOLOv3 Tiny which is the light and more simplified type of YOLOv3 in terms of convolution layers in detection rate. Meanwhile, because YOLOv4 uses Complete Intersection over Union (CIoU) as the object detection regression loss function, compared with the Intersection over Union (IoU) loss function used by YOLOv3. YOLOv4 has better performance in bounding box position than YOLOv3, but there is also object omission detection. The authors also found that YOLOv3 was slightly better than YOLOv4 in the detection of incomplete samples, possibly due to the underfitting of YOLOv4. Secondly, most of the existing systems have to adapt to the characteristics of similar objects in aerial images, like an urban green area and a forest, which both have a lot in common in terms of their physical characteristics. Kopečný and Hnidka [7] proposed a real-time system with the help of data classification based on the histogram available in the new products and the data processing of neural networks. They realised the intensity of colours in the histograms of the urban and forest samples were too different. So, it was too easy to separate. In the Gabor filtering stage, the network showed errors [7]. The performance of the Gabor filter varied based on the chosen sample. Based on their research, data classification based on histograms and Gabor filters is the most important part. However, factors such as season can have an impact on classification quality. A combination of the input data from visible light cameras and infrared imagers might improve the initial quality of the data classification. Regarding forests, Khryashchev and Larionov [8] proposed a modification of U-Net with a balanced batch, which performed better when

segmenting forests in aerial images. They trained the network using a dataset including 17 images, but they realised that the presence of outliers negatively affected the results and overall training, so they performed per channel histogram equalisation with min-max values chosen by thresholding the cumulative distribution function, so they had values from 0 to 1. Then they split the images into training and testing sets. It was discovered that the training dataset is unbalanced, which has an impact on network learning. To increase the size of the training set, they added various image augmentation techniques, which significantly increased the quality of the final segmentation. The authors in [9] tried to propose an accurate algorithm for object detection in aerial images taken by UAVs. They proposed an accurate object detection method to measure areas more precisely in image processing-based masks. They noticed it was necessary to improve the masking performance obtained from Mask Region-based CNN (R-CNN). Regarding the detection of other objects and elements like roads, Ichim and Popescu [11] evaluated a fast and reliable method for road detection in satellite images using CNN. They noticed the input images, which were generated with Agisoft Photoscan Professional Edition [10] with a primary processing filter and colour component, had very efficient effects on the images. Therefore, the primary processing of images was used both in the learning phase and in the operating phase. Their trained network was only effective on asphalt roads; for other types of roads, additional training is required. The performance of road segmentation depends on the altitude of the flight (low or mid-altitude), image resolution, and the CNN structure. Because of the varying size and texture of the roads in aerial images, they realised that the most important step in training a CNN is the pre-processing stage and selecting proper training data; this network can also be applied to pipeline or river segmentations. Meanwhile, Hu and Guo [12] tried to propose a method for the

detection of buildings in city aerial images. In an aerial image of a city, since there are too many buildings of various types and shapes, it is really hard to detect each building. They tested three methods to determine which one was best for detecting buildings in aerial images. They used different colours to mark buildings, and it has been observed that only 65% of buildings were detected correctly, and the rest, due to their small size, were not detected correctly by the system. After comparing three datasets and methods to detect the buildings, they realised that Mask R-CNN performed the best with TensorFlow (open-source machine learning and artificial intelligence software library). Also, they noticed most of our building's outlines are square or rectangular, so they can be detected by their shape as well. There has been little research towards the classification and detection of multi-objects in aerial images. One of the multi land types of research is DeepGlobe Challenges 2018 [32] which aims to proper solutions and approaches for a solid understanding of satellite image content. Their challenge is divided into three different tasks including classification and detection of the buildings, roads and land cover types.

Considering the above research, we were looking for an approach that covers the classification and detection of various land types not limited to a single land type.

Chapter 3

BACKGROUND ON NEURAL NETWORKS FOR LAND CLASSIFICATION IN AERIAL IMAGES

In this chapter, we provide an overview of the concepts and technologies that will be used in this thesis. These include aerial imagery and different machine learning models that are our main topic for detecting the type of land in aerial images. We also explain the various types of aerial imagery that are most common nowadays.

3.1 Aerial Imagery

First time in history in the mid-19th century aerial photography has been done by humans. In 1858, Nadar, a French photographer and balloonist, took the first aerial photograph from a tethered hot-air balloon in Paris's sky. The photograph showed the rooftops and streets of the city from a bird's point of view. Also, the UAV machine's first aerial photographs were taken by Julius Neubronner in Germany in the early 1900s, using pigeons with small cameras attached to their bodies. In the early 20th century, manned aircraft were used for aerial photography, particularly during World War I, when aerial reconnaissance became an essential tool for military operations [1].

After the war, the use of aerial photography expanded to other areas, including cartography, surveying, and agriculture. The development of new camera technologies and aircraft designs, such as the invention of the gyro-stabilised camera mount in the 1930s, made aerial photography more efficient and effective. In the 1960s and 1970s,

the use of satellites for remote sensing and aerial photography further revolutionised the field, allowing for global coverage and the ability to collect data over time [13]. Nowadays, aerial photography has the potential to be used in a wide range of applications such as science and technology fields like urban planning, environmental monitoring, and archaeology. The evolution of precise aerial imagery with the help of new technologies looks significant compared to the early days of the invention.

3.2 Categories of Aerial Imagery

In this section, we will discuss the different categories of aerial imagery, including vertical imagery, oblique imagery, orthoimagery, thermal imagery, multispectral imagery, Light Detection And Ranging (LiDAR), satellite imagery, and drone/UAV imagery. We will define each type of aerial imagery and discuss its applications.

3.2.1 Vertical Imagery

When our point of view of the terrain or ground is directly from top to down it's called vertical aerial imagery which is mostly used for mapping, land use planning, and environmental monitoring [3].

3.2.2 Oblique Imagery

Unlike vertical imagery the term oblique imagery refers to situations where our point of view is a more three-dimensional view of the area in the captured image which most of the time has application in urban planning and visualisations [2].

3.2.3 Orthoimagery

This type of vertical imagery is the corrected version of images due to falsifications and faults because of terrain relief, camera tilt, and other possible issues indeed. These geometrically corrected images are widely used for mapping, land use planning, and emergency response [13].

3.2.4 Thermal Imagery

Unlike other types of aerial imagery when our aim is to capture radiations that are in the infrared range and record the temperature of the ground, it's called thermal imagery which is really useful in environmental monitoring, energy auditing, and agriculture [14].

3.2.5 Multispectral Imagery

The aim of multispectral imagery is to record lights in some wavelengths, such as visible, near-infrared, and infrared which has many applications in environmental monitoring, vegetation analysis, and mineral exploration [15].

3.2.6 LiDAR

In some cases, laser-light technology can be used to create highly precise 3D models of the ground or targeted terrain. This type of aerial imagery has potential for mapping, land use planning, and environmental monitoring [16].

