Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

Saeid Shamsaliei

Improving the semantic segmentation of historical aerial images of riverscapes using both data-centric and model-centric approaches.

Master's thesis in Informatics
Supervisor: Odd Erik Gundersen
Co-supervisor: Knut Alfredsen and Jo Halvard Halleraker
June 2022

NTNU
Norwegian University of
Science and Technology

Saeid Shamsaliei

Improving the semantic segmentation of historical aerial images of riverscapes using both data-centric and model-centric approaches.

Master's thesis in Informatics
Supervisor: Odd Erik Gundersen
Co-supervisor: Knut Alfredsen and Jo Halvard Halleraker
June 2022

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Abstract

Human developments are putting pressure on the riverscapes which endanger the conservation of the land types and biodiversity. To assess the changes and investigate the restoration potential, it is essential to understand the alteration of rivers over time. This assessment is challenging due to lack of data. Even though historical images of riverscapes exist, understanding the evolution of the river requires land cover classification of these riverscapes images. Manual classification of the data is a tedious and time-consuming process and can be considered as the bottleneck of the analysis of evolution of the rivers.

Recently, deep neural networks achieved superior performance on image processing tasks. There exists research which formulated the land cover classification as a semantic segmentation task and solved it using convolutional neural networks. However, very few works focused on historical grayscale images and the results of these models need to be improved, so that these models can be used for a large-scale analysis.

This thesis is built on top of the previous work done by Dalsgård (2020), which proposed a model based on deep convolutional neural networks to predict the semantic segmentation of historical aerial images of riverscapes in Norway. This research increased the MIoU of the test sets of the previous work by 9.53%, which means the error rate is reduced by 26.84%. This improvement was achieved by applying both data-centric and model-centric methods. The code of this thesis is available at: `https://github.com/SaeidShamsaliei/river-segmentation`

Preface

The research presented in this thesis is conducted in the Department of Computer Science of the Norwegian University of Technology (NTNU) in Trondheim, under super vision of Odd Erik Gundersen, from Department of Computer Science. Knut Alfredsen and Jo Halvard Halleraker from Department of Civil and Environmental Engineering were co-supervisor of the thesis.

My warmest thanks to my supervisor, Odd Erik Gundersen, for all your help and support throughout the research. Many thanks to my co-supervisors, Knut Alfredsen and Jo Halvard Halleraker, who instructed me and provided guidance. Special thanks to my friends and colleagues for their kindness and helping me throughout the thesis by providing feedback and suggestions. To my partner, Emilie Tverå, my sincerest appreciation for being so helpful, supportive, patient and understanding. I am very grateful. To my family who unconditionally loved and supported me and made it possible for me to pursue graduate studies. I am forever thankful.

Table of Contents

List of Tables

List of Figures

Abbreviations

OA	=	Online augmentation
RA	=	Rotation augmentation
SWA	=	Stochastic weight averaging

Chapter 1

Introduction

This chapter introduces the context of the thesis, and describes the research approach, the research objectives and the results.

1.1 Background and Motivation

Human developments are constantly and increasingly changing the shape of natural habitats, and riverscapes are not an exception (Grill et al., 2019; Grizzetti et al., 2017). Activities such as hydropower development, development of flood protection construction, urbanization and development of roads and gravel mining are examples of such developments which put pressure on rivers and river surroundings. This pressure has an immense impact on the ecosystem services provided by the river and leads to reduction of biodiversity and habitat loss (Wohl, 2019). Therefore, there is a need for restoration of the environment. To emphasize the importance of restoration, the United Nations declared the coming decade as the decade of ecosystem restoration. Understanding the alteration of river landscape over time is essential to assess the changes and recognize restoration potentials (Alfredsen et al., 2021a). However, this can be a challenging task since it relies on the historical state of the river of which available data is scarce. For many rivers, there exists aerial photography that dates back in time, which could be utilized for this purpose. When using old aerial imagery, there is still a need for manually pre-processing and mapping the images into desired habitats to make the images suitable for the assessment process (Piégay et al., 2020). These manual works are tedious as well as time consuming and can be considered as the bottleneck of understanding and analyzing the evolution of rivers through time. This signifies the need for development of a method that automatically and reliably provides the required data for the assessment (Piégay et al., 2020).

Deep learning is a branch of machine learning that utilizes deep neural networks. Back in 2012, Krizhevsky, Sutskever and Hinton trained a deep convolutional neural network to classify the images of the ImageNet challenge and reduced the top-5 error rate from 26.1% to 15.3% (Krizhevsky et al., 2012). This successful demonstration along with the work done by Cireşan et al. (2010), changed the direction of machine learning and Arti-

ficial Intelligence in general. Ever since, deep learning methods have gained tremendous attention and are now considered to be state-of-the-art in various domains such as speech recognition, language modeling, image classification, object detection and image semantic segmentation (Zhang et al., 2020; Brown et al., 2020; Takase and Kiyono, 2021; Zhai et al., 2021; Zhang et al., 2022; Yuan et al., 2019a). This growth is due to the advancement of deep learning methods (LeCun et al., 2015) plus the increase of computational power of hardware devices and availability of large scale datasets. Thus far, several studies have investigated and developed new methods to improve the deep learning methods. Some designed new neural network architectures (Chen et al., 2017b; Vaswani et al., 2017; Nekrasov et al., 2018). Other works proposed new loss functions (Zhao et al., 2019; Berman and Blaschko, 2017) or new methods to improve the optimization process (Ruppert, 1988).

While data plays an integral role in machine learning and specifically deep learning, it has been overlooked by the design of more sophisticated models. Recently there has been an increase in the awareness of the importance of data as paramount figures in AI, e.g. Andrew NG[1], are promoting the concept of Data-Centric AI (Data-Centric, 2021; Ng et al., 2021). Data-centric AI, unlike conventional model-centric AI, does not treat the data to be fixed and attempts to optimize the data set to train models.

Mapping the aerial images into the desired habitats can be formulated as a semantic segmentation problem. The goal of semantic segmentation is to segment images into predefined classes by providing dense per-pixel prediction. The availability of Remote Sensing (RS) data such as aerial and satellite images of natural landscapes, accompanying the superior performance of deep learning on image semantic segmentation tasks, suggests that neural networks will be fitting for this task. In fact, there exists a considerable number of studies about the applications of deep learning on semantic segmentation of RS data (Lu et al., 2020; Kotaridis and Lazaridou, 2021; Gebrehiwot et al., 2019). However, most of the research is done on colored images with 3 or more channels whereas historical images are grayscale, which only has 1 channel. On the other hand, since data-centric AI is in its initial stages, very little research is done on its application in image semantic segmentation.

1.2 Problem outline

This thesis focuses on improving the semantic segmentation of historical aerial images of riverscapes in Norway using both data-centric and model-centric approaches. The thesis does not propose new deep learning architecture for semantic segmentation or using self-supervision or active learning. Instead, it focuses on improving the performance of commonly used architectures using both data-centric and model-centric methods.

