ARCADA

Predicting Acid Sulphate Soils in Finland's Coastal Areas Using Deep Learning Fusion of Remote Sensing Map Tile Data

Md Golam Moctader

Master's Thesis

MEng in Big Data Analytics

2024

Master's Thesis

Md Golam Moctader

Predicting Acid Sulphate Soils in Finland's Coastal Areas Using Deep Learning Fusion of Remote Sensing Map Tile Data. .

Arcada University of Applied Sciences: MEng in Big Data Analytics, 2024.

Commissioned by:

Arcada University of Applied Sciences via PROFI 2 Project

Abstract:

Acid Sulfate (AS) soils represent a significant ecological risk and are a major environmental concern, particularly in Finland, where they are recognized as one of the country's most pressing environmental challenges. Traditional mapping methods for AS soils are labor-intensive and time-consuming, resulting in limited mapping efforts despite increasing awareness and efforts over the past decade. However, recent technological advancements have revolutionized soil data generation, providing new opportunities for AS soil classification through deep learning (DL) and machine learning (ML) techniques. This master's thesis endeavors to evaluate various DL methods for AS soil classification across a substantial portion of the Finnish landscape, with a specific focus on identifying effective strategies for mapping AS soils. Notably, convolutional neural networks (CNNs) are comprehensively examined in this regard. The study area covers the north-western coast of Finland, with a dedicated dataset tailored for DL analysis. Meanwhile, ML techniques consider only central pixel values. Obtained results highlight that scalar-input models and Random Forest (RF) emerge as highly effective methods for AS soil classification, demonstrating promising improvements in accuracy and efficiency throughout the conducted studies.

Keywords: Acid Sulfate, Non-Acid Sulfate, Convolutional Neural Network, Deep Learning, Machine Learning

Contents

Figures

Tables

Abbreviations

AS	Acid Sulfate
ANN	Artificial Neural Network
AUC	Area Under the Curve
BoVW	Bag-of-Visual-Words
CNNs	Convolutional Neural Networks
CRS	Coordinate Reference System
DL	Deep Learning
DSM	Digital Soil Mapping
ELM	Extreme Learning Machine
FPR	False Positive Rate
KLPP	Kernel Locality Preserving Projection
KDA	Kernel Discriminant Analysis
MSE	Mean Squared Error
ML	Machine Learning
OA	Overall Accuracy Assessment
PCA	Principal Component Analysis
PO	Point Observation
RBF	Radial Basis Function
RBFLN	Radial Basis Function Learning Network
ReLU	Rectified Linear Units
ROC	Receiver Operating Characteristic
RF	Random Forest
SLFN	single-hidden-layer feedforward neural network
SOC	Soil Organic Carbon
SRKLPP	Spectral Regression KLPP
SRKDA	Spectral Regression KDA
SVM	Support Vector Machine
TPR	True Positive Rate
UCML	Unsupervised Clustering for Multi-label
VHR	Very High-Resolution

Foreword

This thesis paper presents research on soil science, environmental issues, and technological advancements. It represents the outcomes of tireless efforts and passionate dedication to understanding and solving pressing problems related to AS soils. The following pages will present a comprehensive study of the classification and mapping of AS soils.

This work is not merely an academic exercise; it is a call to action to confront environmental threats with innovation and determination. Armed with the methodology, analysis, and results presented here, let us continue our efforts to protect the environment and implement sustainable practices.

I express my sincere gratitude to everyone who has contributed to this effort, from mentors Lecturer Anton Akusok, PhD, and Principal Lecturer Leonardo Espinosa Leal, PhD, to colleagues, friends, and family. The support and encouragement have been invaluable and have helped me navigate the ups and downs of this journey. Finally, I would like to thank the Geological Survey of Finland (GTK, https://www.gtk.fi/en/) for the provided point observation data for acid sulfate soils.

I hope this paper will serve as a beacon of knowledge, inspiration, and hope, illuminating the path to a greener and healthier future for future generations.

Md Golam Moctader

1. Introduction

This thesis explores classifying Acid Sulfate (AS) soils of the north-western coast of Finland using soil science and Artificial Intelligence (AI). The impact of AS soils extends far beyond Finland, contributing to wider environmental problems worldwide. AS soils, can produce acids when drained or disturbed, providing serious challenges to agriculture, ecosystems, and water quality worldwide. Fig 1 illustrates the potential acid sensitivity of surface waters across Europe (Austnes et al. 2018). In Finland, the classification of AS soils is very important due to its unique environmental and geological features. Finland has extensive wetlands, coastal areas, and peatlands that are vulnerable to AS soil formation and associated environmental impacts. The presence of AS soils in the region creates challenges to agricultural, forestry, and infrastructure development. Moreover, the extensive release of sulfuric acid and heavy metals from AS soils could pose a threat to freshwater ecosystems, including lakes, rivers, and coastal areas, which are an important part of Finland's environmental and cultural heritage. This study consists of an image of a tiles dataset ranging from 2 to 15 zoom levels of the northwestern coast of Finland, with each zoom level comprising 10 covariate layers. Fig. 2 shows point observations (POs) from AS soils in red and potential occurrences of non-AS soils POs in blue within the Landsat covariate layer map for this research. To address these problems of AS soil classification, this study explores the use of lightweight deep learning (DL) models for potential mobile applications. These models offer efficient processing and can effectively classify AS soil. The obtained results are compared with the baseline Machine Learning (ML) models Random Forest(RF) and Extreme Machine Learning (ELM). This approach will not only improve the accessibility and usability of AS soil classification tools, but also enable stakeholders, including land users, policymakers, and citizens, to contribute to informed decision-making and environmental protection.

1.1 Acid Sulfate Soils

Nowadays, soil science plays a vital role in numerous sectors such as agriculture, energy, and infrastructure, emphasizing the importance of studying soils within their natural environments. Among the various types of soils, AS soils stand out, primarily found in coastal and wetland regions worldwide. These soils raise significant environmental concerns due

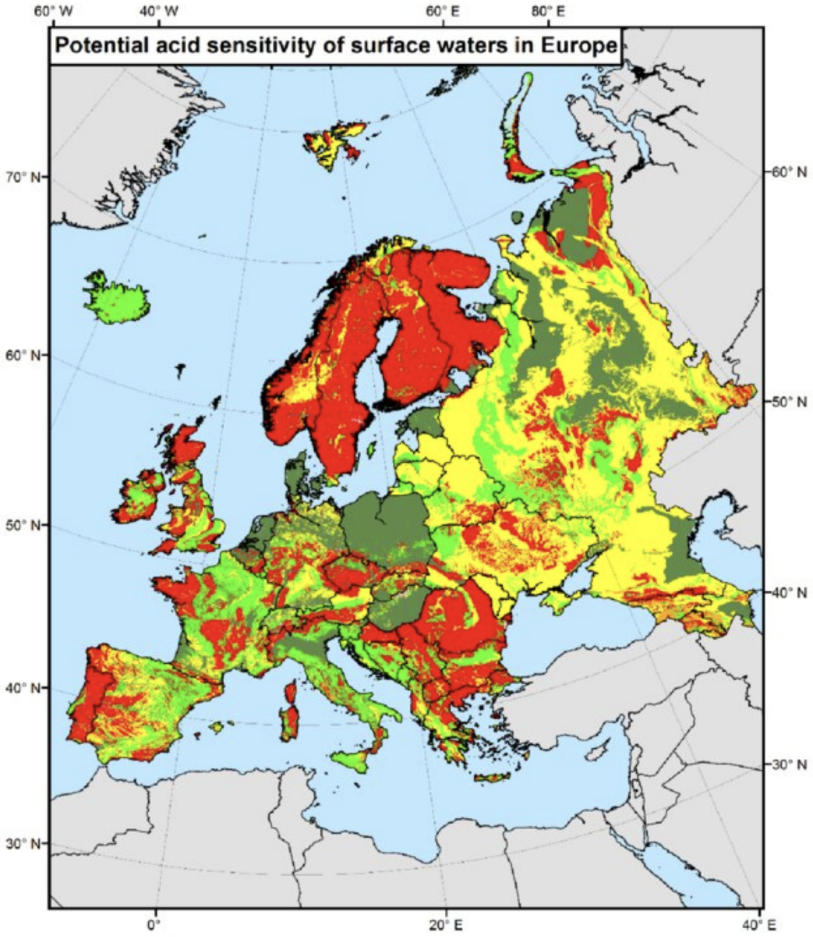

Figure 1. Illustrates the potential acid sensitivity of surface waters across Europe. The color scheme indicates the degree of acid sensitivity: red represents high sensitivity, yellow denotes medium sensitivity, and green signifies low confidence in low sensitivity (Austnes et al. 2018).

to their potential ecological impact. Notably, Finland exhibits the highest concentration of AS soils in Europe, covering an extensive area of approximately 1600-3000 km² along its coastal regions (Fältmarsch et al. 2008). This concentration of AS soils in Finland underscores the country's environmental challenges and highlights the need for effective management strategies to mitigate their impact. In Finland, during the Holocene period AS soils originated at the bottom of the Baltic Sea due to lacking of oxygen. As the land rose, these soils emerged above sea level, becoming a concern. These soils contain high levels of metal sulfides, which react with oxygen to form sulfuric acid, causing soil acidification and metal pollution.

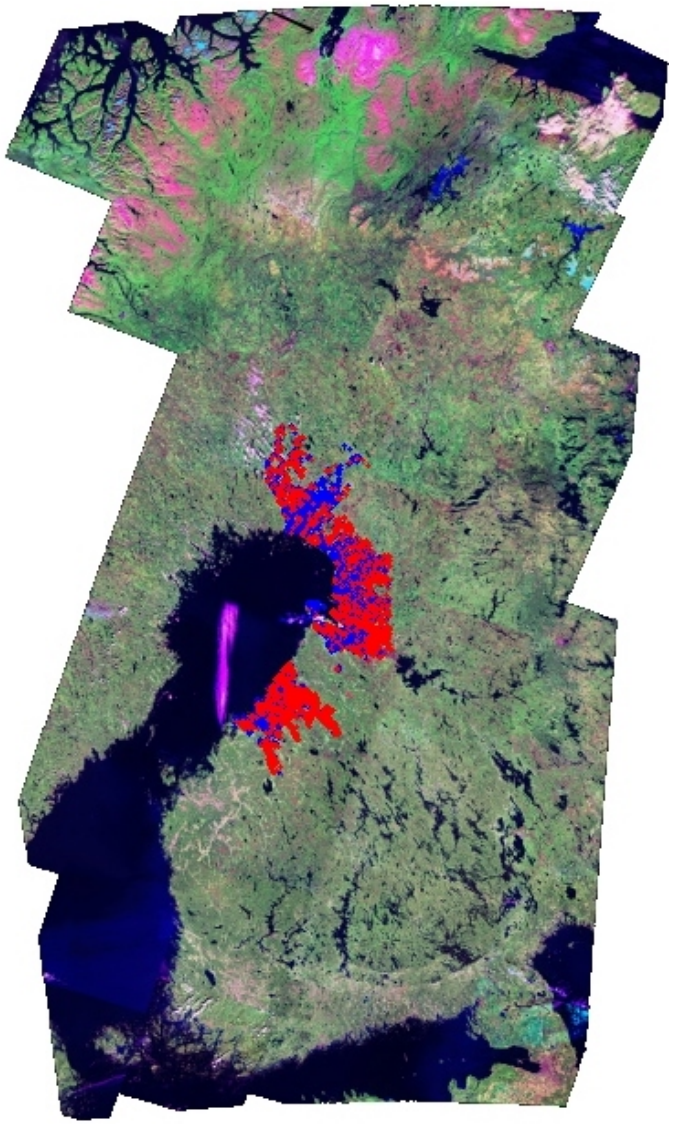

Figure 2. The figure depicts the AS soils point observation (POs) marked in red and the non-AS soils POs Potential Occurrence marked in blue within the Landsat covariate layer map for this research.

1.2 Artificial Intelligence

AI is a rapidly evolving field focused on creating computer systems that can able to perform tasks that basically needs human intelligence. Through AI, machines can analyze data, recognize patterns, and make decisions, mimicking human cognitive abilities. AI technology is already prevalent in various aspects of daily life. In soil science, ML and DL techniques are increasingly employed to classify soil properties and optimize agricultural practices. These technologies enable researchers to extract valuable insights from large datasets, leading to more efficient soil management strategies and sustainable farming practices. As AI continues to advance, it holds the promise of revolutionizing in-

dustries such as healthcare, transportation, and finance, offering solutions to complex problems and improving efficiency and decision-making processes (McCracken & Cate 1986).

1.3 Research questions

In previous studies, several DL techniques have been employed for AS soil mapping, but there are still many architectures that have not been utilized, which may be more suitable. Utilizing more appropriate techniques could enhance the classification accuracy in AS soil mapping. Therefore, the main inquiries of this study are:

1. What data are best suited for classifying AS soils, and what motivates the preference for lightweight DL models over heavier architectures in this classification task?

2. Can different lightweight DL methods accurately classify AS soils, and how effective are these new methods for AS soil classification?

1.4 Structure of the thesis

Apart from the introduction, this study comprises five additional chapters. The forthcoming chapter will focus on related work, providing a comprehensive overview of existing research in the field. Following that, the methodology chapter will offer a detailed description of the methods utilized in this study. The various DL techniques analyzed in this research will be outlined within the third chapter. Moreover, in experiments chapter, a thorough explanation of the dataset creation process required for the CNN will be provided. The fifth chapter will present the results obtained from the different methods employed and their comparative analysis. Lastly, the final chapter will be dedicated to summarizing the findings and drawing conclusions from this research work.

2. Related Work

2.1 DL for Soil Predictions with Map Data

In recent studies, several researchers have employed artificial intelligence methods for predicting soil properties. For instance, (Beucher et al. 2013) selected Radial Basis Functional Link Nets (RBFLN) as their method of choice for mapping AS soils, given its demonstrated effectiveness in handling large geographical areas and addressing issues with slow traditional soil sampling. RBFLN, an Artificial Neural Network (ANN), was a novel choice for AS soil mapping. The shift from traditional methods in Finland was prompted by the recognition that traditional soil sampling is time-consuming. RBFLN, with its spatial modeling techniques, offered a faster and more objective alternative based on geostatistics.

The configuration of RBFLN involves using training points with both AS soil occurrences and non-AS soil sites to make accurate predictions. For modeling, only the imaginary components and the electromagnetic 3 kHz data were used because they provided relevant information to identify AS soils. This study used the metrics of area coverage in different zones where AS soil might occur and compared predictions with known places using validation points. The most accurate prediction covered 23 percent of the area and included 94 percent of the known AS soil spots. The success of RBFLN in various applications made it a suitable choice for AS soil mapping, despite acknowledged limitations like the number of data layers, prompting a suggestion for future improvements. RBFLN, aims for efficiency and accuracy in mapping AS soils. Insights from earlier studies, especially Palko (1996) findings, influenced the adoption of RBFLN as a more effective mapping strategy compared to traditional approaches (Beucher et al. 2013).