3.2.7 Satellite Imagery

Orbiting satellites in space with the help of their powerful imaging sensors are able to capture images and collect a wide range of useful data from the earth's surface including land cover, vegetation, topography, and weather patterns. Although it's mostly considered a type of aerial imagery in general it more fits to large-scale mapping and monitoring purposes compared to normal aerial imagery which is the better option for detailed mapping and analysis of smaller regions. Generally, this imagery type is the best solution for understanding and monitoring the Earth's surface [17].

3.2.8 Drone/UAV Imagery

Similar to satellite aerial imagery UAVs or drones as heard the most these days are able to capture high-resolution images and collect data from our planet [18]. Compared

to satellite imagery UAVs have advantages over satellites as explained below:

- Spatial resolution is higher in UAVs than satellites due to their ability to fly at lower altitudes and utilise high resolutions meanwhile which leads to collecting more detailed data and images from our planet's surface.
- For smaller areas or regions such as individual properties or construction sites UAV aerial imagery will be used more often while satellite imagery can be used in large regions like whole cities or countries.
- The economical costs of UAV aerial imagery due to its less equipment requirement and personnel compared to satellite imagery made them a better solution for small-scale projects.
- The flexibility of UAV aerial imagery compared to satellite imagery is higher since UAVs are able to fly on demand and record images anytime at any desired angle. But satellite imagery's limits due to their orbits made them only capable of recording specific angles and limited time of the day.
- Although environmental factors are able to falsify imagery in both UAVs and satellites since UAVs are able to fly at low altitudes and more challenging situations, they are considered the preferred option compared to satellites [19].

In summary, we can only choose UAVs over satellites depending on certain and specific demands of the project. Both are high-resolution tools capable of capturing high-definition images. Recent technological inventions in UAVs and high-resolution satellites extended the current vast list of aerial imagery applications such as mapping and surveying environmental changes and disasters.

3.3 Aerial Image Segmentation

When we split up an aerial image into more homogeneous segments or meaningful

regions which point at a specific object or feature in the image is so-called aerial image segmentation. The aim of this process is to find out and classify different land cover or land usages, such as buildings, roads, vegetation, wetlands, water bodies, and bare soil. The segmentation process can include many steps but is mostly considered data preprocessing, feature extraction, and classification. The task of correcting geometric and radiometric faults and removing noises or artefacts in images is usually called the preprocessing step. Then, all corresponding features or attributes of the image like shape, texture or pattern, context, and colour will be extracted from every pixel group in the next step (feature extraction) in order to differentiate each object from others by recording its specific attributes. Finally, in the classification step, a machine learning algorithm is used to assign each pixel or group of pixels to a specific land cover or land use class based on the extracted features. This process will be through different techniques including decision trees, neural networks, and some others as well.

Aerial image segmentation can also be used in remote sensing applications, including urban planning, land use mapping, environmental monitoring, and disaster management [20]. By accurately identifying and mapping different land cover or land use classes, aerial image segmentation can help decision-makers to make more informed and sustainable land management decisions.

3.4 Neural Networks

In this section, we will provide an overview of machine learning models as one of the most crucial parts of our approach in order to detect land types in aerial images. Neural networks have a similar structure to the human neuron system and their goal is solving artificial intelligence and complicated problems with the help of some algorithms by detecting and finding relationships in data. Below are some neural network

technologies we used in our solution.

3.4.1 Convolutional Neural Networks

CNNs are a type of deep learning algorithm that is widely used for image identification, detection of objects, and image content segmentation tasks [21]. They are derived from the visual cortex of the brain and are developed for self-learning and extracting relevant features from pictures. These networks include a couple of layers including convolutional layers, pooling layers, and fully connected ones. The convolutional layer is in charge of detecting features in the images through the use of learnable filters or kernels for each part of the picture. These filters learn to recognize specific patterns, such as edges, corners, and textures, at different spatial scales. The pooling layer will be applied in order to minimise the size of the feature maps by performing a downsampling operation, such as max pooling or average pooling. This helps to make the network more powerful to tiny variations in the input image and reduces the number of parameters and required calculations. The fully connected layers segregate the input image based on the extracted features. These layers take the flattened feature maps from the previous layers as input and use a set of weights and biases to produce a probability distribution over the possible classes. CNNs are trained using a large dataset of labelled images and a loss function that measures the difference between the predicted output and the true label. The weights and biases of the network are then updated using an optimization algorithm, such as stochastic gradient descent, to minimise the loss function. The potential of learning hierarchical presentation of the images is an advantage of CNNs which will be done in a way that lower layers are responsible to learn simple features and higher layers are assigned to learning complex features which are a mix of lower layer features indeed [22]. Through this process, CNNs attain their top performance on a variety of image recognition jobs.

Generally, CNNs are robust and pliable tools for computer vision tasks, image object detection, and analysing images which can be used in autonomous vehicles, security purposes, healthcare, and many other fields.

3.5 Machine Learning Models

Algorithms in mathematics developed in order to learn and recognise patterns and relations in data are called machine learning models. These types of the Artificial Intelligences (AI) made machines eligible to make decisions and make predictions based on existing or previous data [23]. Routing input data and output data through sets of training data by finding mathematical functions which can be used for making predictions or models is the target defined for machine learning models. These days, we use machine learning models in various applications such as natural language processing, image and speech recognition, computer vision, robotics, and finance in order to automate complex and complicated tasks, optimise decision-making, and speed up scientific processes. Machine learning models can include but are not limited to linear regression, logistic regression, decision trees, random forests, support vector machines, and neural networks. Machine learning models are a type of artificial intelligence that can learn from data and make predictions. There are three main types of machine learning models: supervised learning, unsupervised learning, and reinforcement learning.

3.5.1 Supervised Learning

In supervised learning, we train the model by utilising labelled samples in a way that inputs are related to the corresponding output or label. Our aim is to learn or obtain a function that is able to predict the labels of new incoming data more precisely.

3.5.2 Unsupervised Learning

In contrast to supervised learning, we use unlabelled samples [24] in order to train our

model in unsupervised learning in order to find invisible patterns in our data which can be used in clustering, anomaly detection, or dimensionality reduction tasks.

3.5.3 Reinforcement Learning

When our model is trained based on responses like rewards or penalties it's called the reinforcement learning model. The goal of such a model is to discover a method that helps the system learn long-term cumulative rewards.

3.6 R-CNN

The term region-based convolutional neural network refers to an algorithm for computer vision that was founded by Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik in 2014 in order to detect objects [25] which basically formed the concept of classifying and recognizing objects in images for the R-CNN networks. In contrast to CNN which processes the whole image, R-CNN by using selective search algorithms produces a set of region proposals which are Regions of Interests (ROI) that mostly include an object inside them. R-CNN implies a separate CNN for each one of the ROIs in order to take out a fixed-length feature vector which this vector later will be sent to a couple of class-specific linear SVMs to forecast the existence or absence of every object. In the end, to filter the location of the object purpose, a bounding box regression model will be used.

Nevertheless, R-CNN has limitations such as slow training and inference times because the process requires each ROI to be independent [26]. Fast R-CNN, Faster R-CNN, and Mask R-CNN are some of the variations developed in order to improve and pointing the current limitation of R-CNN and obtain more precise results in object detection and segmentation tasks compared to the basic R-CNN [27]. Figure 1 represents the progress in R-CNN.