Even though many research attempts to improve the accuracy and reliability of semantic segmentation of RS images. Most of the research is focused on RGB images, where RGB stands for red, green and blue channels, with very little focus on grayscale or historical images. Historical aerial images introduce multiple challenges to semantic segmentation. These images are grayscale and thus algorithms cannot use color information

[1] https://www.andrewng.org

and need to predict only based on the intensity value of images. In addition, historical images are taken using analog technology and afterwards digitized and georeferenced by Kartvarket[2]. This resulted in some areas with low quality images which makes it difficult even for a domain-expert to outline the different land types in an image. Another challenge is having images of different areas taken at various times; these spatial and temporal differences make it challenging to learn the pattern of each class. Other challenges such as class imbalance (Krawczyk, 2016) and multi-scale objects, meaning the same class of objects might have different scales, (Wang et al., 2021a) are not restricted to historical images and are pervasive in most RS datasets. Besides RS images can have large sizes, images provided by Kartverket (Kartvarket, 2021) are mostly 8000×6000 pixels. Having large size input introduces GPU memory issues when applying neural networks for these images.

This thesis is built on top of the previous master's thesis done by Dalsgård (2020). Previous work used deep neural networks for semantic segmentation of historical aerial images which will be referred to as *baseline model*. The baseline model was a U-Net architecture (Ronneberger et al., 2015) with VGG16 (Simonyan and Zisserman, 2014) encoder. The dataset that was produced in the previous work is also called the *initial dataset*. Figure 1.1 briefly illustrates the process of training the baseline model. Despite the good performance of the baseline model, there is still need for improvement to have a suitable model for large scale analysis. As the end goal of this model is to provide the segmentation maps for all the riverscapes in Norway in various time-points, it is crucial to have a reliable model with the ability to generalize well among different river images over different time points. This opens a room for improvement of previous work by using data-centric as well as conventional model-centric approaches to provide reliable maps for large scale analysis of evolution of the rivers in Norway.

1.3 Objective and Research Questions

The objective of the thesis is defined as:

O: *Find out how performance of existing semantic segmentation of historical aerial images of riverscapes of Norway can be improved in order to provide an out of the box tool for large scale analysis of evolution of rivers in Norway.*

The main objective of this work is to improve the semantic segmentation of historical aerial images of riverscapes. This is important due to the final goal of the model. The initial attempts to achieve this objective proved to be unsuccessful, which indicated that this task is more challenging than initially expected. To reach the objective, this thesis attempts to take advantage of the data-centric paradigm as well as the conventional model-centric methods. Data-centric paradigm considers the model to be fixed in the training cycle Figure 4.11. In this paradigm improvement is achieved by manipulation of data which is also known as data tuning. Having this perspective, the thesis objective can be divided into two research questions.

RQ1: *How can Data-Centric AI be used to improve the semantic segmentation results?*

[2]https://www.kartverket.no/en/

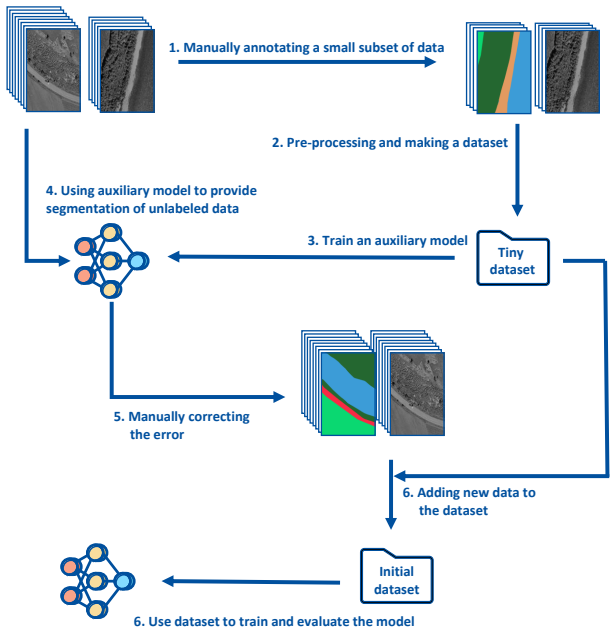


Figure 1.1: Overview of the previous work.

As data-centric AI is a relatively new approach, it is integral to find out what data-centric methods are suitable for the thesis objective, how these methods should be employed and ultimately what is the impact of their usage on the performance of semantic segmentation. This implies that appropriate methods and metrics should be found to show the effectiveness of data-centric methods on both model performance and data. Unlike model performance metrics which are standardized, data quality measurements are considered to be an open question in the community (Aroyo et al., 2021).

RQ2: *Can Model-Centric methods improve the results of semantic segmentation further?*

As mentioned above, the objective of the semantic segmentation model is to provide processed data for large scale analysis of the evolution of all rivers in Norway. As a result, it is important to eliminate the error rate of the model as much as possible. This research question seeks to answer how conventional methods such as new model architecture and optimization techniques can help the performance of the model further.

To answer **RQ1**, first, the baseline model was reproduced. Through reviewing the data-centric literature along with quantitative and qualitative error analysis of the results of the reproduced model, which is elaborated further in chapter 7, suitable data-centric methods for the task were selected, applied and evaluated. **RQ2** was answered by testing different architectures and methods through a model comparison study to see the effect of each method and choose the best performing.

1.4 Research Approach

During the time in which the thesis has been carried out, several methods have been tested. However, initially, none of them led to fulfillment of the research objective. This along with the new shift of attention from model-centric to data-centric AI, led to formulating the the two research questions. This thesis consists of three phases as follows:

1. **Reproduction and background study:** In this phase, the previous study Dalsgård (2020) was reproduced which is used as baseline in this thesis. Next, several model-centric methods were tested to improve the performance of the baseline. After none of the methods led to desired improvement, error analysis was conducted. In addition to the error analysis, a background study was done in this phase and related research in semantic segmentation, remote sensing data and data-centric AI were reviewed. Furthermore, participating in the Neurips workshop of data-centric AI helped to acknowledge the importance of this new paradigm and led to formulating the first research question **RQ1**. During the background study, several model-centric methods were found which led to improvement in similar studies. This led to designing the second research question **RQ2** to test whether these methods will help to achieve the research objective.

2. **Development and implementation:** During this phase, the thesis experiments were designed and runtime environments to conduct the experiments were developed. Additionally, the data-tuning task was done in this phase.

3. **Analysis and evaluation:** Ultimately, results of experiments performed during the previous phase were analyzed and compared with each other.

The type of research conducted in this thesis is *problem solving* (Phillips, 2010), since the work started with a real-world problem and attempted to find the solution for the problem. As was discussed earlier, in order to provide segmentation maps for all the available rivers in Norway, the baseline semantic segmentation model needed to be improved so that the performance of the model becomes more reliable. This work sought to address this problem with focus on data-centric AI, which later opens more possibilities to utilize model-centric methods more effectively.

1.5 Research Contributions

This thesis has three contributions that are briefly listed here. More detailed explanations of contributions are presented in section section 7.4.

C1: *Demonstrating a data-centric approach which improves the performance of existing semantic segmentation of historical aerial images of riverscapes and increased the average MIoU of test sets by 7.05%, meaning the error rate is reduced by 19.84%.*

Being a new approach in the machine learning community, the data-centric AI is an integral part of this thesis, the first contribution is demonstration of effectiveness of data-centric methods in semantic segmentation of historical aerial images of riverscapes.

C2: *Creating a fully annotated semantic segmentation dataset of historical aerial images of riverscapes in Norway with detailed manual annotations.*

Created dataset in this work includes 87 high resolution aerial images of different rivers taken at different times. 83 images have the resolution of 8000×6000 and the resolution of other 4 images are 6400×4800 pixels.

C3: *Improving the performance of semantic segmentation further by employing a set of model-centric methods after applying the data-centric methods. This improved the average MIoU of baseline on all test sets by 9.53%, which means the error rate reduced by 26.84%.*

This contribution was done by conducting a model comparison study which led to finding methods which improve the performance of the semantic segmentation model that was demonstrated in **C1**.