Bierwirth & Brodie (2005) study describes the specific use of airborne gamma-radiometric data, as well as different datasets within a stepwise classification model to create a Geographic Information System (GIS) based map of surface soil acidity (Bierwirth & Brodie 2005). To generate a unified atlas of Australian AS soils, (Fitzpatrick et al. 2008) developed a methodology involving the collation and assembling of all existing AS soil data and various datasets within GIS. In Finland, AS soil mapping was only conducted

by conventional soil sampling and subsequent soil-pH measurements until now (Palko 1996). These studies are referenced to provide context, and Palko's study, highlighting issues with traditional soil sampling, influenced the decision to explore RBFLN as an alternative.

In the study on the Sirppujoki River basin in southwestern Finland, Beucher et al. (2015) chose to integrate ANN techniques, specifically the Radial Basis Function (RBF)-based ANN implemented using RSNNS. They made this choice to predict AS soil properties by incorporating 1:20,000-scale digital maps, LiDAR data, and low electromagnetic frequency components from airborne geophysics. The use of this ANN technique was motivated by its ability to effectively capture the complexity of the region's soil properties, providing valuable insights. The researchers selected this method because it outperformed the RBFLN approach (Beucher et al. 2013), due to its adaptability to the diverse data sources and the specific characteristics of the Sirppujoki River basin. The configuration of the ANN involved leveraging the RBF architecture, demonstrating its suitability for the study's objectives. The authors opted for this method to enhance predictive accuracy and gain a comprehensive understanding of the soil properties in the region (Beucher et al. 2015).

Beucher et al. (2017) extends previous efforts in mapping AS soils, with a particular focus on wetlands in Jutland, Denmark. In Denmark, regulations mandate approval from environmental authorities before draining areas identified as potential AS soils. The 1980s mapping of these soils concentrated on wetlands, utilizing conventional methods like soil sampling and pH determination. Acknowledging the resource-intensive nature of traditional soil mapping, the research explores alternative spatial modeling techniques within the Digital Soil Mapping (DSM) framework. Previous studies have examined various techniques for mapping AS soils, such as the fuzzy k-means algorithm in Australia (Huang et al. 2014) and fuzzy logic and ANN techniques in Finland Beucher et al. (2013), (Beucher et al. 2014), (Beucher et al. 2015). Beucher et al. (2017) choice of the ANN method for this study stems from its efficiency in handling large datasets and its frequent application in DSM for predicting soil attributes or classes. The primary objective is to evaluate the predictive classification capabilities of the ANN technique in mapping poten-

tial AS soils in the wetlands of Jutland, Denmark. The wetlands of Jutland, constituting 20 percent of the region, undergo a predictive study of AS potential. Employing a DSM approach based on the Scopan model and an RBF-based ANN, the study achieves an impressive Overall Accuracy Assessment (OA) of 70 percent. The integrated approach, utilizing 16 environmental variables, LiDAR-based surface parameters, and categorical predictors, demonstrates promising predictive classification performance (Beucher et al. 2017).

The study in Chile aimed to find and explain the choices made in their study regarding soil predictions. They specifically chose CNNs due to their effectiveness in handling large sets of information. This study highlights the limitations of other methods and emphasizes CNNs' flexibility and efficiency. In refining their approach, the study took steps to tackle overfitting by introducing a 3-D array rotation technique, adding robustness to their dataset. The dataset used in this study contains Chilean soil information. The target features chosen for analysis are the soil organic carbon (SOC) content (percent) across different depths, namely 0–5, 5–15, 15–30, 30–60, and 60–100 cm. A total of 485 soil profiles were included so that it includes all profile observations across all depth intervals. utilizing CNNs, significantly outperformed the Cubist model in predicting SOC content across different depth ranges (Padarian et al. 2017). The error reductions of 23.0, 23.8, 26.9, 35.8, and 39.8 percent for depth intervals of 0–5 cm, 5–15 cm, 15–30 cm, 30–60 cm, and 60–100 cm, respectively. The proposed CNN approach achieved error reductions of 24.8, 24.7, 28.5, 28.6, and 23.5 percent for the same depth ranges. Overall, the CNN model demonstrated superior performance throughout the soil profile (Padarian et al. 2019).

Alexandre et al. (2019) used CNN for soil mapping to address the challenge of handling uncertainties in measurements. The CNN is chosen for its effectiveness in integrating uncertainty explicitly, a crucial aspect often overlooked by traditional methods. The decision is influenced by the success demonstrated in previous studies by (Padarian et al. 2019) and (Behrens et al. 2018), where CNNs proved effective in soil mapping, especially in handling contextual covariate information. This proven success becomes a pivotal factor, influencing the preference for CNN over other methods that might have shown success in

specific contexts but lacked the demonstrated effectiveness in handling complex mapping scenarios. Additionally, the CNN's capability to predict soil properties at various depths within a single framework enhances flexibility and efficiency in the mapping process. This advantage sets CNN apart from conventional methods, which often struggle to offer similar flexibility without resorting to separate models or extensive adjustments. Configuring the CNN involved implementing shared layers for a common architecture between topsoil and subsoil, followed by separate sets of fully connected layers for each soil depth. The model effectively handled missing values while maintaining the original size of input images through consistent zero padding. To ensure fair comparisons with RF models, the study emphasized using the same calibration and test sets with normalized measurements and standardized covariates. The CNN employs filters to detect features in nearby areas of images and integrates them to make predictions. Trained on soil carbon levels and environmental factors are elevation and vegetation, the CNN achieved an RMSE of 0.93 for topsoil and 1.08 for subsoil. While displaying smaller RMSE values than RF, but demonstrated a stronger correlation.

2.2 Unsupervised Pre-training of DL Models for Tasks with Limited Labeled Data

Chaib et al. (2017) presents two significant contributions in the domain of very high-resolution (VHR) image scene classification. The methodology involves extracting the first two fully connected layers from both Caffe-Net (Bahrampour et al. 2016) and VGG-Net (Majib et al. 2021) for each image scene. Subsequently, a fusion approach, combining these layers through concatenation and addition. The resulting deep feature descriptor provides a comprehensive and nuanced representation of the image scene, enabling to achieve of high classification accuracy using a Support Vector Machine (SVM) classifier. Through extensive experiments, the proposed method consistently outperforms state-of-the-art techniques with and accuracy of 96.88 percent, underscoring its efficacy in VHR image scene classification. The dataset used is referred to as the UC Merced dataset (Yang & Newsam 2010). This dataset consists of aerial scene images representing different categories. Each image in the dataset has dimensions of 256×256 pixels.

This study Ge et al. (2018), showed pre-trained CNNs like AlexNet (Yuan & Zhang 2016),

VGGM (Zhang et al. 2024), VGG16 (Theckedath & Sedamkar 2020), and GoogLeNet (Tang et al. 2017) originally designed for image classification to enhance high-resolution remote sensing image retrieval. To leverage distinct characteristics between high-level and mid-level layers, employed a direct extraction method for high-level features and an aggregation approach for mid-level features. These experiments revealed that CNN features generally outperform shallow features across most image classes. Notably, the stability of high-level features in VGGM (Shao et al. 2018) and the effectiveness of average pooling for aggregating features, especially in VGG16 and GoogLeNet, were observed. The proposed extraction methods are both simple and effective, demonstrating consistent performance even on large-scale image datasets. The experiments utilize three datasets comprising high-resolution remote sensing images: the UC Merced Land Use (Yang & Newsam 2010), the WHU-RS (Xia et al. 2010), and the PatternNet dataset (Zhou et al. 2018). Preliminary efforts explored feature combinations and compression techniques to address specific CNN feature shortcomings, significantly enhancing retrieval performance. Feature combinations, particularly with Bag-of-Visual-Words (BoVW), proved beneficial, and most compressed high-level features performed well with a reduced dimensionality to 32 dimensions.

This study employs a two-stage classification approach, Marmanis et al. (2015) utilizied the model name Overfeat which is an enhanced version of AlexNet pretrained on the ImageNet dataset (Agarwal & Shekhawat 2022). Overfeat's deep feature generation and classification capabilities are harnessed for processing Unsupervised Clustering for Multi-label (UCML) training data, extracting preactivations from the seventh and eighth hidden layers. The resulting 2-D features, downsampled to match Overfeat's input dimensions, serve as inputs for the subsequent trainable CNN stage. The introduced trainable CNN, comprising two convolutional layers, two fully connected layers, and a Softmax classifier, plays a crucial role in achieving high accuracy by learning complex patterns and effectively separating data into different classes. The CNN's adaptability to the spatial characteristics of 2-D input data is instrumental in this classification scheme. Evaluation over the open-source publicly available aerial image dataset UCML benchmark showcases the two-stage classifier's state-of-the-art results, with a notable 5.5 improvement over alternative architectures. Omitting the pre-trained stage emphasizes its crucial role, and although

CNN training time is extended, both CNN and RF classifiers perform equally fast during testing. Confusion matrix analysis highlights effective land-use class detection, and t-SNE analysis reveals the positive influence of pre-trained features on initial class separation. Model transferability is validated over new data from diverse cities, demonstrating accurate classification even in significantly different scenes (Marmanis et al. 2015).

The proposed method by Chen et al. (2018) involves three key stages: salient region extraction using the JUDD visual saliency model, unsupervised feature learning with sparse filters, and CNN fine-tuning. Initial salient regions from 15 high-resolution Google Earth images create 32,954 patches. For labeled datasets (UC Merced and AID), 80 training and 20 testing, with five-fold cross-validation, ensure comparability. The CNN functions as a feature extractor, and a linear SVM acts as the classifier, reporting mean accuracy, standard deviation, and confusion matrix. Experimental results on UC Merced and AID showcase a 2.24 accuracy improvement over Plain-CNN, demonstrating the method's effectiveness in enhancing CNN performance on large-scale unlabeled satellite imagery for applications like SAR or infrared imagery classification (Chen et al. 2018)

This study addresses challenges in remote sensing scene classification by leveraging pre-trained deep CNN models. The proposed method introduces a feature fusion approach and utilizes multilayer features from a single pre-trained CNN model. This study utilized two authentic high-resolution scene image datasets to assess the proposed methodology. The first dataset UC Merced (UCM), was manually curated from extensive aerial ortho imagery, boasting a pixel resolution of one foot (Yang & Newsam 2010) comprising 21 scene-level land use classifications, each category contains 100 images sized at $256 \times 256 \times 3$ pixels. The second dataset, WHU-RS, was compiled from satellite images sourced from Google Earth (Xia et al. 2010). It encompasses 1005 scene images categorized into 19 classes, with each class comprising approximately 50 images sized at 600×600 pixels and featuring three channels. Experimental results on two datasets demonstrate a classification accuracy of 99.05 percent for UCM and 99.22 percent for WHU-RS. These findings underscore the method's robustness and superiority over traditional scene classification methods, showcasing the effectiveness of pre-trained deep CNN models in remote sensing applications with limited training samples (Du et al. 2018).

3. Research Methodology

3.1 Dataset

In this master's thesis, a comprehensive geospatial dataset is employed for DSM to predict and classify soil types. The methodology integrates POs and covariates. The dataset's structure mirrors a zooming process, resembling a layered narrative shown in Fig. 3 Initially, it covers a broad overview, with each zoom level providing specific information. As progress through the zoom levels, the tiles subdivide into smaller components, offering increasingly detailed insights.

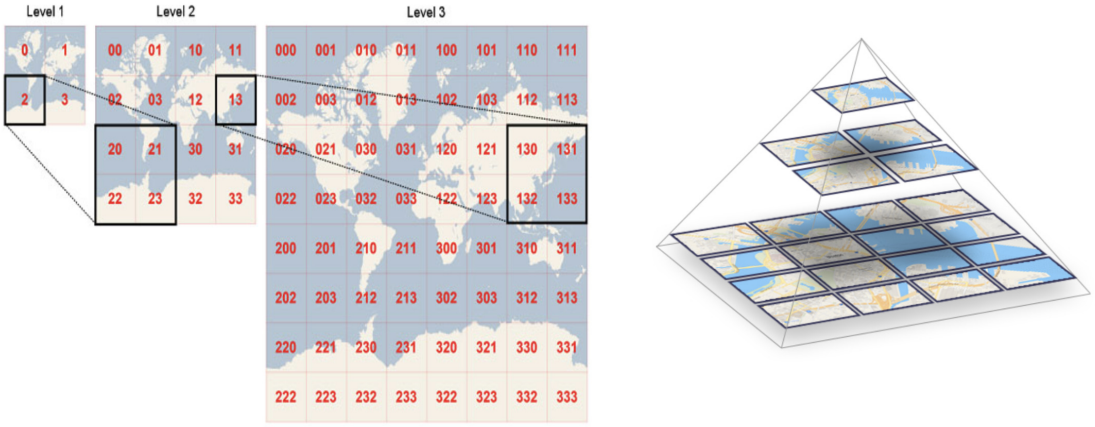

Figure 3. Figure depicts how covariate layers are stored as map tiles (MapTiler 2022), (Microsoft 2022)
.

Given the spatial nature of the data, all covariate layers must share the same coordinate reference system (CRS), which, in this case, is the World Geodetic System (WGS84) in Finland (Decker 1986). The POs also adhere to the same CRS.

3.1.1 Point Observations

The POs refer to soil samples collected by geologists. These sampling instance records are essential for selecting locations on the covariate layers. The Geological Survey of Finland (GTK) has provided this data. Table 1 presents some PO, including spatial coordinates and the classification indicating whether the PO is AS soil or not. The criteria for categorizing AS soils are influenced by various factors, and in Finland, these criteria have been recently updated (Boman et al. 2020).

Table 1. Point observation

No.	xcoord	ycoord	soil type
1	441468.7125	7184281.005	ASS
2	441468.7125	7186002.327	ASS
3	442236.5012	7185249.359	ASS
4	420520.3581	7208257.978	ASS
5	365454.8769	7095720.274	non-ASS
6	364265.9429	7094588.831	non-ASS
7	360325.2436	7091279.457	non-ASS

3.1.2 Covariates Layers

The environmental covariate layers consist of a raster dataset typically derived from remote sensing data. There are ten covariate layers is shown in Fig. 4, comprising five different types: (aem-apparent-resistivity, aem-imaginary-component, aem-real-component) representing Electromagnetic (EM) data, (corine-land-cover) representing Land Cover, (elev-10m, elev-10m-aspect, elev-10m-hillshade, elev-10m-slope) representing elevation, (Landsat) representing Satellite Imagery, and (TPI - Topographic Position Index) representing Topographic Index.