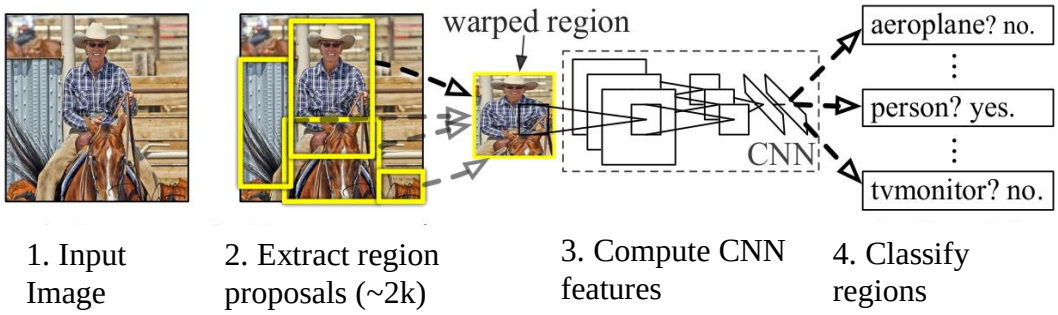

Figure 1: R-CNN process [26]

The potential power of detecting and recognizing objects in complex and difficult environments made R-CNN and all its variants useful and practical in our world like autonomous vehicles, robotics, surveillance, and medical imaging.

3.7 U-Net Network

U-Net as a type of CNN is mostly used for biomedical image segmentation tasks [28], but it has been successfully used in a variety of other applications as well. It was introduced by Ronneberger, Fischer and Brox in [29]. The U-Net architecture is named for its U-shaped structure. The architecture of the U-Net is depicted in Figure 2.

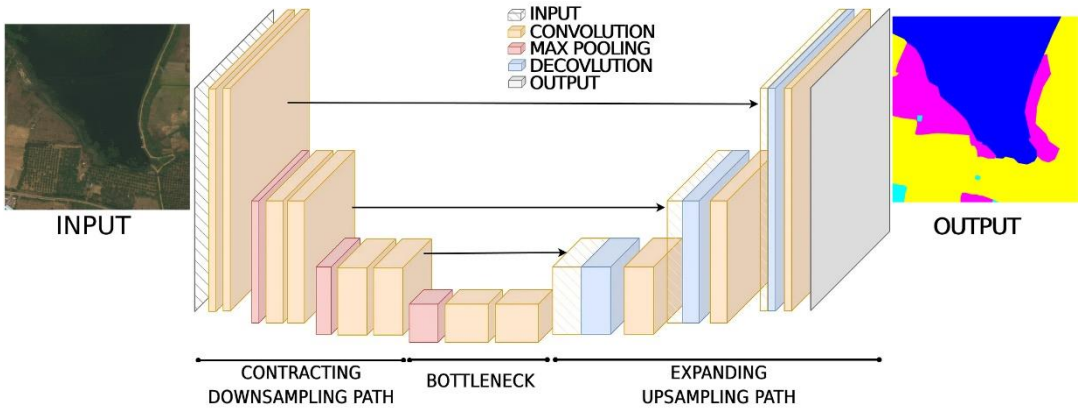

Figure 2: U-Net network architecture

U-Net is proved to be very effective in areas where the dataset might be smaller than

those normally used for convolutional neural networks. It's also designed to provide precise classification or segmentation which makes it ideal for tasks where detailed identification and localization of objects within the image is required [30]. The U-Net architecture, shown in Figure 4.2, will be explained in more detail below.

3.7.1 Contracting/Downsampling Path

This part of the network has the standard architecture of a convolutional network. It includes two 3x3 convolutions each followed by a Rectified Linear Unit (ReLU) and a 2x2 max pooling operation with stride 2 for downsampling. At each downsampling step, the number of feature channels is doubled. The contracting path is a sequence of repeated groups of two 3x3 convolutions and a 2x2 max pooling operation with stride 2 for downsampling. Each time we go one layer deeper in the network, we double the number of feature channels. This part of the network captures the high-level semantic information and encodes it into the feature maps.

3.7.2 Bottleneck

This part of the network enables the model to learn more complex features. It operates after the contracting and before the expanding paths and consists of two convolutional layers with ReLU activations.

3.7.3 Expanding/Upsampling Path

Every step in the expansive path consists of an upsampling of the feature map and a 2x2 convolution (“up-convolution”) that cuts the number of feature channels into two, a concatenation with the correspondingly cropped feature map from the contracting path, and two 3x3 convolutions, each one followed by a ReLU. The expanding path (up-sampling part) is symmetrical to the first part. Every step in the expanding path includes an up-sampling of the feature map followed by a 2x2 convolution (up-convolution), a connection with the correspondingly cropped feature map from the

contracting path, and two 3x3 convolutions, each followed by a ReLU. These operations help to localize and precisely segment the important features in the image.

3.7.4 Final Layer

At the final layer, a 1x1 convolution will be used in order to map each 64-component feature vector to the expected number of classes.

Chapter 4

METHODOLOGY, RESULTS AND DISCUSSIONS

In this chapter, we first explain the methodology used to ensure the reliable performance of the proposed network. The methodology was used to develop a deep learning system for detecting the type of land in aerial or satellite images. We will then discuss our results and the progress we made to train and test our neural network for classifying land types in aerial images in detail. The performance of the proposed U-Net-based approach is also evaluated using performance metrics such as accuracy and precision.

4.1 Methodology

In this section, we will explain the methodology used to develop a deep learning system for detecting the type of land in aerial images.

4.1.1 Data Preparation

We started by preparing our dataset. The dataset was a large collection of 1,146 aerial or satellite images and their corresponding mask image files, collected with various types of land cover such as forests, roads, rivers, rocks, fields, and buildings and corresponding land usage labels (masks) which is well-known as DeepGlobe Challenge 2018 dataset [32]. We created the classes for the pixels in the masks and stored the data in the MATLAB datastore, which allows for efficient reading and preprocessing of large datasets. The images were also preprocessed and prepared by standardising, resizing, and cropping them to a consistent size.

4.1.2 Defining the Network Architecture

In our thesis study, we used a U-Net architecture for our semantic segmentation task. U-Net is particularly effective for image segmentation tasks due to its symmetric structure. The U-Net architecture is composed of a contracting path to capture the context and an expansive path to allow precise localization. U-Net as a type of CNN is perfectly prepared for biomedical image segmentation, but it can be easily adapted for different semantic segmentation tasks including our case in aerial imagery segmentation. The symmetric structure of the U-Net beside a contracting path to capture context led us to an extensive path to precise localization.

4.1.3 U-Net Configuration

U-Net is a deep learning architecture that is well-suited for image segmentation tasks. It has several qualities that make it suitable for our work, including its ability to handle small datasets, produce fine-grained segmentations, train efficiently, use skip connections, and generalize to other tasks, as described below.

4.1.3.1 Effective for Small Datasets

Unlike some other deep learning models, U-Net does not require a large amount of data to produce good results. This is particularly useful in scenarios where we don't have access to a vast amount of labelled training data.