1.6 Thesis structure

This thesis is divided into the eight chapters, the first chapter contains the introduction to the thesis. In the second chapter background theory and the central topics of the thesis are introduced. The third chapter presents the state-of-the-art methods and studies related to the work. In the fifth chapter details related to the datasets and experiments of the thesis are described. The methods used in the thesis are presented in chapter five. The sixth chapter presents the results. Qualitative and quantitative evaluation of the results are done in the chapter seven. Ultimately, chapter eight concludes the thesis and suggests areas for future research.

Chapter 2

Background

Chapter 2 introduces the theory and topics discussed in this thesis. It starts with an overview of topics related to neural networks, which forms a large part of the background, followed by uncertainty estimation in semantic segmentation. For a thorough introduction into the topics, the reader is referred to (Goodfellow et al., 2016; Murphy, 2022; Gonzalez and Woods, 2008).

2.1 Neural networks

Recently, neural networks have become the dominant approach used in computer vision tasks such as semantic segmentation. Being loosely modeled after the human brains, artificial neural networks mimic how biological neurons are connected and communicate with each other. Similar to biological neurons, an artificial neuron receives many different inputs from other units and outputs its activated value. A basic neuron contains a set of weights and a bias which is used to calculate the weighted sum of input values of the neuron. This can be expressed as $o = \sum_{i=1}^{m}(w_i x_i) + b$ where x_i is the i^{th} input of the layer, w_i is the corresponding weight and b is the bias. This weighted sum o can be considered as a linear regression of the inputs. Hence, in order to be able to learn non-linear relations, a non-linear activation function f is needed to be applied to produce the final output $z = f(o)$. Rectified Linear Unit (ReLU) is a simple activation function that is commonly used in practice and is defined as $f(o) = max(0, o)$.

Feed-forward neural networks can be considered as the simplest models of neural networks that are widely used in practice. As the name suggests, feed-forward neural networks send the information in the forward direction. These networks usually consist of multiple layers containing various numbers of neurons and successive layers are fully connected to each other. The number of layers in the network is called the *depth* of the network and the number of neurons in each layer is called the *width* of that layer. First layer in the network is called *input layer*, the output of the input layer is fed into the intermediate layers which are known as *hidden layers*. The final output of the hidden layers then is fed into

the final or *output layer* to produce the final output of the network. A simple feed-forward network is shown in the Figure 2.1

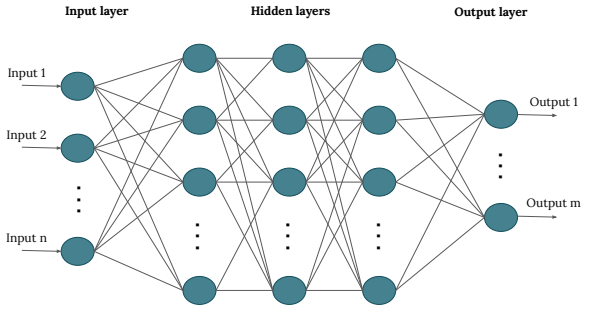


Figure 2.1: A simple feed-forward neural network, inspired by (Bre et al., 2018).

It is possible to formulate the neural networks as a function $y = f(x; \theta)$ where θ are the parameters of the network, i.e., weights and biases. Neural networks are used in supervised learning tasks. In supervised learning, given a set of training data D, containing pairs of (x, y) where x is the input vector and y is the output target, it is desired to approximate the function f^* such that $y = f^*(x)$, by learning the parameters θ in a way that $y = f(x; \theta)$ is the best function approximation for f^*. In the training process, a loss function is defined to evaluate the error of the network with the current parameters given the training set. This makes it possible to define the training process as an optimization problem. The goal of training then can be defined as adjusting the parameters θ with the objective of minimizing the loss value for D. This is done by iteratively calculating the partial derivative of the loss with respect to each of the parameters θ in the back-propagation phase and using algorithms such as stochastic gradient descent (SGD) (Ruder, 2016) to update the parameters θ.

Categorical cross entropy is a very popular loss function in semantic segmentation and is defined in Equation 2.1 where p_i is the softmax probability of output vector, which is defined in Equation 2.2. Many other loss functions are proposed for semantic segmentation, for instance Lovász-Softmax loss proposed by Berman and Blaschko (2017) directly minimizes the Jaccard index. Nevertheless, this work does not cover them and the reader is referred to the survey written by Jadon (2020) for more thorough information. Jaccard index also known as intersection over union (IoU) is a widely used metric for quantifying the performance of semantic segmentation. In multi-class semantic segmentation, the mean intersection over union over all the classes (MIoU) is calculated. Assuming the vector of prediction of the network is y^* and ground truth is $\tilde{y}$, the equation for MIoU is Equation 2.3.

$$CCE_{loss} = -\sum_{i}^{|C|}(w_i y_i log(p_i)) \tag{2.1}$$

$$p_i = \frac{exp(z_i)}{\sum_j^{|C|} exp(z_j)} \tag{2.2}$$

$$MIoU = \frac{1}{\mid C \mid} \sum_{c \in C} \frac{(y^* = c) \cap (\tilde{y} = c)}{(y^* = c) \cup (\tilde{y} = c)} \tag{2.3}$$

Convolutional neural networks are the most popular type of neural networks in image processing, the following describes these networks in more detail.

2.1.1 Convolutional neural networks

Convolutional neural networks (CNN) are neural networks that apply convolution in at least one layer instead of matrix multiplication. Ever since Lecun et al. (1998) used convolutional neural networks for Optical Character Recognition (OCR), CNNs have become the dominant method used for image processing tasks such as image classification, object detection and semantic segmentation. A simple 2-dimensional convolution is defined as

$$S(i,j) = (K * I)(i,j) = \sum_m \sum_n I(i-m, j-n) K(m,n) \tag{2.4}$$

Where S is the output of the convolution or *feature map*, I is the input image and K is a $m \times n$ kernel. Convolutional neural networks are inspired by the visual cortex in animals (Kim et al., 2016). CNNs have hierarchical structure, early convolutional layers extract low-level features from the input image like edges of objects, etc. Deeper layers extract more high-level and more abstract features which helps the interpretation of the context of the image (Lee et al., 2011). In addition, using hierarchical structure allows CNNS to have wide receptive field. This is done by increasing the number of kernels in the filters while decreasing the dimensionality of feature maps in the deeper layers by downsampling methods such as pooling operation or striding. The pooling operation summarizes the input feature maps to reduce the dimensionality of them, max-pooling is the most popular pooling method in semantic segmentation which samples only the largest value of the input feature map in each kernel window.

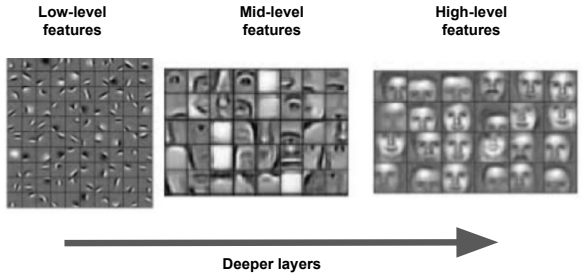


Figure 2.2: The features extracted in a Convolutional neural network, the deeper layers extract more abstract and high-level features. (Lee et al., 2011).