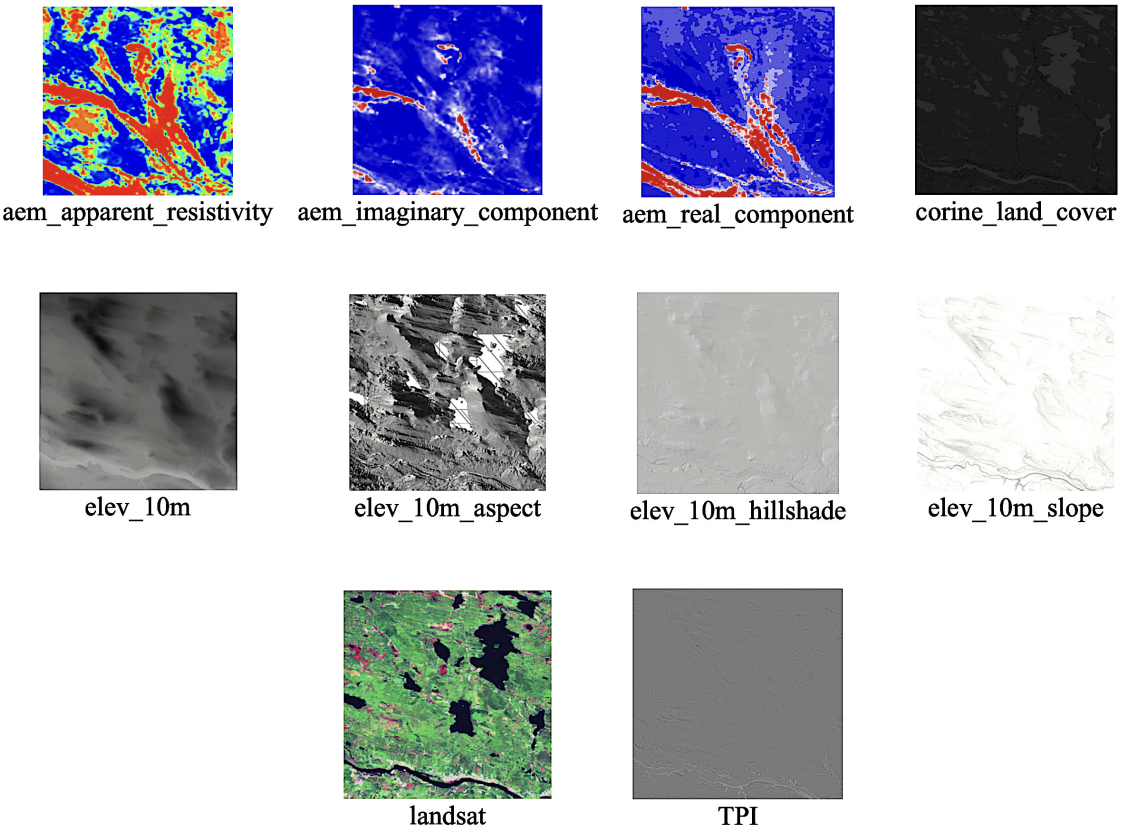

Figure 4. Figure depicts the tiles of the ten different covariate layers.

Given the spatial nature of the data, all covariate layers share the same coordinate reference system (CRS), which, is WGS84 in Finland. The POs also adhere to the same CRS.

3.1.3 Relationship between POs and Covariates Layers

To classify AS soils, it's important to link the PO with environmental covariate layers. This linkage offers an understanding of the characteristic attributes of each covariate layer across both soil classes. In this study, crucial information extracted from the PO comprises spatial coordinates and soil classification (AS or non-AS).

3.2 Machine Learning Fundamentals

Two models have been used to classify AS soils. These are RF and ELM. Here is a brief recap of each model and how they work.

3.2.1 Random Forest

RF is an ensemble learning framework that constructs multiple decision trees during training by drawing a subset of training samples with replacement, employing a bagging approach. A decision tree is like a flowchart that helps make decisions based on input data. It divides the data into smaller groups based on features, aiming to make each group as similar as possible. It keeps dividing until it reaches a point where it can't split anymore or has reached a set limit. In this process, about two-thirds of the samples are utilized to train the trees, while the remaining one-third is reserved for internal cross-validation, estimating the out-of-bag (OOB) error. The final classification decision is determined by averaging the class assignment probabilities calculated by all produced trees. Key parameters, such as the number of decision trees (Ntree) and the number of variables tested for the best split (Mtry), play crucial roles. Typically, Ntree is set to 500, and Mtry is set to the square root of the number of input variables. Known for its computational efficiency, RF is widely employed in various software packages, offering features like variable importance analysis. The computing time required for establishing the RF classification model is influenced by the number of trees (T), variables used in each split (M), and the number of training samples (N). Overall, RF stands out as a powerful and flexible classification method, suitable for diverse applications (Belgiu & Drăguţ 2016).

3.2.2 Extreme Learning Machine

The Extreme Learning Machine (ELM) algorithm is based on a single-hidden-layer feed-forward neural network (SLFN). It utilizes a dataset with sample characteristics x_i and labels y_i to construct a network with hidden nodes and an activation function. The SLFN's mathematical model involves output weights (beta-i), randomly generated input weights w_i, biases b_i, and the activation function (g). The algorithm aims to find optimal parameters to minimize the difference between the hidden layer output matrix (H) and the labels (Y). The least-squares solution yields output weights (beta) calculated as the Moore-Penrose generalized inverse of the matrix H. In classification, the algorithm involves randomly setting input weights and biases, training the ELM classifier to obtain output weights, and then classifying unknown samples based on the learned parameters. The process includes extracting spectral features, training the classifier, and testing to output prediction labels for image classification (Huang et al. 2019).

3.3 Deep Learning Model Development

3.3.1 Deep Learning

DL is the subfield of ML that uses algorithms inspired by the structure and the functions of the brain's neural network (Ertam & Aydın 2017). So, mainly DL is talking about the algorithms that learn from data. This learning occurs either in a supervised or unsupervised form. Neural networks establish an especially useful methodology in DL which permits to gain proficiency with obscure normality for a given set of training data (Hammer et al. 2004).

When the data in the training set is labeled it's called Supervised Learning. Algorithms resolve one to numerous irregularities by averaging over various targets (Jordan & Rumelhart 2013). In the case of supervised learning, both the training and validation data are passed through a model that is labeled. Each piece of the data passing through the model during training is a pair that consists of the input object or sample along with the corresponding label or output value. Necessarily with supervised learning, the model is learning to create a mapping from given inputs to outputs based on learning from the labeled training data (Carpenter et al. 1992).

Unsupervised learning deals with data without having labels. One reason for unsupervised learning might be useful is that unlabeled data is abundant and cheap since finding a ground truth level for each sample is usually a step that takes a lot of time. Another use of unsupervised learning is to learn representation. If there is a large amount of unlabeled data and not enough labeled data, it is possible to pre-train a model on the unlabeled data first. Encoder part of the trained model used as the feature extractor (Hinton & Sejnowski 1999).

Clustering and dimensionality reduction are the most common form of unsupervised learning. The dimensionality reduction mainly reduces the number of variables, which are required to represent data. One reason for dimensionality reduction is that the volume of the feature space increases exponentially as the number of variables increases. In higher dimensional spaces, it's harder to have representative samples for every possible configuration of the parameters, which is known as the curse of dimensionality. In reality, natural data has regularities and doesn't use an entire parameter space. Natural images are created by a small fraction of each possible combination of the pixel values. But In DL, dimensionality reduction is usually not done in separate steps. However, the models learn how to back representation within the model. In the unsupervised fashion, it is required to define some sort of target to train a neural network (Pena et al. 2001).

The model tries to reconstruct its output image similar to the input image is called the autoencoder. Since the input and output are the same. A typical autoencoder is fed forward neural networks that consist of an encoder and decoder model with a bottleneck in between. The encoder model diesels the information to the lower information feature. It's called code, and the decoder tries to reconstruct its input using the code. The inputs and the outputs don't have to be identical; for example, the input can be a noisy restoration of the output. This noise can be added artificially. In this setup, networks try to denoise input (Chen et al. 2017). Another type of unsupervised that is worth mentioning is generative adversarial networks. It is one of the recent breakthroughs in the field (Bousmalis et al. 2017).

When a DL model learns and makes inferences from the data that has already been labeled

is supervised learning. In unsupervised learning, the model learns from the unlabeled data. A little blurry portion can be observed in the boundary between supervised and unsupervised learning. This kind of task can be supervised, unsupervised, and something in between depending on the prospectives.

3.3.2 Neural Networks

Drawing inspiration from the intricate human neuronal synapse system, the methodology incorporates the ANN, an ML technique introduced in the 1950s. In its early stages, the potential of ANN to address real-world problems was hindered by limitations such as computational power constraints and insufficient training data. Challenges, including gradient and overfitting issues in training deep neural architectures, further impeded progress. However, recent advancements have reshaped this landscape, with the advent of big data, enhanced computational capabilities through graphics processing units (GPUs), and the development of novel algorithms. These factors have collectively played a pivotal role in overcoming historical barriers, enabling the effective training of deep neural networks. Notably, the methodology underscores the transformative impact of these technological advancements, highlighting instances where such systems outperform human capabilities in visual and auditory recognition studies (Lee et al. 2017).

3.3.3 Convolutional Neural Networks (CNNs)

The CNN, a computational network consisting of multiple neurons arranged in a step-by-step minimal processing fashion, distinguishes itself from ANN by featuring a structure where only the last layer is fully connected (Lee et al. 2017). Renowned for its effectiveness in image analysis, CNN has expanded its scope to diverse data analysis and classification problems. Its primary application lies in pattern detection within input images, facilitated by hidden layers, known as convolutional layers, that transform inputs into outputs through convolution operations.

Each convolutional layer is endowed with specific filters tasked with detecting distinct patterns like edges or circles. The network's depth contributes to the complexity of filters, empowering deeper layers to discern more sophisticated objects. The precise specification of the number of filters in each convolutional layer is paramount, as they convolve with the inputs, generating outputs for subsequent layers shown in Fig. 5. Recent years have

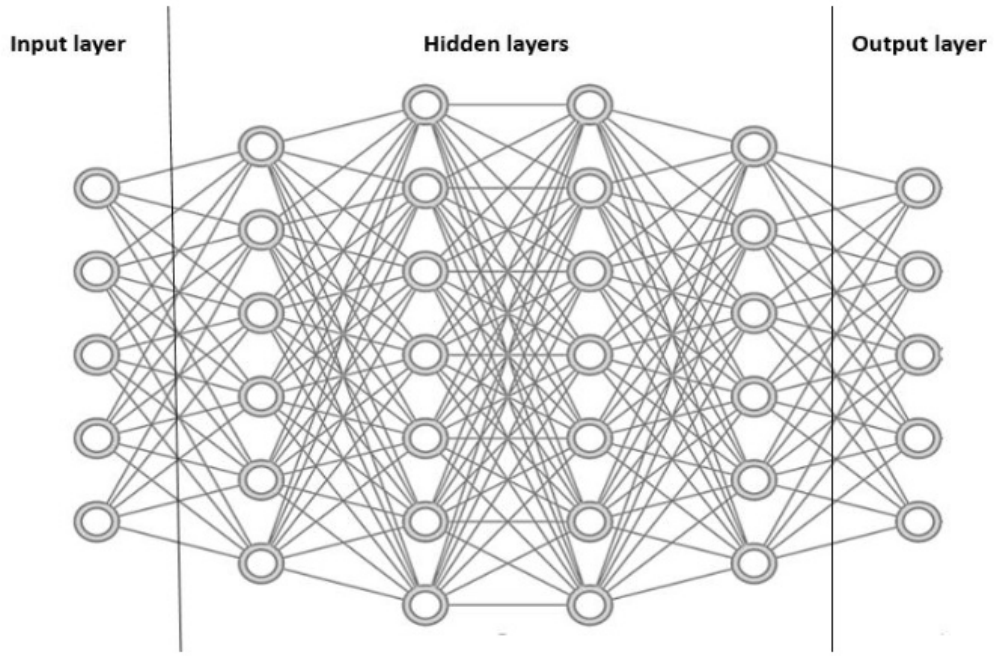

Figure 5. Sketch of a Neural Network, where three distinct layers (input, hidden, and output) are shown in the figure. The input layer provides a set of weights to the hidden layer and produces the output layer (Touretzky & Pomerleau 1989)

.

witnessed the integration of DL into remote sensing applications, presenting promising results. While some techniques involve 2D neural network methods, there is a prevalent use of patch-based methods, which, though requiring more training time, demonstrate their effectiveness in various applications.

3.4 Evaluation Metrics

In the evaluation of each model, a diverse set of metrics has been employed, including recall, precision, f1-score, and ROC (AUC). Utilizing multiple metrics is crucial as they capture distinct facets of model performance. Precision, which denotes the proportion of correctly classified positive elements among all elements classified as positive, may vary based on the chosen positive class. This variability necessitates the use of multiple precision values.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{1}$$

In the concept of classification, The key four concepts are true Positives, True Negatives, False Positives, and False Negatives. A True Positive denotes the model's ability to correctly identify a positive instance as positive. Conversely, a True Negative signifies the model's accurate recognition of a negative instance, mirroring the truth. False Positives, denotes instances where the model incorrectly labels a negative instance as positive. Similarly, False Negatives rise when the model erroneously identifies a positive instance as negative.

For instance, a model achieving perfect recall may still exhibit poor behavior if it simply assigns all inputs to a single class. Recall addresses the concern of false negatives by emphasizing the correct classification of positive elements.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{2}$$

The f1-score, a composite metric that combines precision and recall, provides a balanced measure of a model's overall performance.

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

The ROC curve is formed by calculating the true positive rate (TPR) against the false positive rate (FPR) across different threshold levels. The Area Under the Curve (AUC), provides the classifier's ability to distinguish between classes across the entire range of thresholds.

$$\text{ROC (AUC)} = \int_0^1 \text{TPR}(t), d\text{FPR}(t) \tag{4}$$

Moreover, weighted averages of these metrics, such as global recall, can be computed, taking into account the respective class ratios. The confusion matrix, derived from the raw output, offers additional insights into the model's performance (Lorent 2019).

4. Experiments

4.1 Preprocessing

To prepare the dataset in the context of binary classification between AS and non-AS soils, I had to handle covariate layers, which are essentially raster images containing pixel information. As each covariate layer provides data for both classes, I needed to create a dedicated image dataset for the CNN. This involved a series of coordinate transformations on geographical points. Initially, the longitude and latitude were extracted from the geometry of each point, and the sine of the latitude in radians was calculated to ensure it stayed within a valid range. The geographical coordinates are then converted into pixel values (x and y) based on a tile size of 256, and scaling is applied according to the specified zoom level. The resulting scaled values are used to calculate tile coordinates (tx and ty) by dividing them by the tile size. Additionally, pixel offsets within the tile (px and py) were determined. Subsequently, based on the tile coordinates, generated a list of neighboring tiles within a square region centered around the given tile. This step involved calculating the coordinates of each neighboring tile and adding them to the list. The resulting list contained the coordinates of all the tiles within the defined radius around the original tile.