4.1.3.2 Fine-grained Segmentation

U-Net provides fine-grained segmentation thanks to its symmetric expanding path which allows precise localization of the detected classes. This is particularly useful when we are interested in detailed segmentation, not just image classification.

4.1.3.3 Efficient Training

U-Net's architecture allows for efficient training. The use of up-convolution in the upsampling path allows for faster and less memory-intensive training.

4.1.3.4 Skip Connections

The use of skip connections allows U-Net to use information from multiple resolution scales simultaneously. This is beneficial for tasks where features of different sizes are relevant, and help to address the vanishing gradient problem in deep neural networks.

4.1.3.5 Generalizability

Despite being originally designed for biomedical image segmentation, U-Net's architecture has shown great generalizability and has been effectively used in a variety of other tasks, such as satellite image analysis, object detection in autonomous driving, etc.

Considering these advantages, choosing U-Net for our work involving semantic segmentation seems like a good choice. It is also worth noting that the specific suitability of a model depends on the specifics of the problem at hand, and sometimes it may be beneficial to try out different models to see which performs best on our specific task.

4.1.4 Training Options

For our neural network training which is basically an optimization process where we are looking to minimise the difference (often called the "loss" or "cost") between the network's predictions and the actual labels in our training data. In this case, we used the Adaptive Moment Estimation (ADAM) [30] optimization algorithm for this purpose. ADAM as a method for efficient stochastic gradient descent operates by maintaining a per-parameter learning rate that improves performance when dealing with sparse gradients on noisy problems. It calculates adaptive learning rates for various parameters. During each iteration of the training process, the ADAM optimizer updates the network's weights based on the current batch of data. The weights are

adjusted in the direction that decreases the loss. The magnitude of this adjustment is determined by the learning rate and the gradient of the loss with respect to each weight. Gradually, these adjustments cause the loss to converge to a minimum. At last, to prevent overfitting, we used early stopping during the training progress. This includes halting the training process if the network's performance on a separate validation set fails to improve after a certain number of iterations. This helped us ensure that our model generalises fine to ungiven data. The goal of our training step was to train a model that can accurately analyse the labels of ungiven images, based on the patterns it has learned from the training data.

4.1.5 Network Training

We trained our network using the provided training data including images and mask data and the specified training options. During the training phase, the network adjusts its weights to minimise the difference between the predicted and existing labels of the training images.

4.1.6 Validation and Testing

We tested the trained network on our provided test data to evaluate its performance. We did this by comparing the network's predicted labels with the actual labels.

4.1.7 Performance Evaluation

We compared our network predicted data with existing data to have a clear vision of our network performance.

4.2 Results and Discussions

In this section, we will discuss our results and the progress we made in order to be able to train and test our neural network for classifying land types in aerial images in detail.

We started by preparing our dataset [32] images including 1,146 image files and

corresponding mask images and categorising them into three different folders including training images, validation images and test image. First, we correlated each image to its mask image which is labelled by the desired colour representing the land type as shown in Table 1.

Table 1: Colour definition of lands for each label

Land Type	**Red**	**Green**	**Blue**	**Colour**
urban_land	0	255	255	
agriculture_land	255	255	0	
rangeland	255	0	255	
forest_land	0	255	0	
water	0	0	255	
barren_land	255	255	255	
unknown	0	0	0	

We stored the images and mask files in image Datastores and used MATLAB functions to resize all the images and mask files to ensure all of them have similar dimensions.

We employed image cropping as a part of the data preparation process for our U-Net

segmentation model. The primary reason for choosing cropping over resizing was to retain the original resolution and the intricate details of the input images that could otherwise be lost due to the downscaling process in resizing. After that, we cropped the images by creating non-overlapping 256x256 pixel boxes as shown in Figure 3.

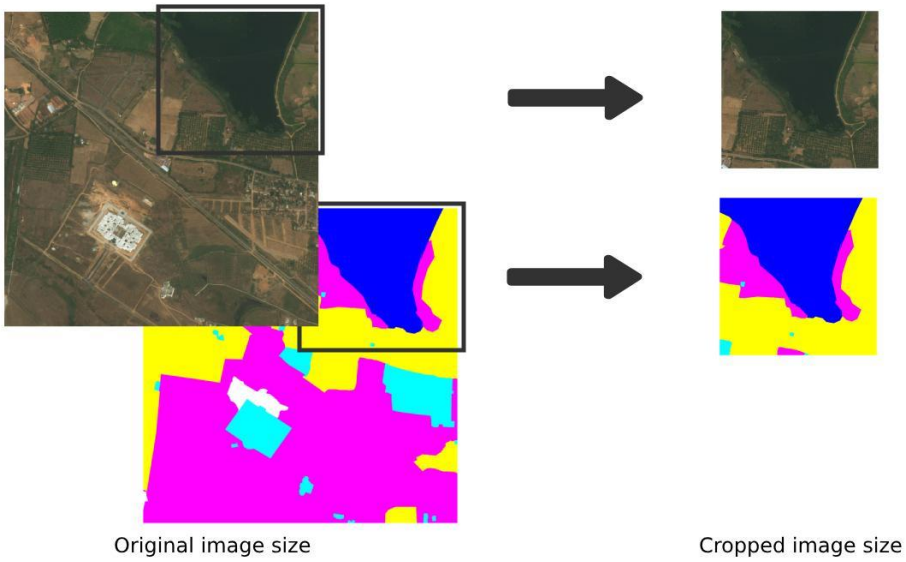

Figure 3: Cropping process to retain original resolution and intricate details

We started by determining the dimensions of our original images, and then sequentially moved through the image, extracting fixed-size 256x256 pixel boxes. The result of this process was multiple smaller cropped images. We also ensured that the corresponding masks (label images) experienced the exact same cropping process. It was vital to maintain the correspondence between the images and their respective masks because the pixel-level labels in the masks are what enable the semantic segmentation process. By following this solution, we ensured that the detailed information inherent in the high-resolution images was perfectly retained for the U-Net model to learn from, thus helping improve the model's performance and accuracy

in segmenting complex images. The cropping progress helped us to prevent future issues regarding low-resolution images or low-quality images and also provided a larger quantity of training samples to achieve a higher accuracy during the training process. Then we designed the network architecture including the size and dimension of input images, input classes (7 classes in our case since we have seven different land types), network depth (4 in our case which is the most common choice for U-Net network) as described in details in Table 4.2 and a layer graph to understand and monitor network architecture (with the help of Deep Learning Toolbox™ [33] and Image Processing Toolbox™ [34]). The network simulation parameters are described in detail in Table 2.

Table 2: Network simulation parameters

Simulation Parameter	**Value**
Number of epochs	1000 epochs
Number of images	1146 images
Image size	256 x 256 pixels
Number of training samples	4584 images
Number of test samples	459 images
Learning rate	0.001
Validation Type	Cross validation
Validation frequency	30
U-net network depth	4
Batch size	32

We started the training process with the minimum number of epochs as shown in Table 3 and gradually increased the number of epochs to compare the accuracy and loss in the training phase.