In image processing, an image can be considered a 2-D input with 1 channel for grayscale images and 3 channels for RGB images. This means that convolution should be applied to the multi-channel input data, additionally, to reduce the computations, downsampled convolutions can be applied with setting the stride s to be larger than 1. This means that the kernel moves every s pixel at a time. The Equation 2.5 defines the downsampled convolutions for multi-channel input:

$$Y_{z,i,j} = C(K, X, s)_{z,i,j} = \sum_{l,m,n} X_{l,(i-1)\times s+m,(j-1)\times s+n} K_{z,l,m,n} \tag{2.5}$$

Where X is the input feature map with l channels and Y is the output with z channels and stride of s. Compared to the feed-forward dense neural networks, CNNs take advantage of parameter sharing which reduces the number of parameters of the network. In other words, kernels in CNNs are reused in all the positions of the image which means less parameters are required. Moreover, CNN introduces useful inductive biases such as translation invariance, which is immensely helpful in semantic segmentation and, CNN allows for having variable input size.

As described above, in standard CNNs, max-pooling or striding is applied in consecutive layers which results in significant reduction of the spatial resolution. Proposed by Chen et al. (2014), **dilated convolutions**, aka atrous convolutions, increase the receptive field without any memory or computation overhead. This is done by inserting zeros among non-zero coefficients of the filter kernels. For example, a 3×3 size filter can be converted to a 5×5 filter with only 9 trainable weights. Figure 2.3 compares the standard CNN approach with atrous convolution which shows the advantage of dilated convolutions.

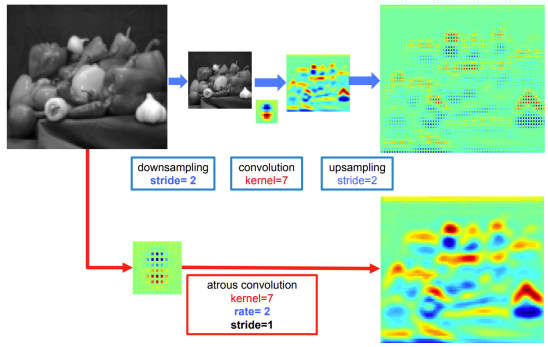


Figure 2.3: Comparison of the sparse feature extraction of standard convolution applied on the downsampled feature map (top row) with the dilated convolution with the dilation rate of 2 applied on the high-resolution input (bottom row), from (Chen et al., 2017a).

As can be seen so far, assuming a convolutional layer with kernel of $m \times n$ size, each output value depends on the input values of all channels, in the same $m \times n$ window. Originally in the work of Chollet (2017), **depth-wise separable convolutions** were introduced to reduce the number of parameters. The idea behind this type of convolution is to apply a depth-wise convolution followed by a 1×1 filter to handle the depth. Depth-wise convolutions simply apply each filter channel to the corresponding input channel. This leads to

a decrease in the number of parameters which facilitates having deeper layers and reduces the risk of overfitting.

In the next chapter, Encoder-Decoder convolutional neural network which is the most common CNN based architecture used for semantic segmentation is described.

Encoder-Decoder convolutional neural network

Unlike image classification which predicts a single class for each input image, in semantic segmentation, each pixel of the input image needs to be classified. This means that the output of the network should be the same size as the input. As the name suggests, Encoder-Decoder architecture is composed of two parts. The encoder E compresses the input x into latent space representation $z = E(x)$ which can be considered as a feature vector. Afterwards z is fed into the decoder to predict the output $y = D(z)$. In the encoder, features of the input image are extracted by applying convolution and downsampling using striding or pooling. This means that in the deeper layers, feature maps have lower resolution but higher depth. The latent space usually consists of several channels but the dimensionality of it is small. That being the case, the decoder needs to upsample the latent vector. This is usually done by upsampling the input feature map using methods like max-unpooling or transposed convolutions. As the name suggests, transposed convolution reverses the operation of convolution and is carried out for upsampling. In practice, this is done simply by modifying the input feature maps and applying standard convolution to the modified input. The main drawback of encoder-decoder models is the loss of spatial information in the encoding phase. Many studies attempted to mitigate this issue and a selection of them are presented in the section 3.1.

In addition, it is common to use the convolutional layers of a known image classification architecture such as VGG16 (Simonyan and Zisserman, 2014) or ResNet (He et al., 2016) as the encoder. In the following, an introduction to the deep residual learning, which is the backbone of ResNet architecture, is provided. ResNet is a deep learning architecture based on convolutional neural network and is used for image feature extraction. ResNet was used as the primary encoder for models used in this thesis. In subsection 4.1.1 the encoder used in this work is presented with more details.

Deep residual learning

After impressive performance of models such as AlexNet, GoogleNet and VGG, one could assume that stacking more layers and having a deeper model leads to more accurate performance. However, in practice this was not the case as illustrated in Figure 2.4. It is noteworthy that this effect is not overfitting since deeper networks have difficulties decreasing the training error. This is counter intuitive since if the deeper layers just act as identity transformation, deeper models should be at least as good as shallower models on the training data. Vanishing gradient is one of the causes of this problem since gradients propagate through the network and the first layers of deeper networks get smaller gradients which hampers the learning process. The work of (He et al., 2016) introduced the deep residual framework to address this problem and drastically influenced the design of deep learning methods. The idea is simple, connecting the input of the residual layer to the output of it, this means that this layer needs to learn residual value instead of a totally

new output. Figure 2.5 depicts a residual layer, the input is x and the output is $F(x) + x$, parameters of F will be learned during training. In the original paper, element-wise addition is proposed to be used as a connection mechanism; however, it is possible to use other mechanisms such as concatenation of inputs. Another way of looking at the residual layers is as a skip connections mechanism. By connecting input of the block to the output, skip connections make pathways for gradients which help the vanishing gradient problem. In the methods chapter the details of specific ResNet architecture that is used in this thesis are described.

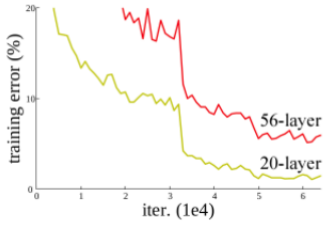


Figure 2.4: Comparison of training two model with 56 and 20 layer depth without using residual connections. The deeper model performs worse compare to the shallow model (He et al., 2016)

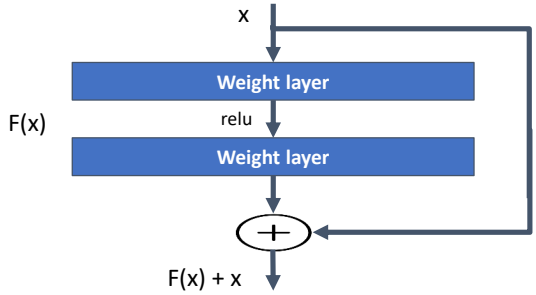


Figure 2.5: A simple residual layer (Chen et al., 2017a).

2.1.2 Regularization

In machine learning, bias corresponds to the assumptions of the learning algorithm, therefore, high bias error can be considered as high training error. This is also referred to as underfitting. On the other hand, high variance error indicates that the learning algorithm is not able to generalize well on test data. Overfitting occurs when the performance of the model on the training set is significantly better than the test set. To tackle the bias-variance trade off, several regularization methods have been developed. Some well-known regularization methods that are used in this thesis are:

Parameter Norm Penalization

Neural networks tend to overfit when the absolute value of a set of parameters, i.e., weights, is high. One way of addressing this issue is to penalize the network when weights get high values. This can be done by adding a function of parameters in the loss function. Two examples of this method are $L1$ and $L2$ regularizations. $L1$ adds the sum of absolute value of the weights to the loss whereas $L2$ adds the sum of squared weights.