To facilitate the CNN's input layer requirements, a code segment is implemented that combines smaller tiles from neighboring locations into a comprehensive map image. This involved iterating through each adjacent tile, adding its image to a collection if the corresponding tile found. Assuming uniform dimensions for all tiles, the code determined their size. It then created a new image three times larger than a single tile and strategically placed each tile image within this larger canvas, considering the central tile. The code calculated central coordinates based on pixel offsets within the tile and set a crop size of 50x50 px around the central pixel for potential subsequent use as shown in Fig. 6. This entire process is crucial in creating a cohesive and detailed composite image, playing a pivotal role in this study for further analysis and visualization within a mapping context.

Afterward, The data frame for further analysis is constructed through a series of steps.

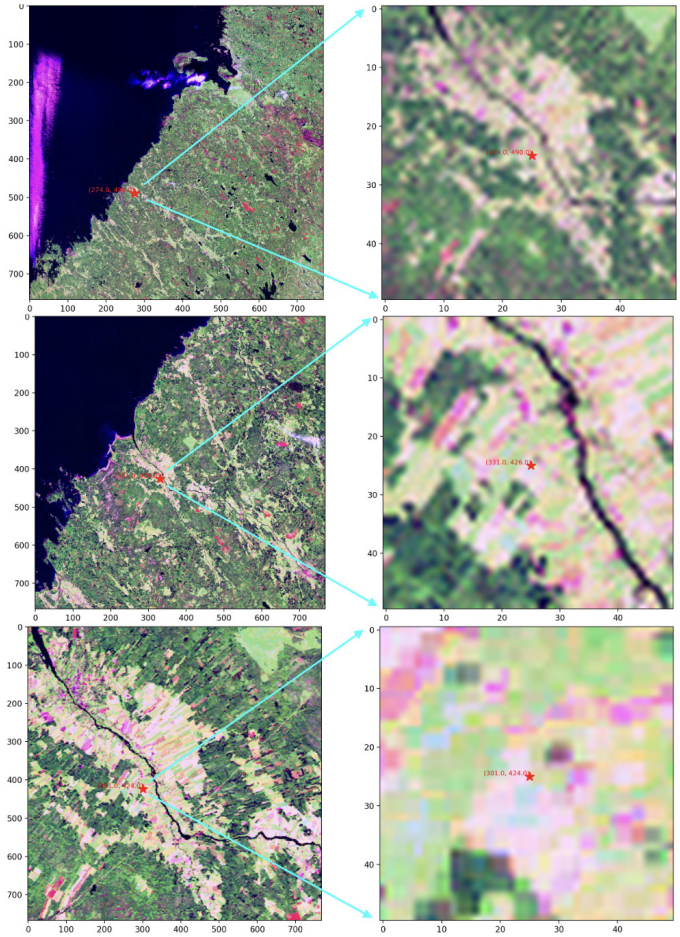

Figure 6. The figure depicts 50x50 pixel images cropped for the PO within this covariate layers tiles of Landsat for zoom levels 8, 10, and 12, arranged from top to bottom.

Initially, the path to the dataset and the desired zoom level are specified. Following this, data from a CSV file containing geographic points is imported into a GeoDataFrame. This GeoDataFrame is then utilized to create a new GeoDataFrame specifically tailored for the analysis. Each sample within this new data frame is characterized by its class, coordinates, and associated filenames. Latitude and longitude values are extracted from the GeoDataFrame to provide spatial context. To facilitate image processing, image files corresponding to the samples are loaded into memory using a custom function. Multi-channel images are formed by merging individual layers sourced from various directories. A binary label is assigned to each sample based on its class, aiding in classification tasks. Rows containing any missing values are subsequently eliminated to ensure data integrity. Finally, the processed data frame is saved as a pickle file, preserving its structure and content for future analysis and modeling endeavors.

4.2 Models

This study utilizes a diverse range of libraries, each meticulously chosen with specific versions tailored to the development and execution of the DL and ML models. Anchored by foundational frameworks TensorFlow (2.16.1) and Keras (3.3.2), alongside specialized tools like Pandas (2.2.2) and Scikit-learn (1.4.2), these libraries provide the essential infrastructure for constructing, training, and evaluating various model architectures. Additionally, the inclusion of visualization libraries such as Matplotlib (3.8.4) and Seaborn (0.13.2) enhances the interpretability of findings, aiding in model performance assessment and understanding of data distributions. Furthermore, Python (3.11.4) serves as the primary programming language. The model training is conducted within the built-in environment of the M1 Mac. Generally, each model takes approximately 20 minutes to build.

4.2.1 Scalar Model Architectures

The custom model architecture evaluated in this study is called a scalar model because it includes two scalar inputs for latitude and longitude. The input layer of the model Fig. 7, denoted as input-feature, is designed to accommodate 3D image data with dimensions (Height, width, 18). This shape corresponds to a Height x width grid with 18 channels, representing different features or filters extracted from the input images.

The CNN architecture incorporates a series of convolutional layers, each followed by batch normalization (Simon et al. 2016), dropout (Baldi & Sadowski 2013), and activation with Rectified Linear Units (ReLU) (Xu et al. 2022). These convolutional layers are intended to capture hierarchical features from the input images. Simultaneously, scalar values are incorporated into the model through a separate input layer named input-scalar, with a shape of (2,), accommodating two scalar features (latitude and longitude).

Following the convolutional layers, a flattening layer is employed to transform the 3D tensor into a 1D tensor, enabling the integration of spatial features for subsequent processing. The flattened features from the convolutional layers and the scalar inputs are concatenated using the Concatenate layer. This step merges the spatial information extracted from the images with additional scalar information. The concatenated input is then passed through

additional dense layers, each followed by batch normalization, dropout, and activation with ReLU. These layers aim to further process and extract high-level representations from the combined information. The final layer of the model is a dense layer with a single neuron and a sigmoid activation function (Balaji & Baskaran 2013). This architecture is used for binary classification tasks, providing an output in the range [0, 1].

In summary, this CNN architecture is designed to capture both spatial features from the input images through convolutional layers and additional scalar information through dense layers. The model concludes with a sigmoid activation function, making it suitable for binary classification tasks. The utilization of batch normalization and dropout layers contributes to the model's generalization capabilities and robustness.

4.2.2 MobileNet

The model architecture presented consists of a series of convolutional layers followed by depthwise separable blocks. The initial convolutional layer has 32 filters with a kernel size of 3 and a stride of 2, followed by batch normalization and ReLU activation. The depthwise separable blocks comprise a depthwise convolutional layer followed by batch normalization and ReLU activation, followed by a pointwise convolutional layer with 1x1 filters, batch normalization, ReLU activation, and a dropout layer with a dropout rate of 0.3. These blocks are stacked to increase the model's depth and complexity. Additionally, max-pooling layers with a kernel size of 3 and strides of 2 are interspersed to downsample the feature maps. Global average pooling is applied after the last depthwise separable block, followed by a dense layer with 256 units and ReLU activation. Dropout regularization with a dropout rate of 0.5 is applied before the final output layer, which consists of a single neuron with sigmoid activation for binary classification. The model is trained using the Adam optimizer with a custom learning rate of 0.007. During training, early stopping based on validation loss is employed to prevent overfitting (Howard et al. 2017).

4.2.3 ResNet

The model begins with an input layer followed by a convolutional layer with 64 filters and a kernel size of 7x7. Batch normalization and ReLU activation are applied after the convolutional layer. The ResNet blocks are then added to the model, each consisting of two convolutional layers with batch normalization and ReLU activation, along with

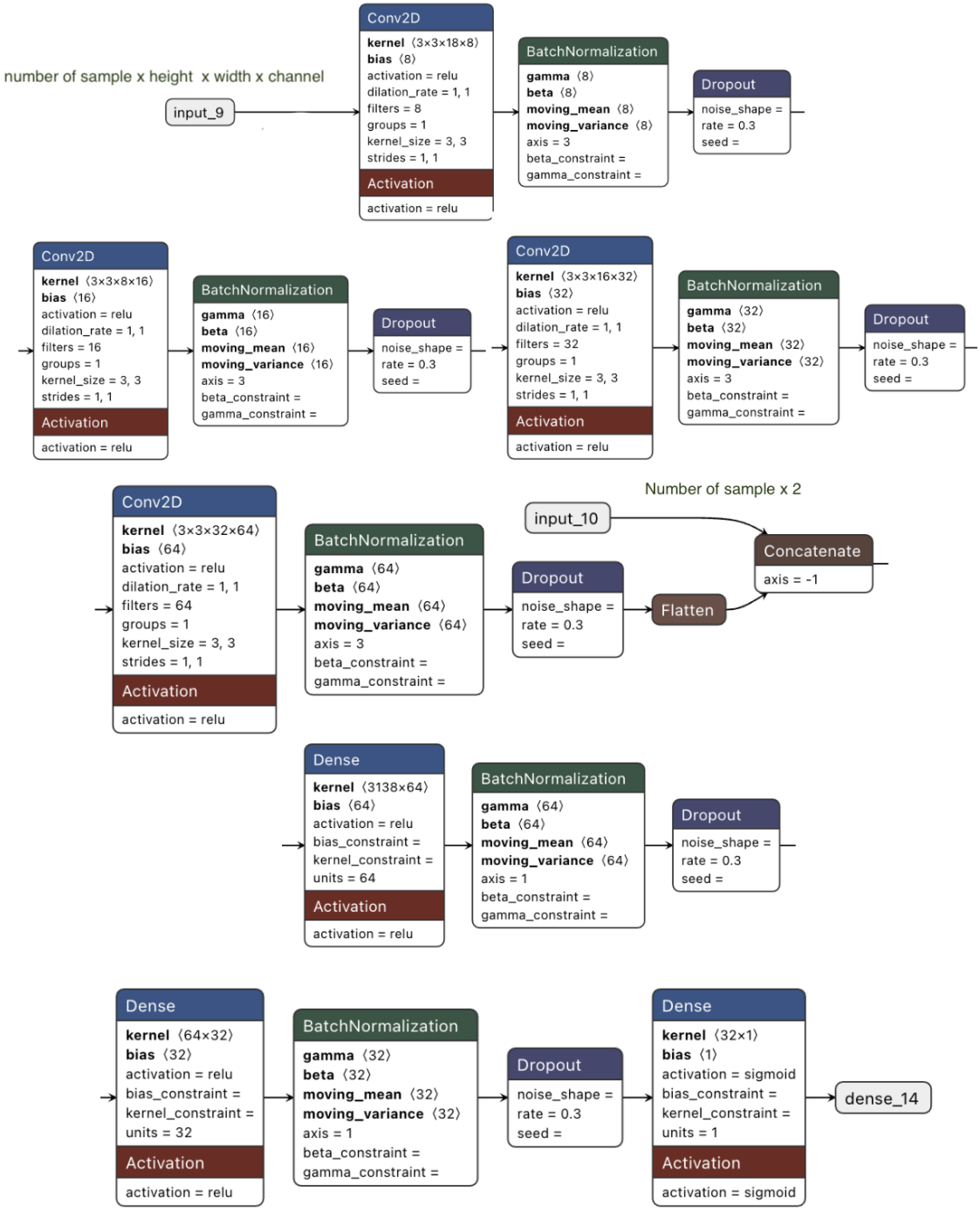

Figure 7. Architecture of the scalar model.

dropout regularization. The ResNet blocks help alleviate the vanishing gradient problem by introducing skip connections, enabling the model to learn more effectively. The model architecture includes four of these ResNet blocks with increasing filter sizes, and downsampling is performed using stride=2 in certain blocks to reduce spatial dimensions. After the ResNet blocks, the output is flattened and passed through a dense layer with 64 units and ReLU activation, followed by dropout regularization. Finally, a dense output layer with a sigmoid activation function is added for binary classification (He et al. 2016).

4.2.4 Xception

The model is based on the Xception architecture, which is inspired by the Inception architecture and designed to achieve better performance and efficiency. The Xception model consists of four Xception-like blocks. Each block contains two depthwise separable convolutional layers followed by batch normalization and ReLU activation. Additionally, dropout regularization is applied after each block to prevent overfitting. After the blocks, global average pooling is performed, followed by a dense layer with ReLU activation. Finally, dropout is applied again before the output layer, which consists of a single neuron with sigmoid activation for binary classification (Chollet 2017).

4.2.5 EfficientNetBO

The model is designed based on the EfficientNet architecture. The model begins with an input layer followed by four efficientnet-like blocks, each consisting of depthwise separable convolution layers, batch normalization, and ReLU activation functions. Dropout regularization is applied after each block to prevent overfitting. The number of filters in the convolutional layers gradually increases in the deeper blocks. After the efficientnet-like blocks, global average pooling is applied to reduce the spatial dimensions, followed by dense layers for classification. Dropout regularization is again applied to the dense layer to further prevent overfitting. The output layer consists of a single neuron with a sigmoid activation function for binary classification (Tan & Le 2019).

4.3 Baseline Models

This study includes two ML models for performance comparison.

4.3.1 Extreme Learning Machine

The ELM model consists of two main layers: the input layer and the output layer. In between, there are added neurons with different activation functions. Specifically, the model starts with an input layer that accepts the feature inputs. Then, it adds 20 neurons with a sigmoid activation function, followed by 10 neurons with an RBF activation and L2 norm (Alavi et al. 2009, Luo et al. 2016). The choice of activation functions helps the model learn complex patterns in the data. During training, the model is trained using the Leave-One-Out (LOO) cross-validation technique, which helps prevent overfitting and ensures robust performance (Akusok et al. 2021). Finally, the output layer produces predictions based on the learned patterns. Overall, this structure allows the ELM model to effectively classify data into two categories (Huang et al. 2004).

4.3.2 Random Forest

The RF classifier employed in the analysis is carefully tuned using a grid search technique to optimize its structure and hyperparameters. The model's architecture and key parameters are determined through this process. Specifically, the number of estimators, which represents the decision trees in the forest, is chosen from a range of options including 50, 100, 200, 300, and 500. The maximum depth of each decision tree is explored with values of None, 5, 10, 20, and 30, with "None" indicating unrestricted tree growth. Additionally, the minimum number of samples required to split an internal node and to form a leaf node were varied, with options of 1, 2, 3, 5, and 10 for both parameters. Subsequently, the RF classifier is trained using this best combination of hyperparameters, ensuring that it is well-suited for the task at hand and capable of providing robust predictions (Breiman 2001).