Table 3: Network training phase options and results

Number of Epochs	Accuracy	Loss
50	93.23%	0.3879
100	98.06%	0.1148
1000	98.55%	0.0093

Figure 4 represents the accuracy and loss graph during the last network training phase try. Since after 200 epochs, the graph remained stable and didn't show any considerable fluctuations we can say the rest of the epochs were not effective for training our network. As we can see in the graph, we experienced some fluctuations during early iterations and epochs while after 20 to 30 iterations the accuracy steadily increased and meanwhile the loss decreased gradually with a smooth trend.

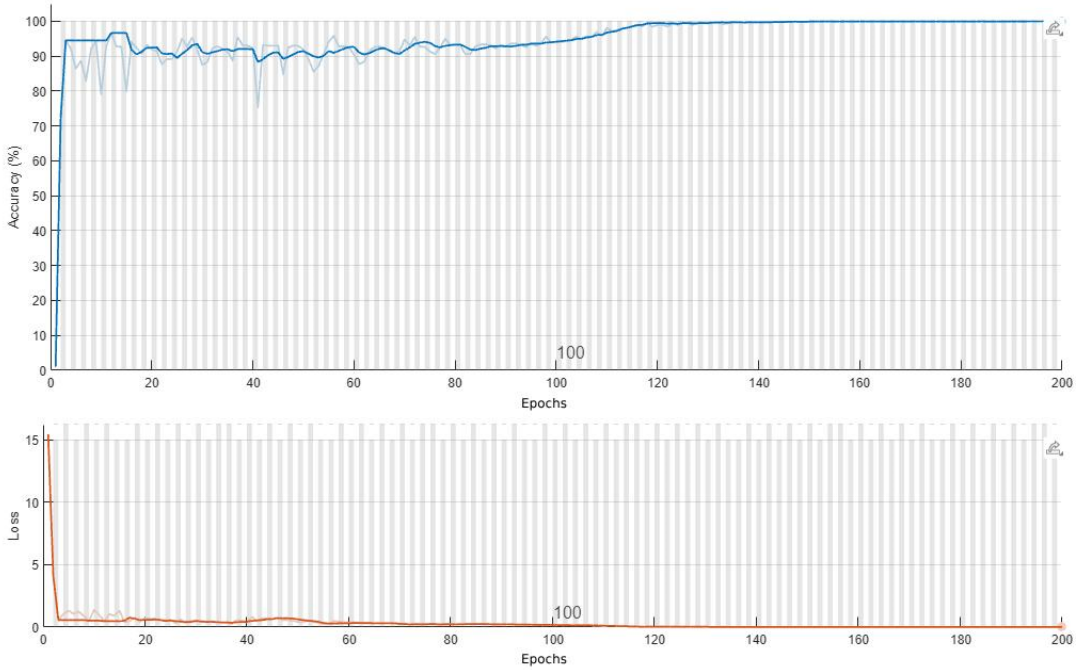

Figure 4: Accuracy and loss graph over the iteration during the network training process

Increasing the number of epochs directly affected our network training duration. A comparison of the training duration time is represented in Table 4.

Table 4: Network training duration comparison

Number of epochs	**Duration**
50	40 mins 32 sec
100	2 hrs 57 mins 48 sec
1000	9 hrs 47 mins 59 sec

The machine we used in this regard was a personal computer with Core™ i7-7560U with 16 GB RAM and a GeForce GTX 1080 GPU. In early attempts to train the

network, we experienced lots of obstacles, so we optimized the code to receive the maximum performance of the GPU and system. After training the network we tested our network with some unseen and ungiven input images to evaluate our network performance as illustrated in Figure 5 The proposed network was able to predict forest lands, rangelands and barren lands more efficiently and more precisely compared to other land types. The weakness of our proposed network was in detecting urban lands and water (wetlands) where we experienced more difficulties.

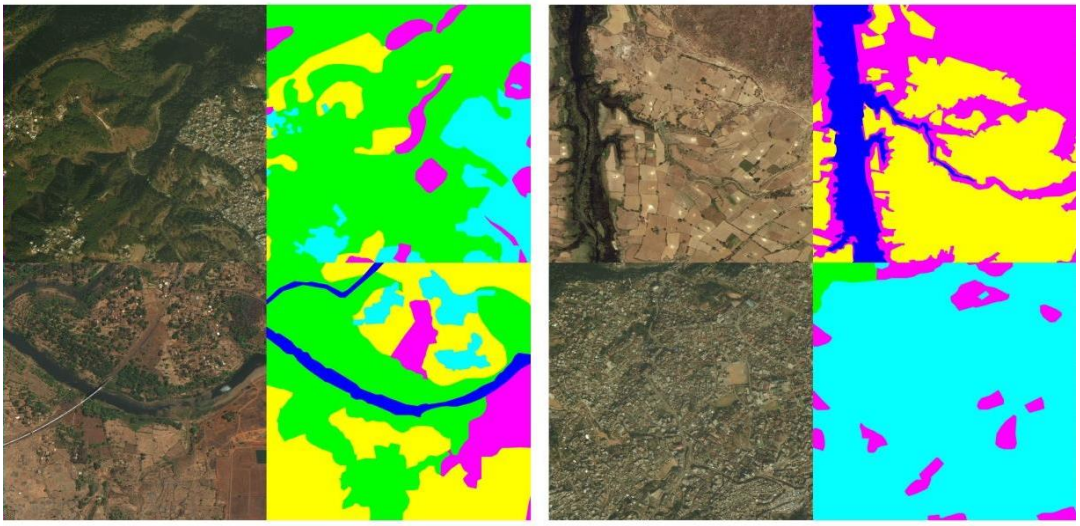

Figure 5: Comparison of the original image and the network's prediction

We can observe the highly accurate prediction in some classes like Urbanland and Water classes in the predicted results. Although some objects which are not defined in our network like roads exist in the images, but our network was mostly able to ignore unnecessary objects and focus on the main land type classes. After testing our network with unseen images, we evaluated its performance using a confusion matrix. The confusion matrix, also called in some cases the error matrix, is a specific table that allows visual analysis of the performance of a normally supervised learning model algorithm. This table is called a confusion matrix because it can show us what classes

our model is "confused" about and is not able to predict easily. Each row of the matrix represents the samples in a predicted class while each column shows the samples in an actual class. The confusion matrix makes it easy to see if the system is confusing two classes. In binary classification, the confusion matrix includes four parameters:

1. **True Positives (TP)**: The cases in which the model predicted yes, and the true label is also yes.
2. **True Negatives (TN)**: The cases in which the model predicted no, and the true label is no.
3. **False Positives (FP)**: The cases in which the model predicted yes, but the true label is no.
4. **False Negatives (FN)**: The cases in which the model predicted no, but the true label is yes.

This applies similarly to our case which is a multi-class classification problem, where each class is considered separately, and we have a 2x2 confusion matrix for each class. We also evaluated precision, recall and F-score measures based on the confusion matrix. We generated a confusion matrix for our system, as shown in Figure 6 with the help of MATLAB’s predefined functions.