Data Augmentation

Data augmentation is a common practice of mitigating the overfitting problem by artificially increasing the size of a dataset. Increasing the size of the dataset can be done by either transforming existing data or generating new data samples. Transforming the existing data can be done by simply applying rotation, mirroring, flipping, shearing, squeezing, etc. to the data or by changing the intensity of the image, noise injection, applying filters such as gaussian blur etc. As it was described, convolutional neural networks are transition invariance (Krizhevsky et al., 2012), however aforementioned transformations increase the robustness of the model. These transformations are dataset-specific. For example, flipping and mirroring which was used at Krizhevsky et al. (2012) cannot be used for MNIST since flipping a number symbol will alter the semantic of it. Increasing the data samples makes the dataset more complete and improves the robustness of the model which will improve the generalization ability of the model. There are more advanced methods such as adversarial training, utilizing generative models and feature space augmentation.

Data augmentation techniques related to semantic segmentation are discussed in section 3.4.

Dropout

Proposed by Srivastava et al. (2014), dropout can be considered as a regularization method which zeroes out the output of some nodes of the network architecture at random. This leads the network to learn more robust representation of the input and utilizes the whole architecture instead of solely relying on a subset of nodes. In the image processing, some work used different type of dropout called *spatial dropout*. Spatial dropout randomly assigns zero to one of the feature maps instead of nodes.

2.1.3 Transfer learning

In the domain of machine learning, transfer learning is a technique where knowledge learned from a relatively large dataset is used to achieve better generalization on a relatively smaller dataset (Goodfellow et al., 2016). Fine-tuning is arguably the most common transfer learning practice in computer vision. This approach starts with a model pre-trained on a task which has sufficient amount of data, *source task*, then trains this pre-trained model on the *target dataset* (Guo et al., 2018). In image classification, ImageNet is the most popular source task, this is because ImageNet contains millions of images and models that are trained on it have powerful feature extraction ability.

2.2 Uncertainty Estimation of semantic segmentation

Neural networks are capable of mapping complex high dimensional data to arbitrary output vectors by learning powerful representations. However, they lack proper explanation of the mapping. Uncertainty estimation of deep learning models becomes more important when these models are being deployed for applications used in the real world, like this work. In this section methods used for uncertainty estimation of deep neural networks are briefly explained. In the section 4.6 the method that was used to estimate the uncertainty in this thesis is presented.

One can model uncertainty in two major types, *Epistemic* and *Aleatoric uncertainty*. Epistemic uncertainty captures the uncertainty of the model and describes what the model does not know due to not having suitable training data. Aleatoric uncertainty on the other hand captures intrinsic noise of the data such as malfunctioning of a sensor while gathering the data. Generally, there are three well-known methods to quantify the uncertainty of neural networks:

- *Softmax Entropy*, the most common metric for uncertainty, is the entropy of softmax distribution of the output of the network. Assuming $p(y|x, \theta)$ is the softmax distribution of the output of the network where y is the output, x is the input array and θ are the network parameters. Then the uncertainty is approximated as $\mathbb{H}[p(y|x, \theta)]$ where $\mathbb{H}$ is the entropy function defined as Equation 2.6:

$$\mathbb{H}[p(x)] = -\mathbb{E}_{x \sim p(x)}[log(p(x))] \tag{2.6}$$

  Intuitively, entropy indicates how broad the distribution is. In cases where softmax output is close to uniform distribution, which means the probability of all the possible classes are similar and there is no dominant class, entropy reaches the maximum. Conversely if softmax distribution peaks at one class, entropy of the distribution is minimum. Softmax entropy is known to capture aleatoric uncertainty and not epistemic uncertainty.

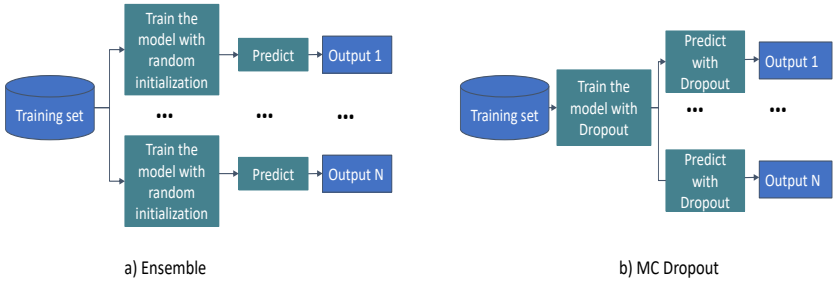


Figure 2.6: Deep ensemble and MCDropout uncertainty estimation.

- *Deep Ensembles*, another intuitive approach to estimate the uncertainty, utilizes ensemble of neural networks. Entropy of ensemble predictions can be used to capture

the uncertainty. Lakshminarayanan et al. (2017) trained multiple neural networks by only setting random initialization and data shuffling to be different among different models. The ensemble is considered to be a uniformly-weighted mixture model, and the ensemble prediction is approximated as a Gaussian with mean and standard deviation equal to the mean and standard deviation of the mixture.

- *Monte Carlo Dropout*, Gal and Ghahramani (2016) showed that Bayesian inference can be approximated by simply applying dropout during the test time. By applying the forward pass multiple times while the dropout layers are activated, the resulting softmax outputs can be used to compute the approximation of the uncertainty by either predictive entropy or mutual information. Predictive entropy corresponds to the predictive uncertainty which is formed of both epistemic and aleatoric uncertainty while mutual information captures the epistemic uncertainty. Mutual information can be computed by Equation 2.7

$$\mathbb{I}[x; y] = D_{KL}(p(x, y) \parallel p(x)p(y)) \tag{2.7}$$

Figure 2.6 illustrates an overview of deep ensemble and MC Dropout uncertainty estimation.

Chapter 3

State of the Art

This chapter reviews the state of the art of research related to semantic segmentation of remote sensing data together with data-centric AI.

3.1 Semantic Segmentation

Deep learning methods dominated the semantic segmentation task due to not needing biased manual feature extraction along with great performance (Yuan et al., 2019b; Alfredsen et al., 2021b; Ratajczak et al., 2019). Huge amount of work has been done in the application of deep neural networks in semantic segmentation. In this section a brief overview of important studies done in this field is provided. For more thorough information, the reader is referred to the surveys done by Minaee et al. (2021) and Zhou et al. (2022).

Long et al. (2015) proposed Fully Convolutional Network (FCN), the first model based on convolutional neural networks to semantically segment images. The authors modified well known existing CNN architecture such as GoogLeNet and VGG16 by changing the last fully-connected layer with convolutional layers to output segmentation map. In order to extract feature maps with more semantic information while not losing the appearance information, feature maps of the final layers are upsampled and concatenated with feature maps of earlier layers which plays the role of skip connection. FCN showed that end-to-end training of deep neural networks for semantic segmentation can be achieved. Many works were devoted to improving the FCN architecture. For example, Liu et al. (2015) proposed ParseNet which utilized the global feature vector made by global pooling to improve the incorporation of the context in the final segmentation.

Most recent work in semantic segmentation features encoder-decoder architectures (Samy et al., 2018). U-Net is one the most widely used models based on encoder-decoder architecture which was originally proposed by Ronneberger et al. (2015) to segment dental X-ray images. U-Net added skip connection between encoder and decoder layers to aid the recovery of the details in the decoder. HRNet is another recent model with encoder-decoder architecture which was proposed by Sun et al. (2019). The key of HRNet is preserving the high-resolution representation during the encoding process. This

network architecture consists of parallel high-to-low resolution sub-networks. Output of sub-networks with different resolutions but same depth are connected with each other to exchange information across resolutions. Figure 3.1 illustrates the connection among these multi-resolution sub-networks. HRNet is used as backbone of many new state-of-the-art models (Yuan et al., 2019b; Huynh et al., 2021).