4.4 Performance Evaluation

The steps undertaken to evaluate the performance of the proposed CNN model. The evaluation involves the analysis of learning curves, the computation of classification metrics on the testing data, and the visualization of key performance indicators.

4.4.1 Learning Curve Analysis

To comprehend the training dynamics of the model, the learning curve is plotted using the training and validation accuracy over the epochs. This visualization provides insights

into the model's convergence and potential overfitting.

4.4.2 Model Evaluation on Testing Data

Following model training, the learned CNN model is evaluated on the reserved testing set. The model's predictions are obtained, and various classification metrics, including precision, recall, and F1 score, are computed. The confusion matrix is also presented to offer a detailed breakdown of predicted versus true labels.

4.4.3 Confusion Matrix Heatmap

The confusion matrix is visualized as a heatmap, offering an intuitive representation of true positive, true negative, false positive, and false negative predictions.

4.4.4 Receiver Operating Characteristic (ROC) Curve

To assess the model's discriminatory power, the ROC curve is computed, along with the Area Under the Curve (AUC) metric. This visual representation aids in understanding the trade-off between true positive and false positive rates.

The presented experimental setup aims to comprehensively evaluate the performance of the proposed CNN model. Learning curves, classification metrics, and visualizations provide a thorough understanding of the model's behavior and effectiveness in capturing relevant patterns from the input data.

5. Results

The results presented in this master's thesis are derived from two approaches: conventional methods utilizing PO linked to covariate values generating datasets from these covariate layers for each zoom level, and applying DL. The results are presented here in three subsections. Firstly, model evaluation for the dataset at a specific zoom level (10) is discussed. Secondly, the comparison of the models' performance across zoom levels 2 to 15 is presented. where evaluation metrics are test accuracy, precision, recall, F1 score, and ROC (AUC) Curve. Lastly, the obtained results are compared with the previous study in this field of research.

5.1 Scalar Input Model Evaluation

In this architecture, the model is evaluated using four different input sizes: 5x5, 9x9, 15x15, and 50x50 pixels. The obtained results are as follows:

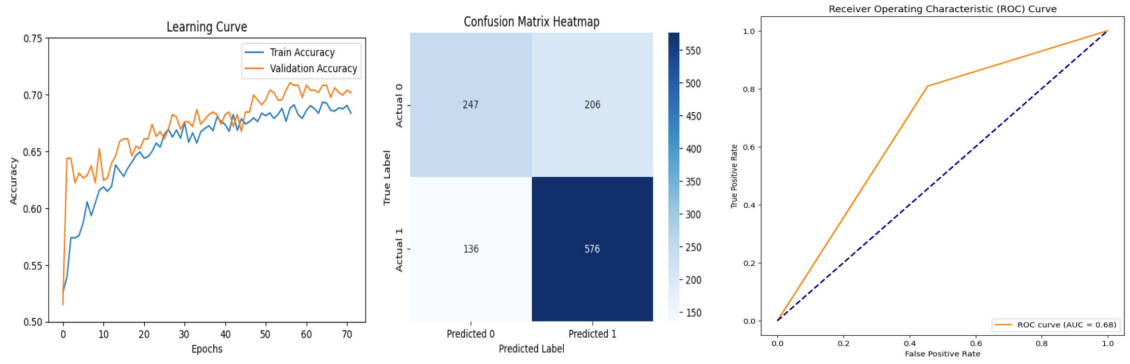

Figure 8. Visual Insights of the Model Performance: Learning Curve, Confusion Matrix, and ROC Curve for the Input size of 5x5px images.

The evaluation of the model with input size 5x5px. In (Fig. 8) shows promising results across various metrics. The test loss is calculated 0.58, indicating the average loss incurred during the evaluation process. In terms of accuracy, the model achieved a commendable score of 70.64 percent, showcasing its ability to make correct predictions. Precision, measuring the proportion of true positive predictions among all positive predictions, stood at 73.66 percent, while recall, representing the proportion of true positives identified correctly, reached 80.90 percent. The F1 score, a harmonic mean of precision and recall, is calculated 77.11 percent, reflecting a balanced performance in both metrics. A closer examination of the confusion matrix reveals the model's performance in

classifying instances. With 247 true negatives (non-ASS), 206 false negatives, 136 false positives, and 576 true positives (ASS), the matrix provides a comprehensive overview of the model's classification accuracy and potential areas for improvement. Additionally, an AUC of 0.68 suggests a moderate discriminatory power, indicating the model's ability to distinguish between classes.

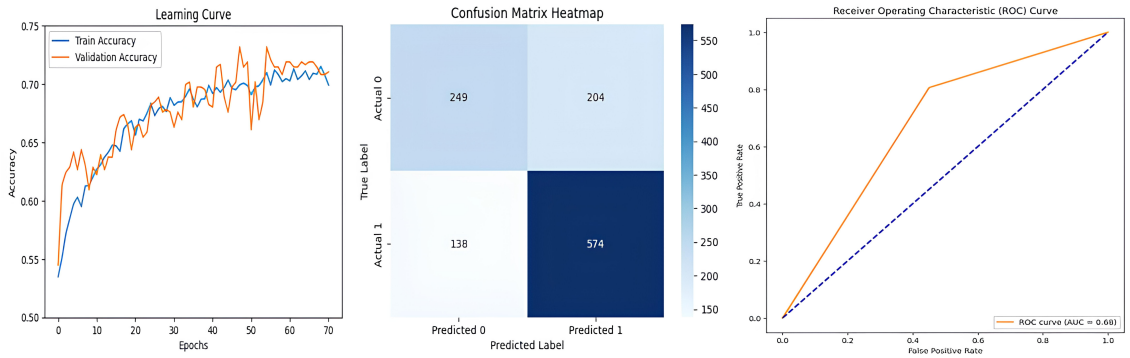

Figure 9. Visual Insights of the Model Performance: Learning Curve, Confusion Matrix, and ROC Curve for the Input size of 9x9px images.

The evaluation of the model with 9x9px is shown notable results in (Fig. 9). The test loss is calculated 0.58, with a corresponding test accuracy of 70.64 percent. Precision, recall, and F1 score are determined 73.78, 80.62, and 77.05 percent respectively. The confusion matrix further illustrates the model's performance, with 249 true negatives, 204 false negatives, 138 false positives, and 574 true positives. These metrics collectively provide a comprehensive overview of the model's classification accuracy. Overall, the results suggest promising performance, indicating the model's effectiveness in distinguishing between different classes of images.

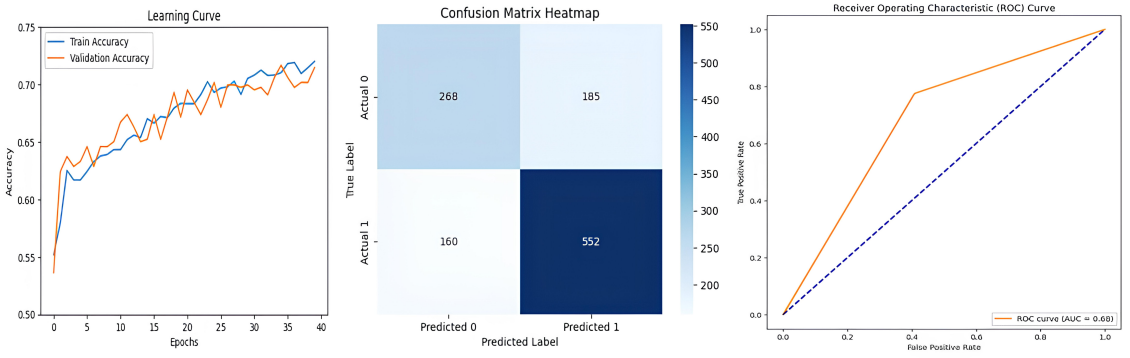

Figure 10. Visual Insights of the Model Performance: Learning Curve, Confusion Matrix, and ROC Curve for the Input size of 15x15px images.

The (Fig. 10) shows the performance of the model, based on the input image size 15x15

px, underscores its performance across various key metrics. Notably, the test loss is calculated at 0.58, with a corresponding test accuracy of 70.39 percent. Precision, recall, and F1 score demonstrate values of 74.90, 77.53, and 76.19 percent respectively, reflecting the model's ability to effectively classify instances. The confusion matrix provides further insight into the model's classification accuracy, revealing 268 true negatives, 185 false negatives, 160 false positives, and 552 true positives. Moreover, the ROC curve analysis yields an AUC value of 0.68, indicative of the model's moderate discriminatory power. These comprehensive metrics collectively paint a picture of the model's performance, showcasing its capability to make accurate predictions.

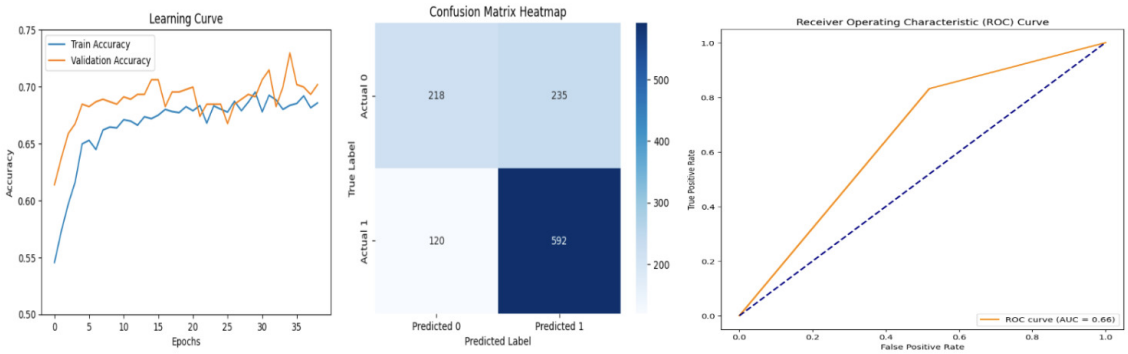

Figure 11. Visual Insights of the Model Performance: Learning Curve, Confusion Matrix, and ROC Curve for the Input size of 50x50px images.

The evaluation of the model is shown in (Fig. 11) with the input image size 50x50 px showcases its performance across various key metrics. The test loss is computed at 0.59, while the test accuracy stands at 69.53 percent. Precision, recall, and F1 score demonstrate values of 71.58, 83.15, and 76.93 percent respectively, indicating the model's effectiveness in correctly identifying instances of interest. The confusion matrix offers a detailed breakdown, revealing 218 true negatives, 235 false negatives, 120 false positives, and 592 true positives. Additionally, the ROC curve analysis yields an AUC value of 0.66, reflecting the model's moderate discriminatory power. These findings collectively provide insights into the model's classification accuracy, suggesting areas for potential improvement.

5.2 EfficientNetBO

The evaluation of the EfficientNetBO reveals its performance through metrics is shown in (Fig. 12). The test loss, at 0.60, indicates the average loss incurred during evalua-

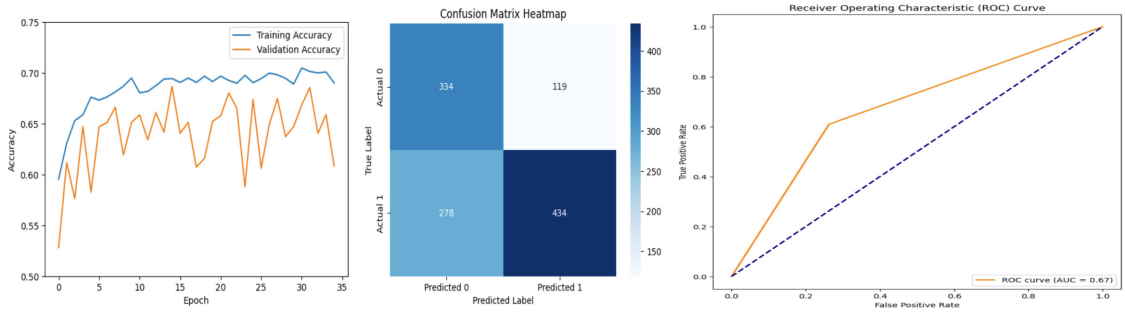

Figure 12. Visual Insights of the Model Performance: Learning Curve, Confusion Matrix, and ROC Curve for the EfficientNetBO model

tion, while the test accuracy stands at 65.92 percent, reflecting the proportion of correct predictions made. Precision is found 78.48 percent which measures the accuracy of positive predictions, while recall, indicates 60.96 percent of actual positives are correctly predicted. The F1 score, serving as a balance between precision and recall, is calculated at 68.62 percent, demonstrating overall performance. The confusion matrix provides a clear breakdown of the model's predictions, with 334 true negatives, 434 true positives, 119 false negatives, and 278 false positives. Additionally, the ROC curve, with an AUC of 0.67, shows the model's ability to distinguish between classes. These metrics together offer valuable insights into the model's classification accuracy and its capacity to make informed predictions.

5.3 MobileNet

The performance of the MobileNet model reveals its performance through various metrics as depicted in (Fig. 13). The test loss, recorded at 0.59, represents the average loss incurred during evaluation, while the test accuracy stands at 68.58 percent, indicating the proportion of correct predictions made. Precision, measuring the accuracy of positive predictions, is found 72.29 percent, while recall, indicating the proportion of actual positives correctly predicted, is at 78.79 percent.

The F1 score, serving as a balance between precision and recall, is calculated at 75.40 percent, demonstrating the overall performance of the model. The confusion matrix provides the model's predictions, with 238 true negatives, 215 false negatives, 151 false positives, and 561 true positives. Additionally, the ROC curve analysis, with an AUC value of 0.66, illustrates the model's ability to distinguish between classes. Together, these metrics of-

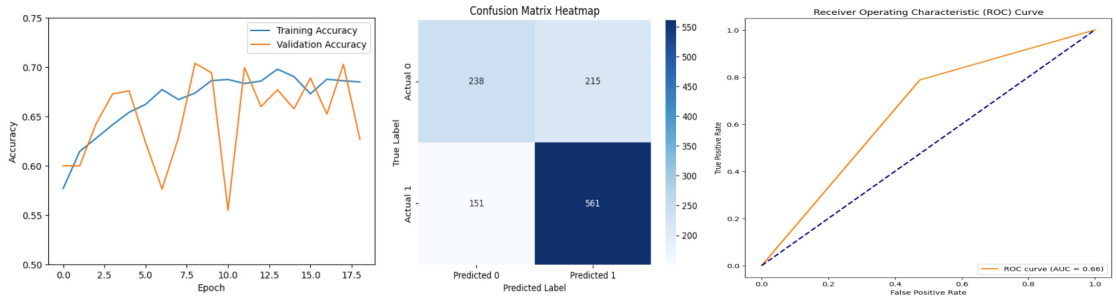

Figure 13. Visual Insights of the Model Performance: Learning Curve, Confusion Matrix, and ROC Curve for the MobileNet model.

fer valuable insights into the model's classification accuracy and its capacity to make informed predictions.