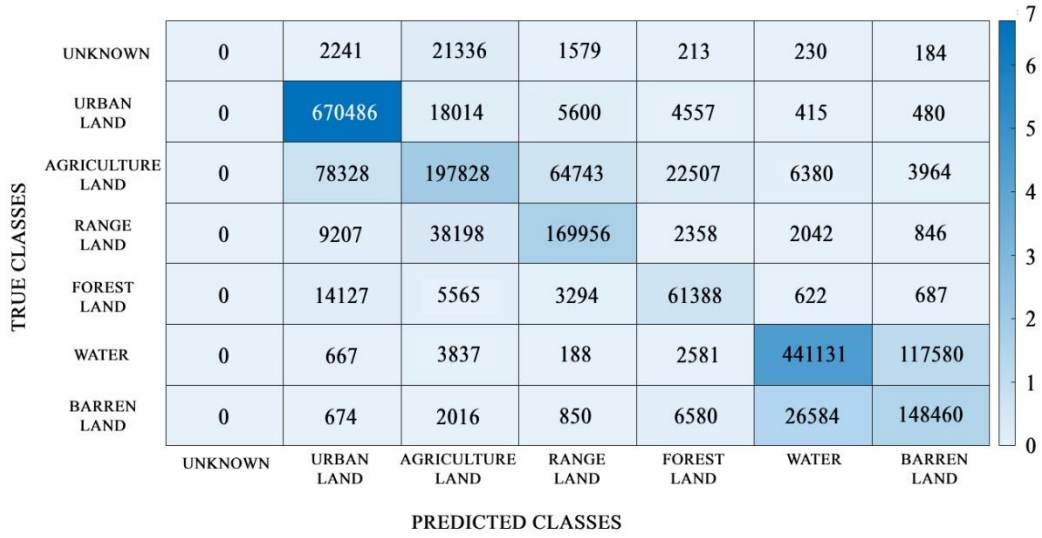

Figure 6: Confusion matrix of the system

As we can see in the confusion matrix, the most confusing land type class for our trained network was Urbanland, while Forestlands showed the least confusing amount compared to the other land type classes. Based on our confusion matrix which represents the TP, FP, FN and TN for each land type class, we can calculate performance metrics such as accuracy, precision, recall, and F1 score. Accuracy is commonly used for binary and multi-class classification networks. It is calculated as the number of correct predictions (both positive and negative) divided by the total number of predictions and defined by the following formula:

$$Accuracy = \frac{(TP + TN)}{(TP + FP + TN + FN)}$$

This calculation gives us a ratio between 0 and 1 (or a percentage) that represents the proportion of predictions our model was right. A value of 1 means our model correctly predicted all samples, while a value of 0 means it incorrectly predicted all samples. Also, we can generate the precision which quantifies the ability of a classification model to make right and precise positive predictions. It's also called Positive Predictive Value (PPV) and can be calculated as follows:

$$Precision = \frac{TP}{TP + FP}$$

In other words, precision is the ratio of correctly predicted positive observations (TPs) to the total predicted positives, which is the sum of TPs and FPs. High precision represents a low FP rate, i.e., a smaller number of positive predictions were incorrect. Since in most cases precision and accuracy cannot be used alone as a reliable metric, we used recall which is a metric that measures the ability of a classification model to find all the relevant cases within a dataset. The formula to calculate recall is:

$$Recall = \frac{TP}{TP+FN}$$

In simple words, recall is the ratio of correctly predicted positive observations (TPs) to all observations in the actual class. High recall presents that the class is correctly recognized. To make our performance analysis more reliable we calculated the F1 score based on the below formula:

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

The F1 score is useful because it checks a balance between precision and recall which we already obtained. It's a perfect metric to use when the costs of FPs and FNs are very different. Also, we calculated the specificity for our test results based on the following formula:

$$Specificity = \frac{TN}{FP + TN}$$

Specificity which also is known as the TN rate measures the proportion of actual negatives that are correctly identified. Specificity is particularly useful when we want to focus more on the correct classification of negatives. In some scenarios, avoiding FPs is more important than catching all positives. Thus, we have Table 5 as follows:

Table 5: Network performance evaluation

Class	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-score (%)**	**Specificity (%)**
UrbanLand	93.78	86.43	95.85	90.90	92.79
AgricultureLand	87.73	68.98	52.93	59.90	95.02
Rangeland	94.03	69.03	76.35	72.50	96.06
ForestLand	97.08	61.28	71.65	66.06	98.13
Water	92.54	92.40	77.94	84.56	97.72
BarrenLand	92.57	54.54	80.18	64.92	93.73
Unknown	98.81	0	0.00	0	100.00

Based on the confusion matrix, we can represent the distributions of the samples for each class type. As mentioned earlier we had 4,125 image files for training which each image file could contain various number of samples. It means for each one of the 4,125 image files we could have one or more sample of each land type class. The exact amount for number of samples of each class type in our images is depicted in Table 6.

Table 6: The number of samples for each class

Class Name	**Number of Samples**
UrbanLand	25,783
AgricultureLand	699,552
Rangeland	373,750
ForestLand	222,607
Water	85,683
BarrenLand	565,984
Unknown	18,5164

Apart from normal F1-score and normal network performance evaluation, we tried to add more network performance evaluation parameters including the micro and macro F1-score. The micro F1-score is actually a type of F1-score which gives equal weight to each individual sample. It can be useful when we have a multiclass problem with imbalanced class sizes. Basically, micro precision is equivalent to micro recall in this scenario, and equivalent to the accuracy of the system. So, the micro F1-score can be considered as a way to measure the harmonic mean of the system accuracy, taking into account class imbalance. To calculate the micro F1-score for our network, we calculate the global count of TPs, FPs, and false negatives across all class types, and then we use those counts to calculate precision and recall, which are used to compute the F1-

score as bellow:

$$Net\ TP = 0+670486+197828+169956+61388+441131+148460 = 1689249$$

$$Net\ FP = 0+105244+88966+76254+38796+36273+123741 = 469274$$

$$Net\ FN = 25783+29066+175922+52651+24295+124853+36704 = 469274$$

$$Micro\ Precision = Net\ TP/(Net\ TP+Net\ FP) = 1689249/(1689249+469274) = 78.25\%$$

$$Micro\ Recall = Net\ TP/(Net\ TP+Net\ FN) = 1689249/(1689249+469274) = 78.25\%$$

$$Micro\ Precision = Micro\ Recall = Micro\ F1\text{-}Score = Micro\ Accuracy = 78.25\%$$

Also, regarding the Macro F1-score which is another type of the F1-score, it is useful in multiclass classification cases like ours. Unlike the Micro F1-score which calculates metrics generally by counting the total TPs, FNs and FPs, the macro F1-score calculates metrics for each label and then finds their unweighted mean. The macro F1-score treats all classes equally, ignoring their size, which makes it a perfect metric when dealing with imbalanced classes. To calculate the macro F1-score, we can compute the F1-score for each class and then we average them. This includes computing the precision and recall for each class and then using these values to compute the F1-score for each class:

$$Macro\ Precision = 72.10\%$$

$$Macro\ Recall = 64.98\%$$

$$Macro\ F1\text{-}Score = 73.13\%$$

Considering the above metrics and the fact that we can calculate the weighted accuracy, weighted precision, weighted recall and weighted F1-score based on the below formula:

$$Weighted\ Accuracy = \frac{(class\ 1\ Accuracy \times Number\ of\ Samples + \cdots + class\ n\ Accuracy \times Number\ of\ Samples)}{(Class\ 1\ Number\ of\ Samples + \cdots + Class\ n\ Number\ of\ Samples)}$$