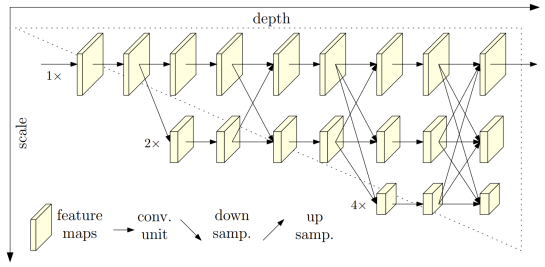


Figure 3.1: HRNet architecture (Sun et al., 2019)

DeepLabs are another group of architectures developed by Google, and are among the most popular architectures. Each version has improved the previous one by proposing a novel method. DeepLabV1 proposed using dilated convolutions to prevent reduction of resolution (Chen et al., 2014). DeeplabV2 used Atrous Spatial Pyramid Pooling (ASPP), which applies several filters with different dilation or sampling rate to the input feature maps and fusing them together to capture objects with different shapes. To mitigate the localization problem Dense Conditional Random Field, a probabilistic graphical model, is applied to the final output of the network (Chen et al., 2017a). DeepLabV3 applied depth-wise separable convolutions to improve the computation efficiency (Chen et al., 2017b). DeepLabV3+ can be considered as an encoder-decoder architecture. It extended the DeepLabV3 by adding an effective decoder to refine the object boundaries further (Chen et al., 2018).

Another category of popular architectures take advantage of the multi-scale processing scheme which is an old idea in image processing. Feature Pyramid Networks (FPN) (Lin et al., 2017) is one the most notable models of this group. Even though it was originally proposed for object detection, FPN proved to be effective for semantic segmentation as well (Seferbekov et al., 2018a). FPN consists of *bottom-up* and *top-down pathways* along with *lateral connections*. Lateral connections can be considered as skip connections. The bottom-up is feed-forward computation of backbone CNN to extract features similar to encoders. It decreases the resolution of the feature maps in multiple stages and output feature maps of each stage is used for enriching the top-down pathway by using the lateral-connections. The top-down pathway constructs higher resolution predictions from previous predictions, which have more semantic information, by upsampling. Feature maps of corresponding bottom-up and top-down pathways merged using element-wise addition. Afterwards, a 3×3 convolution is applied to the merged feature maps to generate the final feature maps used for final prediction.

Another type of semantic segmentation models are based on Transformers. Transformers proposed in Vaswani et al. (2017), is fairly new architecture that revolutionized

the field of Natural Language Processing (NLP). Despite transformers achievements in Speech Processing and NLP tasks, Convolutional neural networks are still dominant in computer vision. Visual transformers (ViT), modified the original NLP Transformers and introduced a *patchify* layer to split input into patches. Despite achieving good results in image classification tasks, ViT has quadratic complexity with respect to input image size and does not have any image specific inductive biases. Many works tried to address these issues and to make Transformers behave more similar to CNNs. Swin Transformer (Liu et al., 2021) is a notable example.

In semantic segmentation, some work is done to develop architectures purely based on transformers such as Segmenter by Strudel et al. (2021) and SegFormer by Xie et al. (2021). Another approach is to use a hybrid of CNN and Transformers to take advantage of the properties of convolutional layers such as translation invariance. Chen et al. (2021) proposed TransUNet, a hybrid model which modified the encoder-decoder architecture of U-Net by adding a Transformer model after the CNN encoder. Transformer encodes the feature map output of CNN in order to extract the global context which then is upsampled by the CNN decoder. In the original paper, ResNet50 with ViT constructed the encoder of TransUNet. Liu et al. (2022) showed that convolutional layers are able to compete favorably with transformers in image classification tasks. Moreover, most vision transformer architectures comprise a massive amount of learnable parameters, for example ViT-G/14 contains 1843 million parameters and this number for DaViT-G is 1437 million. This indicates that these models are computationally expensive to train and huge amounts of training data is required to learn the model parameters and avoid overparameterization and consequently overfitting problem. This thesis only focuses on models that are purely based on convolutional neural networks and does not study the applications of transformer models.

The recent work of (Kirillov et al., 2020) is another notable approach in semantic segmentation. This work proposed to improve the predicted segmentation maps of the semantic segmentation model by employing classical computer graphics rendering algorithms and showed that this yields significant gains on common segmentation challenges.

The next section, focuses more on research done semantic segmentation of remote sensing (RS) data.

3.2 Semantic Segmentation of Remote Sensing Data

Until recently, classical machine learning methods such as support vector machine (SVM) (Cortes and Vapnik, 1995) has been the most popular approach for semantic segmentation of remote sensing data. (Ratajczak et al., 2019; Liu et al., 2017; Mountrakis et al., 2011). When deep learning became the dominant method in computer vision, the remote sensing community started to employ them for tasks such as land type classification, road extraction, etc. It is worth mentioning that in the remote sensing community, land cover classification and semantic segmentation are used interchangeably. Abundant work is done in the application of deep learning methods on semantic segmentation of remote sensing images, many of which proposed new architectures. This chapter mainly focused on the research conducted on DeepGlobe (Demir et al., 2018) and landcover.ai (Boguszewski et al., 2021) datasets. The reason for selecting those dataset is that the research community is actively

working on these two datasets and they are similar to the riverscapes dataset of this work. In the following section subsection 3.4.1 related datasets are described with more details.

Samy et al. (2018) proposed the NU-Net, a new architecture for land cover classification that captures more global information without losing the local details. This network features the Wide Field of View (FoV) module to process the input in multiple scales at once. The input to FoV is parallely downsampled with different sampling rates followed by applying a 3x3 convolution and up-sampling. The up-sampled outputs are then concatenated and a 1x1 convolution is applied to handle the depth and create the output. Element-wise addition is applied to the input and output as a skip-connection. Furthermore, weighted cross entropy was chosen as the loss function where weights are inverse of class percentage. NU-Net achieved 42.8% on MIoU on DeepGlobe land cover classification and placed at the top of the leaderboard of this competition. Seferbekov et al. (2018b) outperformed the previous work by using FPN architecture with ResNet50 as backbone (bottom-up module) with spatial dropout in the final layer before the before prediction. This work introduced a new loss function which is the combination of two terms, categorical cross entropy Equation 2.1 and weighted average of IoU where weighted assigned manually. This led to achieving 49.3% MIoU on DeepGlobe land cover challenge. DIResUNet is an architecture for semantic segmentation of RS proposed by Priyanka et al. (2022). This work modified U-Net with a dense global spatial pyramid pooling module (DGSPP) to help with global context information extraction. DGSPP is a stack of convolutions which is added at the deepset skip connection between encoder and decoder. This work improved the MIoU of U-Net baseline by 10% in the Landcover challenge. However, it is not mentioned if this achievement can be attributed to increasing the parameter space of the model or the architecture design.

Another group of works proposed to improve the optimization process. Rakhlin et al. (2018) showed that stochastic weight averaging (SWA) improves the performance of semantic segmentation of DeepGlobe dataset. In this work conventional U-Net with Resnet encoder was used as the model architecture and Lovasz-Softmax (Berman and Blaschko, 2017) was set as the loss function. This work showed that stochastic weight averaging led to improvement of their validation MiOU.