5.4 ResNet

The output of the model provides a comprehensive overview of its performance across various key metrics as shown in (Fig. 14). The test loss, computed at 0.58, serves as an indicator of the average loss incurred during evaluation. With a test accuracy of 68.50 percent, the model demonstrates its ability to make correct predictions. Precision, which measures the accuracy of positive predictions, is determined 72.03 percent, while recall, representing the proportion of actual positives correctly predicted, reaches 79.21 percent. The F1 score, a balance between precision and recall, is calculated at 75.45 percent, indicating the model's overall performance. The confusion matrix further elucidates the

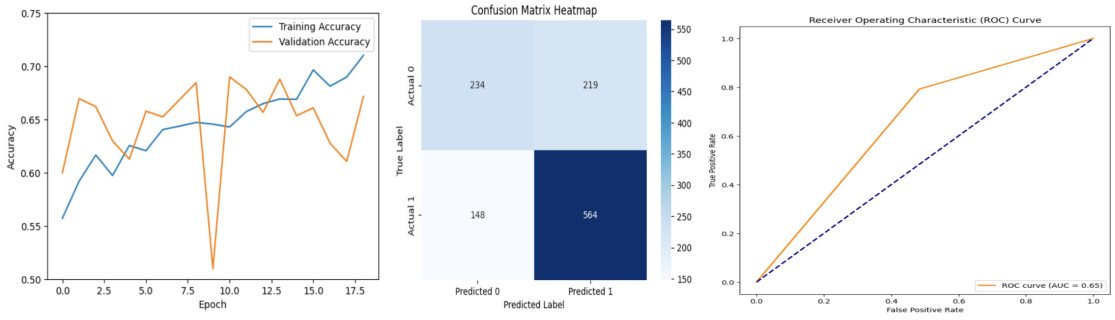

Figure 14. Visual Insights of the Model Performance: Learning Curve, Confusion Matrix, and ROC Curve for the ResNet model

model's predictive capabilities, showing 234 true negatives, 219 false negatives, 148 false positives, and 564 true positives. Additionally, the ROC curve analysis with an AUC value of 0.65 highlights the model's ability to distinguish between classes. These find-

ings collectively provide valuable insights into the model's classification accuracy and its capacity to make informed predictions, offering a comprehensive understanding of its performance.

5.5 Xception

The xception model reveals its performance across various key metrics. The test loss, computed at 0.58, signifies the average loss incurred during evaluation, while the test accuracy stands at 68.07 percent, reflecting the proportion of correct predictions made. Precision, measuring the accuracy of positive predictions, is determined 72.55 percent, indicating the model's proficiency in correctly identifying positive instances.

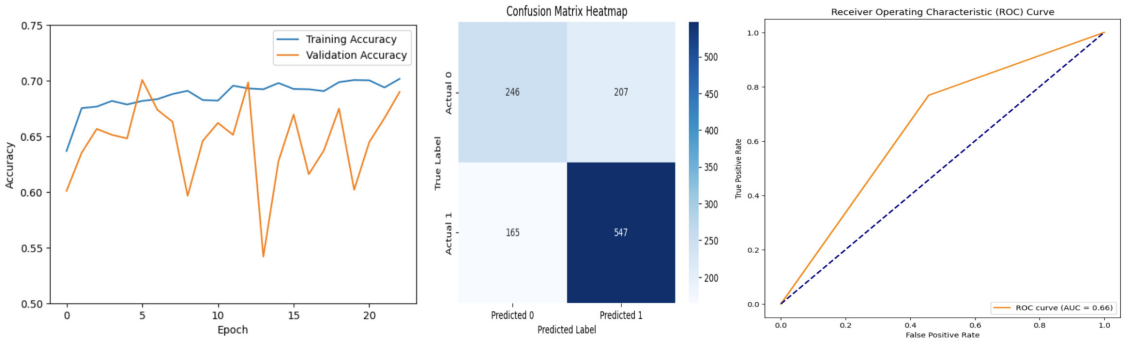

Figure 15. Visual Insights of the Model Performance: Learning Curve, Confusion Matrix, and ROC Curve for the Xception model

Meanwhile, recall, representing the proportion of actual positives correctly predicted, reaches 76.83 percent, demonstrating the model's ability to capture the most positive instances. The F1 score, which combines precision and recall into a single metric, is calculated at 74.62 percent, showcasing a balanced performance in both aspects.

The confusion matrix is shown in (Fig. 15) provides a detailed breakdown of the model's predictions, showing 246 true negatives, 207 false negatives, 165 false positives, and 547 true positives. Additionally, the ROC curve analysis reveals an AUC of 0.66, indicating the model's ability to distinguish between classes. These metrics collectively offer valuable insights into the model's classification accuracy and its capacity to make informed predictions, providing a comprehensive assessment of its performance.

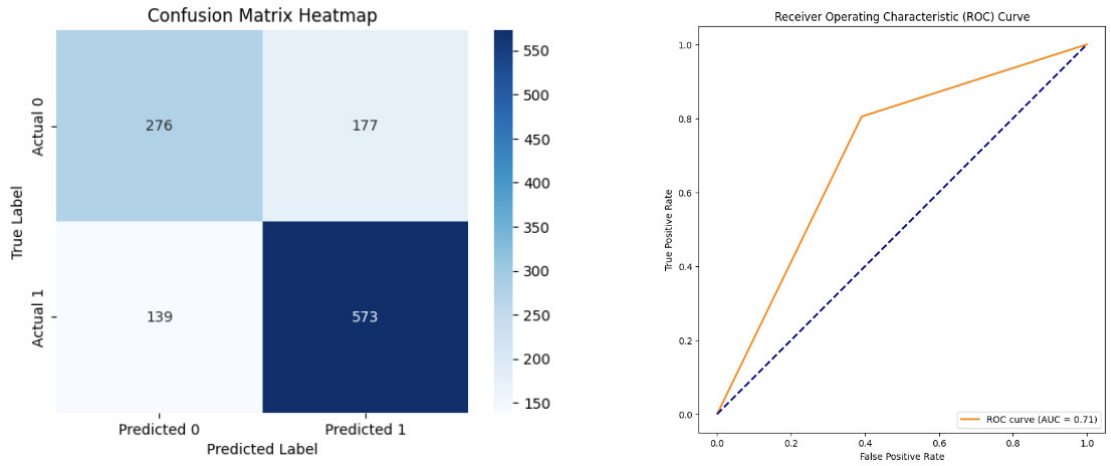

Figure 16. Visual Insights of the Model Performance: Learning Curve, Confusion Matrix, and ROC Curve for the RF

5.6 Random Forest

RF reveals promising performance. With an accuracy of 73 percent, it demonstrates the proportion of correct predictions made by the model. Precision, measuring the accuracy of positive predictions, is determined 76 percent, while recall, indicating the proportion of actual positives correctly predicted, reaches 80 percent. The F1 score, a harmonic mean of precision and recall, is calculated at 78 percent, reflecting a balanced performance in both aspects. The confusion matrix shows in (Fig. 16) a detailed breakdown of the model's predictions, showing 276 true negatives, 177 false negatives, 139 false positives, and 573 true positives. Additionally, the ROC curve analysis with an AUC of 0.71 indicates the model's ability to distinguish between classes. These metrics collectively provide valuable insights into the model's classification accuracy and its capacity to make informed predictions, offering a comprehensive assessment of its performance.

5.7 Extreme Learning Machine

ELM model illustrates its performance, with an accuracy of 66 percent, it reflects the proportion of correct predictions made. Precision, which measures the accuracy of positive predictions, stands at 69 percent, while recall, indicating the proportion of actual positives correctly predicted, is notably higher at 82 percent. The F1 score, balancing precision and recall, is computed at 75 percent, signifying a harmonious performance across both metrics. Examining the confusion matrix reveals a detailed breakdown of the model's predictions in (Fig. 17), with 189 true negatives, 264 false negatives, 131 false

positives, and 581 true positives.

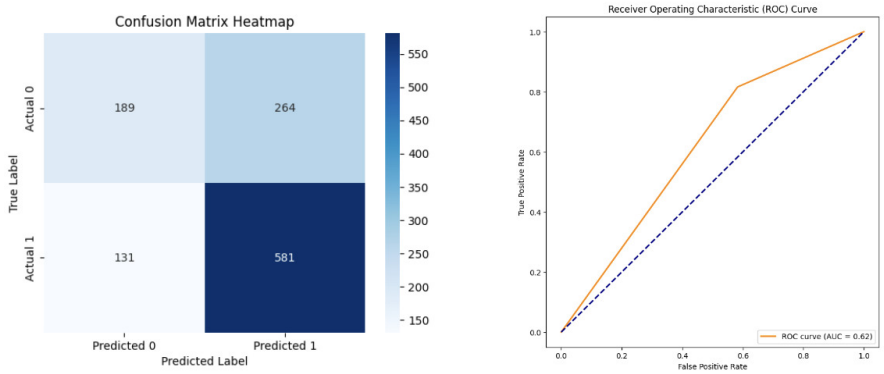

Figure 17. Visual Insights of the Model Performance: Learning Curve, Confusion Matrix, and ROC Curve for the Extreme Learning Machine

Additionally, the ROC curve analysis, with an AUC value of 0.66, underscores the model's ability to differentiate between classes. Collectively, these metrics provide valuable insights into the model's classification accuracy and its capability to make informed predictions, offering a comprehensive assessment of its performance.

5.8 Models Performance Across Map Zoom Levels

RF consistently achieves high accuracy across zoom levels, especially at higher levels as shown in Fig. 18 while Scalar-Model-5x5 and Scalar-Model-9x9 also perform competi-

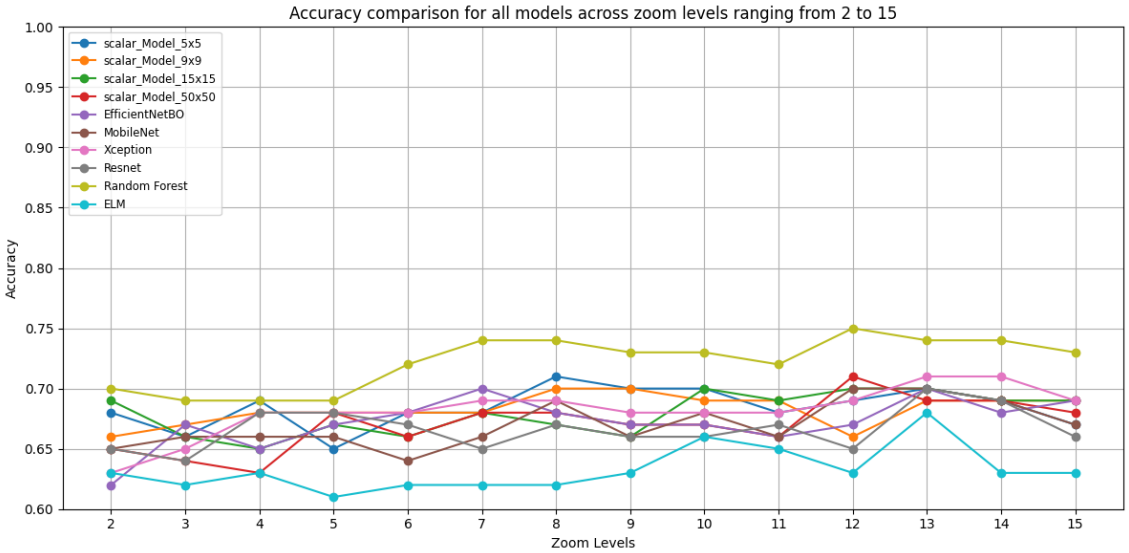

Figure 18. Accuracy comparison for all models across zoom levels ranging from 2 to 15

tively. EfficientNetBO and MobileNet show stable but slightly lower accuracy compared

to RF. Xception and Resnet maintain stable performance, while ELM demonstrates comparatively lower accuracy.

The precision comparison across different models reveals nuanced performance differences at different zoom levels. Among the scalar models, scalar-Model-15x15 consistently exhibits higher precision, particularly at higher zoom levels. Conversely, Mo-

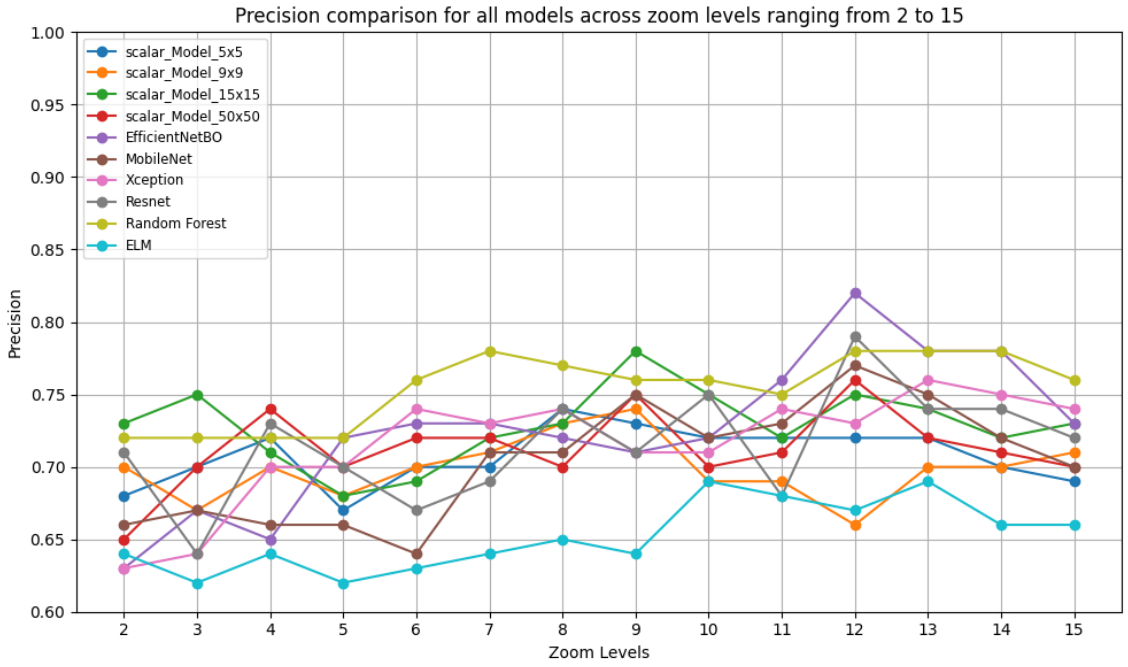

Figure 19. Precision comparison for all models across zoom levels ranging from 2 to 15

bileNet and Xception exhibit stable but comparatively lower precision across all zoom levels. Interestingly, RF presents a remarkable precision, especially at mid to higher zoom levels as shown in Fig 19. However, ELM yields lower precision across the board, indicating potential limitations. These insights underscore the significance of model selection tailored to specific zoom level requirements in precision-sensitive applications.