So, we can have Table 7 for analysis overview:

Table 7: Average network performance analysis overview

Metric	**Micro**	**Macro**	**Weighted**
Accuracy	78.2	93.7	92.2
Precision	78.2	72.1	59.6
Recall	78.2	64.9	63
F1-Score	78.2	73.1	60.2

According to Tables 5 and 7, the results of the semantic segmentation model as detailed in those tables showed varied performance for various classes of land type. The model illustrated particularly strong performance on the UrbanLand and Water class types, which could be due to various affecting factors including the individual and comparable nature of these classes that allowed our model to easily discern patterns and make more accurate predictions. Regarding UrbanLand class type, the model achieved an accuracy of 93.78%, in addition to precision, recall, and F1 score of 86.43%, 95.85%, and 90.90% respectively. These high metrics showed that our model was very effective in identifying and correctly classifying this type of land, similarly because of unique and distinctive urban features such as buildings, roads, and other human-made structures which create clear boundaries and contrasts compared to

natural landscapes. The Water class types also demonstrated high performance with an accuracy of 92.54% and an impressive precision of 92.40%. It appears that our model could easily detect water classes, most probably due to their distinct spectral attributes and homogeneity that significantly varies from earthly features. On the contrary, the model's performance for other classes was less compatible. For instance, the AgricultureLand and BarrenLand classes had significantly lower F1 scores of 59.90% and 64.92% respectively. This difference could be attributed to factors such as the complex nature of these land types, which might include a variety of features that are difficult for our model to separate. Additionally, the similarity of these classes to others, such as Rangeland, might have led to confusion and difficulty, thus impacting our model's precision and recall for these class types. The general performance metrics showed a micro F1-score of 78.25% and a macro F1-score of 73.13%. The micro F1-score is a perfect measure when class imbalance is an issue, as it aggregates the contributions of all classes to calculate the average metric. On the other hand, the macro F1-score calculates the metrics for each class separately and then receives the average, thus treating all classes equally, ignoring their size. The difference in these metrics indicates the varied performance of our model across various class types and the presence of a class imbalance in our dataset. Considering the results we achieved, we can suggest potential avenues for improving the model's performance, such as additional training data, fine-tuning the model's parameters, or exploring data augmentation techniques, especially for the class types where the model showed low performance. Figure 7 demonstrates graphs for comparison of various land type classes.

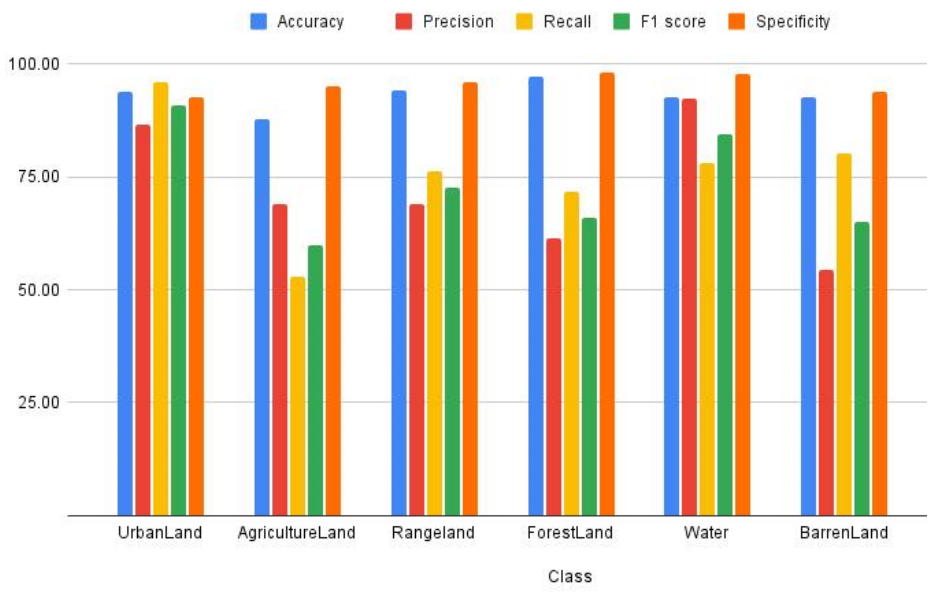

Figure 7: Network performance analysis

Chapter 5

CONCLUSION

5.1 Conclusion

In this thesis, we tried to propose a system to classify land types in aerial imagery with the help of CNNs. We chose a proper dataset of aerial images and eliminated those unclear and cloudy ones to be able to optimize our network performance and training accuracy. Choosing a proper dataset and training the network with large image dimensions besides a low learning rate and proper network training options highly affect the network accuracy and performance. Our aim was to establish a deep-learning solution to classify various land types simultaneously with the help of the U-Net model since we found other existing solutions single purpose. The semantic segmentation model performed well on the UrbanLand and Water class types. This is likely due to the distinct spectral attributes and homogeneity of these land types, which made them easier for the model to identify. However, the model's performance for other classes was less compatible, such as the AgricultureLand and BarrenLand classes, which had low F1 scores. This difference could be attributed to the complex nature of these land types, which might include a variety of features that are difficult for the model to separate.

Considering the results we achieved, we can suggest potential avenues for improving the model's performance, such as additional training data, fine-tuning the model's parameters, or exploring data augmentation techniques, especially for the class types

where the model showed low performance. Since our proposed system covers only 6 land types and the system is only able to operate in existing and available image datasets and we did not consider the cloudy and other possible obstacles such as oblique images.

5.2 Future Work

As a future work, we also plan to broaden the land types included in our system and try to find a real-time system which is able to do this task in a real-time aerial video recording by UAVs or satellites regardless of altitude and camera angles or even cloudy weather conditions.

REFERENCES

[1] T. Driver, “A History of Aerial Photography and Archaeology: Mata Hari’s Glass Eye and other Stories. By MartynBarber,” *The Archaeological Journal*, Jan. 2011, doi: 10.1080/00665983.2011.11020897.

[2] J. Y. Rau, J. P. Jhan and Y. C. Hsu, "Analysis of Oblique Aerial Images for Land Cover and Point Cloud Classification in an Urban Environment," in *IEEE Transactions on Geoscience and Remote Sensin*g, vol. 53, no. 3, pp. 1304-1319, March 2015, doi: 10.1109/TGRS.2014.2337658.

[3] R. C. Gonzalez and R. E. Woods, "Digital Image Processing," *4th edition, Pearson Education*, Inc., 2018.

[4] K. Maesako and L. Zhang, "AVIS: An Innovative Image Preprocessing Method for Object Detection of Aerial Images," *2022 IEEE Wireless Communications and Networking Conference (WCNC*), Austin, TX, USA, 2022, pp. 920-925, doi: 10.1109/WCNC51071.2022.9771814.