Most remote sensing images are high resolution. Due to the low memory capacity of current GPU cards, it is impractical to feed these images directly to the network. Patch processing and downsampling are ad-hoc ideas to avoid this issue as all the mentioned works so far are based on patch processing. The disadvantage of the patch processing is losing the global information that might be necessary for prediction and downsampling results in loss of detail. Chen et al. (2019) investigated the importance of surrounding context in the DeepGlobe dataset. To do so, the authors used both downsampling and patch processing on the DeepGlobe dataset to train the same models and compared the performance of two models on the validation set. For downsampling, large images, with resolution of 2048x2048, downsampled to have 500x500 resolution. For patch processing, large images cropped into 500x500 small images. The results of experiments show that downsampling achieves comparably better results. Furthermore, this work proposed a novel architecture to take advantage of the benefit of both patch processing and downsampling by combining the downsampling and cropping branch into one architecture called GLNet. Huynh et al. (2021) noted that there is a huge gap between the scale of downsampled images

and patches which makes it difficult to combine the information in a single feed-forward process. To address this issue, MagNet was proposed. Concisely, MagNet starts with segmenting the coarsest (smallest) scale of input image, then progressively refines the segmentation output by increasing the input scale from coarsest to finest to improve the detail of segmentation by incorporating local information. The detail of this architecture is presented in the Method section subsection 4.1.5. Currently, MagNet is the state-of-the-art in DeepGlobe land cover challenge. In contrast with previous work (Chen et al., 2019), experiments of this work showed that patch processing led to better performance in semantic segmentation of DeepGlobe dataset as FPN achieved 70.98% MIoU when patch processing was applied and 67.86% was the performance of downsampling.

Most of the current research is done on images with at least 3 input channels e.g., RGB images which makes it challenging to apply them directly to grayscale historical images. In terms of downsampling and global context information, it is important to consider that the DeepGlobe dataset contains more than 4000 large images which provides large enough training samples for training a neural network when downsampling is applied. On the other hand, Initial dataset of this thesis only contains disconnected patches which makes it not feasible to use GLNet or MagNet model for training.

3.3 Data-Centric AI for Semantic Segmentation

Regardless of established understanding that data is a vital part of AI, it is the most undervalued part of the AI ecosystem (Aroyo et al., 2021). It has been a common saying that ”data is oil”. If that is the case, refineries to optimize the data to be used more effectively are missing. Recently, the machine learning community started to pay specific attention to data (Whang and Lee, 2020). In 2021, a Data-Centric AI competition was introduced by Ng et al. (2021) with the goal of improving an image classification task only by modification of training data. Additionally, The Conference and Workshop on Neural Information Processing Systems (NeurIPS) 2021 held the Data-Centric AI workshop with 100 accepted papers [1]. Being a novel approach, works that are done in Data-Centric AI are scattered which makes it difficult to present them in an organized manner. The following briefly presents the main trends in Data-Centric AI to give an overview of the current research in the field that can be inspired for the purpose of this thesis. Data-Centric AI can be defined as the discipline of engineering the data used in AI systems in a systematic manner [2]. Aroyo et al. (2021) introduced four key properties to achieve data excellence derived from Software Engineering.

Reliability: Consistency, replicability, and reproducibility of the dataset. In this thesis, how reliable and reproducible the riverscapes dataset and the data pipeline are when this dataset is being used for training a data-driven model such as neural network.

Maintainability: Related to maintenance of the data. In this thesis, how well the riverscapes dataset is stored and what software and hardware solutions are being used to store the data in a reliable manner.

[1] https://nips.cc/

[2] https://datacentricai.org/

Fidelity: How well the dataset represents the real world. In this thesis, how well the dataset of the work represents the historical images of riverscapes in Norway.

Validity: How well the phenomena captured by data is explained by data itself. In this thesis, the quality of ground truth of rivers is related to validity.

Currently there are no standardized metrics which quantify the validity and fidelity of datasets.

Some studies focused on designing Data-Centirc methods to create benchmarks. Schmarje et al. (2021) described a Data-Centric image classification benchmark for acquiring consistent labels by assigning soft label instead of conventional hard label for each image where soft label is a distribution over all the classes. Having a soft label is helpful because it captures the uncertainty of images and the annotation process, such as disagreement of annotators. Kiela et al. (2021) introduced a platform to create the NLP benchmark datasets more dynamically, resulting in more informative and robust benchmarks. In this platform, annotator's objective is to create samples which are difficult to classify by the target network but easy for another human agent.

Another group of studies assessed or modified the existing datasets. Some studies devoted to fairness and privacy issues in mainstream image datasets. Yang et al. (2020) investigated the fairness in the "person" subtree of ImageNet. They identified 1593 labels of 2832 labels to be unsafe. Unsafe labels mean if they are sensitive or offensive, e.g. "racist". Moreover, 1080 labels were identified as non-imageable such as "vegetarian". Same group of people recently submitted new work in which they blurred the faces of the people presented in the ImageNet dataset and showed that the new obfuscated dataset can achieve similar model accuracy compare to the original ImageNet and the visual features learned from the modified dataset is equally transferable for other downstream vision tasks. Another direction of research focused on the label correctness in existing datasets (Roth et al., 2021). Northcutt et al. (2021b) showed that labels of popular datasets such as MNIST, CIFAR10 and ImageNet are not completely correct. The authors found that average labeling error on 10 popular benchmarks of image classification is 3.4%. In the work of Northcutt et al. (2021a) , Confidence Learning was proposed which is a theoretically grounded, mode-agnostic method to find label errors for classification tasks. This method calculates the approximation of the joint distribution of noisy observed label $\tilde{y}$ and unknown true label y^* to find errors in labels and to produce a clean dataset. They showed that cleaning data prior to training led to moderate improvement of the results.

There are other works which research the application of data-centric AI to improve the performance of machine learning systems. Motamedi et al. (2021) proposed a dataset generation pipeline for image classification to achieve better results with less data. Initially, data is manually cleaned and used to train an auxiliary model. The loss value of the model is then used to automatically select a group of candidate data points. These data points are later augmented to improve the class imbalance using a GAN network to make the final dataset. The idea of applying data cleaning and augmentation in the data pipeline exists in most leaderboards of the data-centric challenge (Ng et al., 2021). Terzi et al. (2021) proposed a data-centric approach to improve object detection performance of brain MRI images. First, the authors investigated the errors in the performance of the model and discovered that the object detection model has a bias problem. The authors attributed the bias

problem in the prediction to the labeling scheme in the data pipeline and proposed a new labeling method to rectify the error; this was done simply by adding a new class which was considered to be background before.

So far, it appears that data-centric methods are rarely used in semantic segmentation. Zlateski et al. (2018) discovered that the performance of CCNs depends on the amount of time spent on annotating the training data. However, this work investigated the hours spent to achieve finer annotations and did not investigate the effect of quality of annotations in terms of inconsistencies in the labels or presence of noise in the data. Roth et al. (2021) considered image augmentation to be a data-centric method since only data is modified to improve the performance while the model remains unchanged. The paper focused on the semantic segmentation of sensory image data and making it more robust towards noises. To sum up, there is very limited research on the application of data-centric AI in semantic segmentation. Given remarkable results of these methods in image classification and object detection, it is sensible to assume that data-centric methods would lead to improvement in semantic segmentation as well. There is however a noteworthy difference between the data in image classification and semantic segmentation, in semantic segmentations labels are dense and per-pixel. This suggests that semantic segmentation might require more sophisticated data-centric methods compared to image segmentation.