The line graph illustrates in Fig 20 the recall performance of different models across zoom levels 2 to 15. Each line represents a specific model, showcasing how their recall values vary with changing zoom levels. Models like RF and ELM generally maintain higher recall scores across all zoom levels, indicating their effectiveness in correctly identifying relevant information. On the other hand, models like EfficientNetBO and MobileNet exhibit fluctuating recall values, suggesting varying degrees of sensitivity to different zoom

levels. Xception and Resnet demonstrate relatively stable performance throughout the zoom levels, indicating consistent recall rates across different scales. Overall, the graph offers insights into how different models perform in terms of recall under varying levels of zoom, highlighting their strengths and weaknesses in handling spatial information.

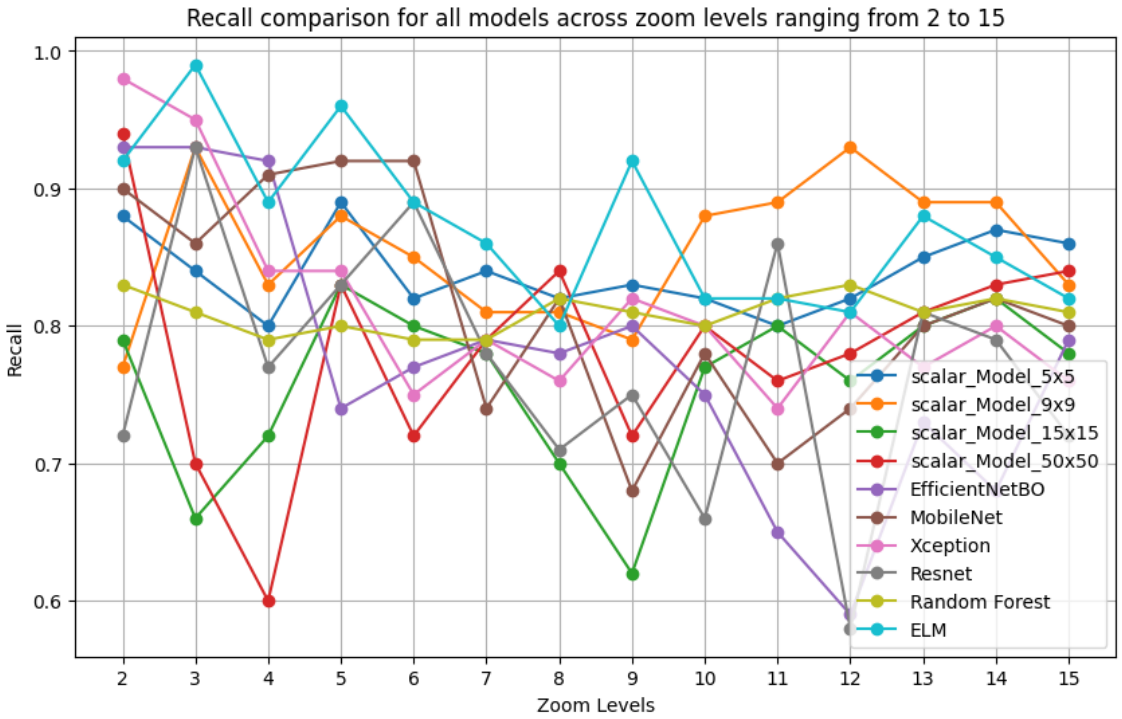

Figure 20. Recall comparison for all models across zoom levels ranging from 2 to 15

The F1 scores for different models exhibit varying trends as shown in Fig 21. For instance, RF consistently maintains high F1 scores, indicating robust performance. On the other hand, ELM demonstrates comparatively lower F1 scores, especially at higher zoom levels. Models like scalar-Model-15x15 and EfficientNetBO, display fluctuations in performance, with peaks and troughs at different zoom levels. Overall, the F1 score comparison highlights the effectiveness of certain models in handling different zoom levels and underscores the importance of selecting appropriate models based on specific application requirements.

In Fig. 22, RF consistently demonstrates strong performance across all zoom levels, with ROC AUC scores ranging from 0.70 to 0.72. Models like EfficientNetBO and Resnet show relatively stable performance across zoom levels, while others, such as ELM, exhibit more variability. MobileNet performs relatively well at lower zoom levels but expe-

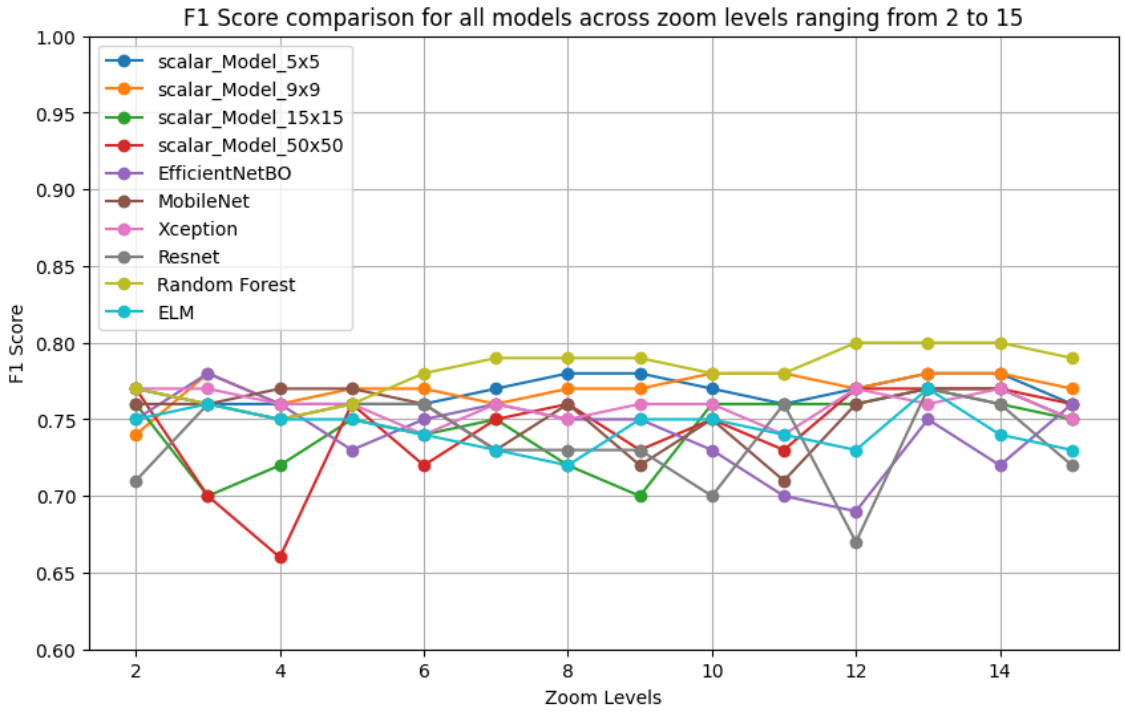

Figure 21. F1 Score comparison for all models across zoom levels ranging from 2 to 15

riences a decline in performance as zoom levels increase. On the other hand, Xception shows a steady improvement in performance with increasing zoom levels. Scalar input models mostly show stable performance across the zoom levels. The insights gained from the heatmap analysis can inform the decision-making process when choosing models for large-scale mapping tasks.

5.9 Comparison of the Results

The performance of ten different models across various metrics including accuracy, precision, recall, F1 score, the Area Under the ROC Curve (AUC), and the confusion matrix. Below is the table. 2 shows the comparison of their performance for the zoom level 10 as a reference:

Models such as scalar-Model-5x5, scalar-Model-9x9, and scalar-Model-15x15 exhibit high accuracy and precision, indicating their ability to make correct predictions and minimize false positives. while scalar-model-50x50 and RF demonstrate high recall, implying their capability to identify a high proportion of true positives among all actual positives. RF achieves the highest F1 score, signifying a balanced performance between precision and recall. RF also demonstrates the highest AUC, implying its superior ability to distin-

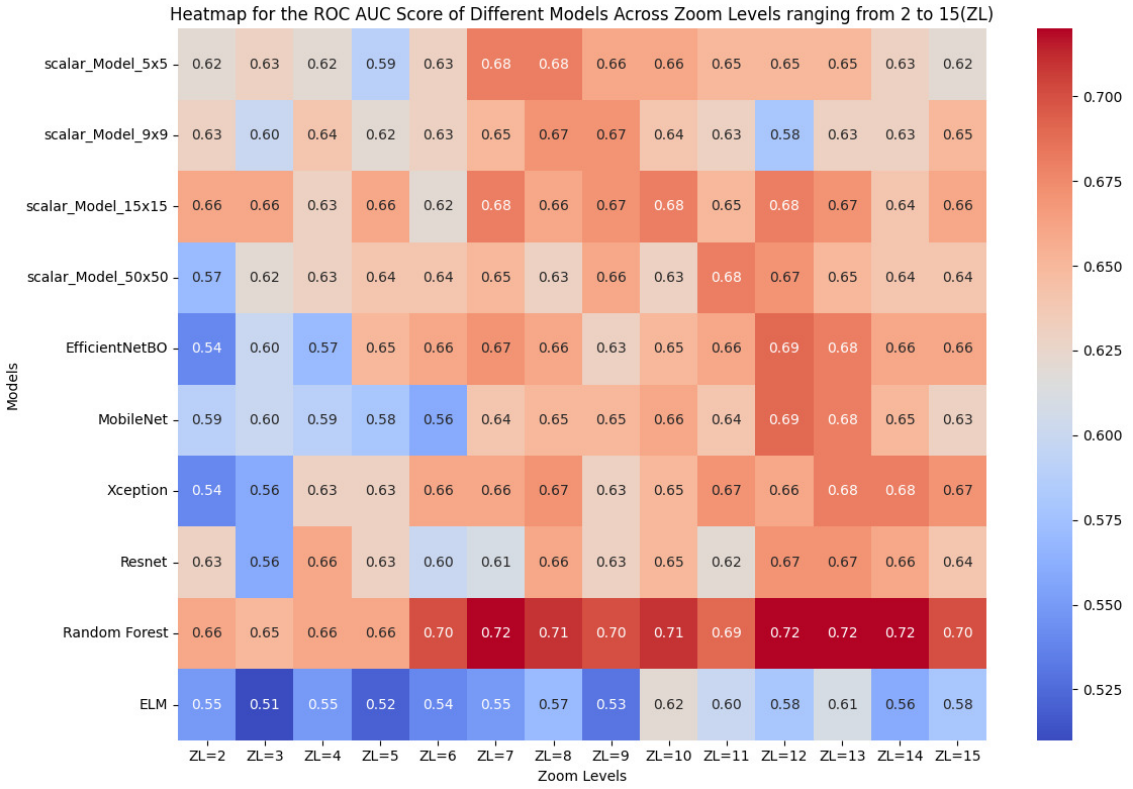

Figure 22. Heatmap for the ROC AUC score of different models across zoom levels ranging from 2 to 15.

guish between classes.

The results suggest that scalar models of different input sizes perform consistently well across various metrics, indicating their robustness. However, models like RF and EfficientNetBO also exhibit strong performance, particularly in terms of recall and AUC. While RF stands out as the top-performing model overall with the highest accuracy, precision, recall, F1 score, and AUC, it's essential to consider factors such as computational efficiency and model complexity when selecting the final deployment model.

In contrast, Amélie Beucher et al.'s study shows an overall accuracy of 68 percent, utilizing 14 covariate layers as input data (Beucher et al. 2022). Additionally, Virginia's research, focusing on acid sulfide soil classification, employs 5 covariate layers and achieves an accuracy range of 55-60 percent (Estévez Nuño 2020). Moreover, Atsemegiorgis et al. (2023) utilized 13 environmental covariate layers in their study and achieved an overall model accuracy of 71 percent for both RF and ELM models (Atsemegiorgis et al. 2023).

Table 2. Model Evaluation Metrics for zoom level 10

Models	**Accuracy**	**Precision**	**Recall**	**F1 Score**	**ROC curve (AUC)**
scalar_Model_5x5	0.71	0.74	0.81	0.77	0.68
scalar_Model_9x9	0.71	0.74	0.81	0.77	0.68
scalar_Model_15x15	0.70	0.75	0.78	0.76	0.68
scalar_Model_50x50	0.70	0.72	0.83	0.77	0.66
EfficientNetBO	0.66	0.78	0.61	0.69	0.67
MobileNet	0.69	0.72	0.79	0.75	0.66
Xception	0.68	0.73	0.77	0.75	0.66
Resnet	0.68	0.72	0.79	0.75	0.65
Random Forest	0.73	0.76	0.80	0.78	0.71
ELM	0.66	0.69	0.82	0.75	0.62

While this master's thesis employs 10 covariate layers, achieving a maximum accuracy of 75 percent at zoom level 12 for RF, CNN scalar models demonstrate stable performance with around 70 percent accuracy across all zoom levels. This comparison highlights the diverse performance outcomes influenced by input data complexity and model design choices.

6. Conclusions

This master's thesis presented a comprehensive study of AS soil classification across a significant portion of the Finnish landscape. This work comprises two main parts. The first involves creating datasets for the study, primarily accomplished using Python. The second part entails analyzing various DL architectures for AS soil classification.

Although considerable effort has been made in studying AS soils in the country over the last decade, there is a scarcity of AS soil maps. The objective of this thesis has been to identify lightweight DL methods capable of accurately classifying AS and Non-AS soils. This would enable more precise mapping of AS soils. All the methods examined for this binary classification are supervised DL techniques, many of which have not been previously applied to AS soil classification.

The findings suggest that lightweight scalar-input models, utilizing raster datasets sourced from different origins and incorporating a variety of input sizes, consistently demonstrate strong performance across multiple metrics. This underscores their resilience and potential suitability for mobile applications, drawing inspiration from the concept of Akusok et al. (2023). Notably, the RF model shows strong performance, especially in accuracy, recall, and AUC. Although the RF emerges as the top-performing model overall, excelling in accuracy, precision, recall, F1 score, and AUC, it's crucial to consider factors like computational efficiency and model complexity when choosing the final deployment model.

In future research, there are several avenues to further enhance the classification of AS soils. One promising direction is to build upon the success of the scalar-input models observed in the current study. Notable performance is achieved using a dataset comprising ten covariate layers. Expanding this dataset to include a greater number of covariate layers could provide a more nuanced understanding of AS soils and lead to improved classification accuracy. Furthermore, in future work, exploring unsupervised pre-training of DL models for tasks with limited labeled data holds potential. This approach could help the available data more effectively and enhance the performance of classification models.