[5] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[6] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," *2017 IEEE*

Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, pp. 6517-6525, doi: 10.1109/CVPR.2017.690.

[7] L. Kopečný and J. Hnidka, "Aerial Landscape Recognition via Multi-Input Neural Network," *2021 International Conference on Military Technologies (ICMT),* Brno, Czech Republic, 2021, pp. 1-5, doi: 10.1109/ICMT52455.2021.9502749.

[8] V. Khryashchev and R. Larionov, "Wildfire Segmentation on Satellite Images using DeepLearning," *2020 Moscow Workshop on Electronic and Networking Technologies (MWENT),* Moscow, Russia, 2020, pp. 1-5, doi: 10.1109/MWENT47943.2020.9067475.

[9] S. Kim, Y. Han, S. Jeon and D. Seo, "Improvement of Object Segmentation Accuracy in Aerial Images," *2022 IEEE International Conference on Consumer Electronics (ICCE),* Las Vegas, NV, USA, 2022, pp. 1-5, doi: 10.1109/ICCE53296.2022.9730543.

[10] "Agisoft Metashape: Professional Edition." https://www.agisoft.com/features/professional-edition/

[11] L. Ichim and D. Popescu, "Road Detection and Segmentation from Aerial Images Using a CNN Based System," *2018 41st International Conference on Telecommunications and Signal Processing (TSP),* Athens, Greece, 2018, pp. 1-5, doi: 10.1109/TSP.2018.8441366.

[12] Y. Hu and F. Guo, "Automatic Building Extraction Based on High Resolution

Aerial Images, "*2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE)*, Xiamen, China, 2019, pp. 1017-1020, doi:10.1109/EITCE47263.2019.9094824.

[13] D. Cosgrove and W. W. Fox, Photography and Flight. 2010.

[14] I. Lahouli, Z. Chtourou, R. Haelterman, G. De Cubber and R. Attia, "A Fast and Robust Approach for Human Detection in Thermal Imagery for Surveillance Using UAVs," *2018 15th International Multi-Conference on Systems, Signals & Devices (SSD)*, Yasmine Hammamet, Tunisia, 2018, pp. 184-189, doi: 10.1109/SSD.2018.8570637.

[15] T. Keaton and J. Brokish, "A level set method for the extraction of roads from multispectral imagery," *Applied Imagery Pattern Recognition Workshop, 2002. Proceedings.*, Washington, DC, USA, 2002, pp. 141-147, doi: 10.1109/AIPR.2002.1182268.

[16] S. Yi, S. Worrall and E. Nebot, "Geographical Map Registration and Fusion of Lidar-Aerial Orthoimagery in GIS," *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, Auckland, New Zealand, 2019, pp. 128-134, doi: 10.1109/ITSC.2019.8917305.

[17] Q. Ma, Y. Su and Q. Guo, "Comparison of Canopy Cover Estimations From Airborne LiDAR, Aerial Imagery, and Satellite Imagery," *in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 9, pp. 4225-4236, Sept. 2017, doi: 10.1109/JSTARS.2017.2711482.

[18] Y. Bazi, "Two-Branch Neural Network for Learning Multi-label Classification in UAV Imagery," *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, Yokohama, Japan, 2019, pp. 2443-2446, doi: 10.1109/IGARSS.2019.8898895.

[19] A. Bouguettaya, H. Zarzour, A. Kechida and A. M. Taberkit, "Vehicle Detection From UAV Imagery With Deep Learning: A Review," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 11, pp. 6047-6067, Nov. 2022, doi: 10.1109/TNNLS.2021.3080276.

[20] H. Wang, C. Zhuang, J. Zhao, R. Shi, H. Jiang, Y. Yuan, X. Guo, Z. Xue, "Research on Evaluation Method of Aerial Image Segmentation Algorithm," *2022 7th International Conference on Signal and Image Processing (ICSIP)*, Suzhou, China, 2022, pp. 415-419, doi: 10.1109/ICSIP55141.2022.9886900.

[21] T. Sinha and B. Verma, "A Non-iterative Radial Basis Function Based Quick Convolutional Neural Network," *2020 International Joint Conference on Neural Networks (IJCNN)*, Glasgow, UK, 2020, pp. 1-6, doi: 10.1109/IJCNN48605.2020.9206798.

[22] G. Lou and H. Shi, "Face Image Recognition Based on Convolutional Neural Network," in *China Communications*, vol. 17, no. 2, pp. 117-124, Feb. 2020, doi: 10.23919/JCC.2020.02.010.

[23] X. Zhao, "The Prediction of Apple Inc. Stock Price with Machine Learning Models," *2021 3rd International Conference on Applied Machine Learning*

(ICAML), Changsha, China, 2021, pp. 222-225, doi: 10.1109/ICAML54311.2021.00054.

[24] S. Hussein, P. Kandel, C. W. Bolan, M. B. Wallace, and U. Bagci, "Lung and Pancreatic tumor Characterization in the Deep Learning Era: Novel Supervised and Unsupervised learning Approaches," *IEEE Transactions on Medical Imaging,* vol. 38, no. 8, pp. 1777–1787, Aug. 2019, doi: 10.1109/tmi.2019.2894349.

[25] R. Girshick, J. Donahue, T. Darrell, and J. Malik, Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. 2014. doi: 10.1109/cvpr.2014.81.

[26] "Getting Started with R-CNN, Fast R-CNN, and Faster R-CNN - MATLAB & Simulink - MathWorks United Kingdom." https://uk.mathworks.com/help/vision/ug/getting-started-with-r-cnn-fast-r-cnn-and-faster-r-cnn.html

[27] O. Hmidani and E. M. Ismaili Alaoui, "A comprehensive survey of the R-CNN family for object detection," *2022 5th International Conference on Advanced Communication Technologies and Networking (CommNet),* Marrakech, Morocco, 2022, pp. 1-6, doi: 10.1109/CommNet56067.2022.9993862.

[28] JB. Woo and M. Lee, Comparison of tissue segmentation performance between 2D U-Net and 3D U-Net on brain MR Images. 2021. doi: 10.1109/iceic51217.2021.9369797.

[29] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” in Lecture Notes in Computer Science, Springer Science+Business Media, 2015, pp. 234–241. doi: 10.1007/978-3-319-24574-4_28.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", *International Conference on Learning Representations* (ICLR), 2015.

[31] P. Gudzius, O. Kurasova and E. Filatovas, "Optimal U-Net Architecture for Object Recognition Problems in Multispectral Satellite Imagery," 2019 *IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)*, Abu Dhabi, United Arab Emirates, 2019, pp. 1-2, doi: 10.1109/AICCSA47632.2019.9035305.

[32] I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, F. Hughes, D. Tuia, R. Raskar, "DeepGlobe 2018: A Challenge to Parse the Earth through Satellite Images". 2018. doi: 10.1109/cvprw.2018.00031.

[33] “Deep Learning Toolbox.” https://www.mathworks.com/products/deep-learning.html

[34] “Image Processing Toolbox Documentation.” https://www.mathworks.com/help/images/