References

Agarwal, Rahul & Shekhawat, Narpat Singh. 2022, Enhanced bag of features using AlexNet and henry gas solubility optimization for soil image classification, In: *Proceedings of International Conference on Data Science and Applications: ICDSA 2021, Volume 2*, Springer, pp. 493–503.

Akusok, Anton; Espinosa Leal, Leonardo; Björk, Kaj-Mikael & Hu, Renjie. 2023, Native interaction experience for computational maps with mobile devices, In: *Proceedings of the 16th International Conference on PErvasive Technologies Related to Assistive Environments*, pp. 162–164.

Akusok, Anton; Leal, Leonardo Espinosa; Björk, Kaj-Mikael & Lendasse, Amaury. 2021, Scikit-elm: an extreme learning machine toolbox for dynamic and scalable learning, In: *Proceedings of ELM2019 9*, Springer, pp. 69–78.

Alavi, AH; Gandomi, AH; Gandomi, M & Sadat Hosseini, SS. 2009, Prediction of maximum dry density and optimum moisture content of stabilised soil using RBF neural networks, *The IES Journal Part A: Civil & Structural Engineering*, vol. 2, no. 2, pp. 98–106.

Alexandre, M. C. W.; Padarian, J. & Minasny, B. 2019, Multi-source data integration for soil mapping using deep learning.

Atsemegiorgis, Tamirat; Espinosa-Leal, Leonardo; Lendasse, Amaury; Mattbäck, Stefan; Björk, Kaj-Mikael & Akusok, Anton. 2023, Acid Sulfate Soils Classification and Prediction from Environmental Covariates Using Extreme Learning Machines, In: *International Work-Conference on Artificial Neural Networks*, Springer, pp. 614–625.

Austnes, Kari; Aherne, Julian; Arle, Jens; Čičendajeva, Marina; Couture, Suzanne; Fölster, Jens; Garmo, Øyvind Aaberg; Hruska, Jakub; Monteith, Don; Posch, Max et al.. 2018, Regional assessment of the current extent of acidification of surface waters in Europe and North America, *NIVA-rapport*.

Bahrampour, Soheil; Ramakrishnan, Naveen; Schott, Lukas & Shah, Mohak. 2016, Comparative study of caffe, neon, theano, and torch for deep learning.

Balaji, S Arun & Baskaran, K. 2013, Design and development of artificial neural networking (ANN) system using sigmoid activation function to predict annual rice production in Tamilnadu, *arXiv preprint arXiv:1303.1913*.

Baldi, P. & Sadowski, P. J. 2013, Understanding dropout, In: *Advances in Neural Information Processing Systems*, vol. 26.

Behrens, T.; Schmidt, K.; MacMillan, R. A. & Viscarra Rossel, R. A. 2018, Multi-scale digital soil mapping with deep learning, *Scientific Reports*, vol. 8, no. 1, p. 15244.

Belgiu, M. & Drăguţ, L. 2016, Random Forest in Remote Sensing: A Review of Applications and Future Directions, *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 114, , pp. 24–31.

Beucher, A.; Adhikari, K.; Breuning-Madsen, H.; Greve, M. B.; Österholm, P.; Fröjdö, S. & Greve, M. H. 2017, Mapping potential acid sulfate soils in Denmark using legacy data and LiDAR-based derivatives, *Geoderma*, vol. 308, , pp. 363–372.

Beucher, A.; Fröjdö, S.; Österholm, P.; Martinkauppi, A. & Edén, P. 2014, Fuzzy Logic for Acid Sulfate Soil Mapping: Application to the Southern Part of the Finnish Coastal Areas, *Geoderma*, vol. 226, , pp. 21–30.

Beucher, A.; Siemssen, R.; Fröjdö, S.; Österholm, P.; Martinkauppi, A. & Edén, P. 2015, Artificial neural network for mapping and characterization of acid sulfate soils: Application to Sirppujoki River catchment, southwestern Finland, *Geoderma*, vol. 247, , pp. 38–50.

Beucher, A.; Österholm, P.; Martinkauppi, A.; Edén, P. & Fröjdö, S. 2013, Artificial neural network for acid sulfate soil mapping: Application to the Sirppujoki River catchment area, south-western Finland, *Journal of Geochemical Exploration*, vol. 125, , pp. 46–55.

Beucher, Amélie; Rasmussen, Christoffer B; Moeslund, Thomas B & Greve, Mogens H. 2022, Interpretation of convolutional neural networks for acid sulfate soil classification, *Frontiers in Environmental Science*, vol. 9, , p. 809995.

Bierwirth, P. N. & Brodie, R. S. 2005, *Identifying acid sulfate soil hotspots from airborne gamma-radiometric data and GIS analysis*, Canberra, 59-60 p..

Boman, A.; Mattbäck, S.; Becher, M.; Sohlenius, G.; Auri, J.; Öhrling, C.; ... & Edén, P. 2020, Classification of acid sulfate soils and materials in Finland and Sweden: Re-introduction of pseudoacid sulfate soil materials, In: *9th International Acid Sulfate Soils Conference Abstract Book*, p. 45.

Bousmalis, K.; Silberman, N.; Dohan, D.; Erhan, D. & Krishnan, D. 2017, Unsupervised Pixel-Level Domain Adaptation with Generative Adversarial Networks, In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3722–3731.

Breiman, Leo. 2001, Random forests, *Machine Learning*, vol. 45, , pp. 5–32.

Carpenter, G. A.; Grossberg, S.; Markuzon, N.; Reynolds, J. H. & Rosen, D. B. 1992, Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps, *IEEE Transactions on Neural Networks*, vol. 3, no. 5, pp. 698–713.

Chaib, S.; Yao, H.; Gu, Y. & Amrani, M. 2017, Deep feature extraction and combination for remote sensing image classification based on pre-trained CNN models, In: *Ninth International Conference on Digital Image Processing (ICDIP 2017)*, vol. 10420, SPIE, pp. 712–716.

Chen, H.; Zhang, Y.; Kalra, M. K.; Lin, F.; Chen, Y.; Liao, P.; ... & Wang, G. 2017, Low-dose CT with a Residual Encoder-Decoder Convolutional Neural Network, *IEEE Transactions on Medical Imaging*, vol. 36, no. 12, pp. 2524–2535.

Chen, J.; Wang, C.; Ma, Z.; Chen, J.; He, D. & Ackland, S. 2018, Remote sensing scene classification based on convolutional neural networks pre-trained using attention-guided sparse filters, *Remote Sensing*, vol. 10, no. 2, p. 290.

Chollet, François. 2017, Xception: Deep learning with depthwise separable convolutions, In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1251–1258.

Decker, B LOUIS. 1986, World geodetic system 1984, *Defense Mapping Agency Aerospace Center St Louis Afs Mo*.

Du, P.; Li, E.; Xia, J.; Samat, A. & Bai, X. 2018, Feature and model level fusion of pre-trained CNN for remote sensing scene classification, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 8, pp. 2600–2611.

Ertam, F. & Aydın, G. 2017, Data classification with deep learning using Tensorflow, In: *2017 International Conference on Computer Science and Engineering (UBMK)*, IEEE, pp. 755–758.

Estévez Nuño, Virginia. 2020, Machine Learning methods for classification of Acid Sulfate soils in Virolahti.

Fitzpatrick, R. W.; Powell, B. & Marvanek, S. 2008, Atlas of Australian acid sulfate soils. Inland acid sulfate soil systems across Australia, In: *Proceedings of the International Symposium on Inland Acid Sulfate Soils*, vol. 249, pp. 75–89.

Fältmarsch, R. M.; Åström, M. E. & Vuori, K. M. 2008, Environmental risks of metals mobilised from acid sulphate soils in Finland: a literature review, *Undefined*.

Ge, Y.; Jiang, S.; Xu, Q.; Jiang, C. & Ye, F. 2018, Exploiting representations from pre-trained convolutional neural networks for high-resolution remote sensing image retrieval, *Multimedia Tools and Applications*, vol. 77, , pp. 17489–17515.

Hammer, B.; Micheli, A.; Sperduti, A. & Strickert, M. 2004, A general framework for unsupervised processing of structured data, *Neurocomputing*, vol. 57, , pp. 3–35.

He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing & Sun, Jian. 2016, Deep residual learning for image recognition, In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Hinton, G. & Sejnowski, T. J., eds.. 1999, *Unsupervised Learning: Foundations of Neural Computation*, MIT Press.

Howard, Andrew G.; Zhu, Menglong; Chen, Bo; Kalenichenko, Dmitry; Wang, Weijun; Weyand, Tobias; ... & Adam, Hartwig. 2017, Mobilenets: Efficient convolutional neural networks for mobile vision applications, *arXiv preprint arXiv:1704.04861*.

Huang, F.; Lu, J.; Tao, J.; Li, L.; Tan, X. & Liu, P. 2019, Research on Optimization Methods of ELM Classification Algorithm for Hyperspectral Remote Sensing Images, *IEEE Access*, vol. 7, , pp. 108070–108089.

Huang, Guang-Bin; Zhu, Qin-Yu & Siew, Chee-Kheong. 2004, Extreme learning machine: a new learning scheme of feedforward neural networks, In: *2004 IEEE international joint conference on neural networks (IEEE Cat. No. 04CH37541)*, vol. 2, IEEE, pp. 985–990.

Huang, J.; Nhan, T.; Wong, V. N.; Johnston, S. G.; Lark, R. M. & Triantafilis, J. 2014, Digital soil mapping of a coastal acid sulfate soil landscape, *Soil Research*, vol. 52, no. 4, pp. 327–339.

Jordan, M. I. & Rumelhart, D. E. 2013, Forward models: Supervised learning with a distal teacher, In: *Backpropagation*, Psychology Press, pp. 189–236.

Lee, J. G.; Jun, S.; Cho, Y. W.; Lee, H.; Kim, G. B.; Seo, J. B. & Kim, N. 2017, Deep Learning in Medical Imaging: General Overview, *Korean Journal of Radiology*, vol. 18, no. 4, pp. 570–584.

Lorent, S. 2019, *Fake news detection using machine learning*, Master's thesis.

Luo, Xiong; Chang, Xiaohui & Ban, Xiaojuan. 2016, Regression and classification using extreme learning machine based on L1-norm and L2-norm, *Neurocomputing*, vol. 174, , pp. 179–186.

Majib, Mohammad Shahjahan; Rahman, Md Mahbubur; Sazzad, TM Shahriar; Khan, Nafiz Imtiaz & Dey, Samrat Kumar. 2021, Vgg-scnet: A vgg net-based deep learning framework for brain tumor detection on mri images, *IEEE Access*, vol. 9, , pp. 116942–116952.

MapTiler. 2022, *Google Maps Coordinates, Tile Bounds, and Projection*. Accessed: April 4, 2024. Available: https://www.maptiler.com/google-maps-coordinates-tile-bounds-projection/#3/15.00/50.00.

Marmanis, D.; Datcu, M.; Esch, T. & Stilla, U. 2015, Deep learning earth observation classification using ImageNet pretrained networks, *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 1, pp. 105–109.

McCracken, R. J. & Cate, R. B. 1986, Artificial intelligence, cognitive science, and measurement theory applied in soil classification, *Soil Science Society of America Journal*, vol. 50, no. 3, pp. 557–561.

Microsoft. 2022, *Bing Maps Tile System*. Accessed: April 4, 2024. Available: https://learn.microsoft.com/en-us/bingmaps/articles/bing-maps-tile-system?redirectedfrom=MSDN.

Padarian, J.; Minasny, B. & McBratney, A. B. 2017, Chile and the Chilean soil grid: a contribution to GlobalSoilMap, *Geoderma Regional*, vol. 9, , pp. 17–28.

Padarian, J.; Minasny, B. & McBratney, A. B. 2019, Using deep learning for digital soil mapping, *Soil*, vol. 5, no. 1, pp. 79–89.

Palko, J. K. 1996, Acid sulphate soils and their agricultural and environmental problems in Finland.

Pena, J. M.; Lozano, J. A.; Larranaga, P. & Inza, I. 2001, Dimensionality Reduction in Unsupervised Learning of Conditional Gaussian Networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 590–603.

Shao, Zhenfeng; Yang, Ke & Zhou, Weixun. 2018, A Benchmark Dataset for Performance Evaluation of Multi-Label Remote Sensing Image Retrieval., *Remote Sensing*, vol. 10, no. 6.

Simon, M.; Rodner, E. & Denzler, J. 2016, Imagenet pre-trained models with batch normalization, *arXiv preprint arXiv:1612.01452*.

Tan, Mingxing & Le, Quoc V. 2019, Efficientnet: Rethinking model scaling for convolutional neural networks, In: *International conference on machine learning*, PMLR, pp. 6105–6114.

Tang, Pengjie; Wang, Hanli & Kwong, Sam. 2017, G-MS2F: GoogLeNet based multi-stage feature fusion of deep CNN for scene recognition, *Neurocomputing*, vol. 225, , pp. 188–197.

Theckedath, Dhananjay & Sedamkar, RR. 2020, Detecting affect states using VGG16, ResNet50 and SE-ResNet50 networks, *SN Computer Science*, vol. 1, no. 2, p. 79.

Touretzky, David S & Pomerleau, Dean A. 1989, What's hidden in the hidden layers, *Byte*, vol. 14, no. 8, pp. 227–233.

Xia, Gui-Song; Yang, Wen; Delon, Julie; Gousseau, Yann; Sun, Hong & Maître, Henri. 2010, Structural high-resolution satellite image indexing, In: *ISPRS TC VII Symposium-100 Years ISPRS*, vol. 38, pp. 298–303.

Xu, Jin-Jian; Zhang, Hao; Tang, Chao-Sheng; Cheng, Qing; Tian, Ben-gang; Liu, Bo & Shi, Bin. 2022, Automatic soil crack recognition under uneven illumination condition with the application of artificial intelligence, *Engineering Geology*, vol. 296, , p. 106495.

Yang, Yi & Newsam, Shawn. 2010, Bag-of-visual-words and spatial extensions for land-use classification, In: *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*, pp. 270–279.

Yuan, Zheng-Wu & Zhang, Jun. 2016, Feature extraction and image retrieval based on AlexNet, In: *Eighth International Conference on Digital Image Processing (ICDIP 2016)*, vol. 10033, SPIE, pp. 65–69.

Zhang, Xinxun; Jiao, Pengfei; Gao, Mengzhou; Li, Tianpeng; Wu, Yiming; Wu, Huaming & Zhao, Zhidong. 2024, VGGM: Variational Graph Gaussian Mixture Model for Unsupervised Change Point Detection in Dynamic Networks, *IEEE Transactions on Information Forensics and Security*.

Zhou, Weixun; Newsam, Shawn; Li, Congmin & Shao, Zhenfeng. 2018, PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval, *ISPRS journal of photogrammetry and remote sensing*, vol. 145, , pp. 197–209.