3.4 Data Augmentation for semantic segmentation

As mentioned in section 2.1.2, data augmentation is an approach to mitigate the overfitting problem in neural networks. In computer vision tasks such as semantic segmentation, it is a common practice to use data augmentation and is an essential part of state-of-the-art methods (Tang et al., 2020).

Like most machine learning tasks, simple transformation of the input image is the most common method used in semantic segmentation. In the original U-Net paper, elastic deformation proved to be the key to achieve good performance on the medical image dataset with few data samples. Boguszewski et al. (2021) made 9 data points from each image by randomly altering hue, saturation, brightness, sharpness and grayscale of images along with noise injection, flipping, mirroring and cropping operation. This was referred to as offline augmentation.

Some recent augmentation methods tend to be unintuitive yet effective. Mixup is a method proposed by Zhang et al. (2018) which randomly samples pairs of images from the training set, afterwards adds their weighted combination to the training set as a new datapoint. Cutout was introduced by Devries and Taylor (2017). Simply by masking out a square region of image, the performance of image classification models improved on datasets such as CIFAR-100. CutMix, on the other hand, mixes the crops from input images to create new data points (Yun et al., 2019). Similar to CutMix, Olsson et al. (2020) proposed ClassMix which uses the class mask to blend images. ClassMix can be used for semi-supervised learning as well as supervised learning. The difference is that in semi-supervised learning, pseudo-labels are used to blend images instead of actual labels. There are studies that developed task-dependent augmentation methods inspired by the mentioned methods. DepthMix is an excellent example of such study which is an alteration of ClassMix (Hoyer et al., 2021). By leveraging self-supervised depth estimation,

DepthMix blends the label of images while preserving the integrity of the structure of image. In order to use this approach, it should be possible to estimate the depth of the objects in the image.

Another common trend of augmentation methods in semantic segmentation is using generative models to generate training data points for training. Zhang et al. (2021) proposed using the power of generative adversarial networks. The authors showed that latent code made by the generative model could be decoded to predict the semantic segmentation of the image. This method results in an infinite annotated dataset generator. Tritrong et al. (2021) proposed Auto-Shot segmentation. The authors used a pretrained StyleGAN to generate the pixel-wise representation of the input image. This representation is used as input for the segmentation model. They showed that this representation is so effective that the segmentation network needs only one example for training. So far, all the methods rely on manually designed methods. Cubuk et al. (2018) proposed AutoAugment, a method to automatically find an effective data augmentation policy for any target dataset, to address this issue. The main idea is to use the power of recent Model-Free Deep Reinforcement Learning to learn best performing augmentation policies from designed augmentation space. The policy consists of several sub-policies and each of them apply 2 augmentation operations such as rotation and scaling to the input image. The policy is trained using Proximal Policy Optimization (PPO) algorithm (Schulman et al., 2017) with the validation loss as the reward. The authors showed that learned policies are transferable between datasets. It means that the policy learned for the ImageNet dataset as target is effective on other datasets such as Caltech-101 Fei-Fei et al. (2004). One of the winner of the data-centric AI challenge used policy learned on SVHN dataset to augment their target dataset Ng et al. (2021). Tang et al. (2020) pushed this idea further and proposed a method to optimize the augmentation policy along with the target network in parallel. It is important to consider that unlike image classification, in semantic segmentation, many augmentation methods change the corresponding label of the image. For example, by rotating the image, the image mask also needs to be rotated with respect to the image rotation. Random erasing of the image leads to loss of information for the corresponding mask.

3.4.1 Related Datasets

In what follows, currently available Remote Sensing datasets are presented, and their similarities and differences to the task at hand are discussed.

Due to recent improvements of the Remote Sensing (RS) technology, a lot of data has now been made available: from aerial photography to satellite images. Considering the success of deep learning within computer vision tasks, there is unsurprisingly an active area of research within application of deep learning in RS. There are several works that have provided suitable datasets for training and testing the Neural Networks. Moreover, some libraries are developed to make it easier to use geographical data in widely used deep learning platforms. For example TorchGeo, which integrates Geo-Spacial data in Pytorch and also provides useful tools such as pre-trained models for multi-spectral satellite images (Stewart et al., 2021).

Long et al. (2020) provided a thorough review of Remote Sensing (RS) benchmarks that are currently available. This includes scene segmentation, classification and object

detection datasets. Additionally, the authors have described some problems that these benchmarks are facing and discussed the challenges of making a suitable RS benchmark, including how to create a benchmark efficiently. The authors pointed out that pixel-wise annotation of RS images is a time-consuming and labor-intensive task due to the fact that all the processes involved rely heavily on manual operations. Moreover, it is mentioned that there is a need to develop new tools to make annotations for RS images. This is because common tools used for annotating semantic segmentation datasets, such as LabelImg (LabelImg, 2015) and LabelMe (Russell et al., 2007), are facing challenges when they are being used to annotate RS images. These challenges can be summarized as:

- RS images are large scale and cover a large area, unlike datasets such as natural images. Therefore visualizing the images for annotation is a challenge in current tools.
- Some RS datasets contain hyper-spectral images. It makes it challenging for the tools to provide an appropriate visualization of images for the annotators.

There are 3 annotation strategies for making such datasets.

Manual Annotation: The most common way of making semantic segmentation dataset is using manual work to annotate images. Even though this approach benefits from accurate annotation, it is laborious and not cost-effective. Specifically for complex tasks such as medical or geographical images where domain knowledge is needed for annotations. As a result manually annotated datasets are susceptible to be biased. Machine learning models can be used to speed up the process of manual annotations by providing the preliminary annotation (Andriluka et al., 2018).

Automatic Annotation: Leveraging machine learning scheme can be used to provide automatic annotation and reduce the cost of annotation. In this approach, first a model is trained using an initial set of data via supervised or weakly supervised approach. The interpretation of the model is used as annotation information. Iterative and incremental learning can be used to enhance the annotation. The main drawback of this approach is reliance on the performance of the model which depends on the initial set of images.

Interactive Annotation: This is a semi-automatic approach which is designed to improve the efficiency of dataset generation. In this approach human annotators and machine learning model work in an iteration. The model provides the initial annotation and human annotators provide intervention to the model. Therefore, instead of annotating the whole image, simpler operations, such as scribbles or point clicks, are needed to provide the intervention to the model. Moreover, active learning can be employed to expand the dataset size in an iterative way.

Table 3.1 summarizes the current benchmarks in semantic segmentation of RS images, which are stated in (Long et al., 2020), along with the last row which was published very recently in NeurIPS 2021. Among datasets that are mentioned in the table above, LandCover.ai, DeepGlobe and Agriculture-Vision are the most similar to the current task.

Dataset	#Class	#Image	Resolution	#Channel	Size	Year
ISPRS Vaihingen (Gerke et al., 2014)	6	33	0.09	5	2500x2500	2012
ISPRS Potsdam (Gerke et al., 2014)	6	38	0.05	5	6000x6000	2012
Massachusetts Buildings (Mnih, 2013)	2	151	1	RGB	1500x1500	2013
Massachusetts Roads (Mnih, 2013)	2	1171	1	RGB	1500x1500	2013
Zurich Summer (Volpi and Ferrari, 2015)	8	20	0.62	NIR,RGB	1000x1150	2015
SPARCS Validation (Hughes and Hayes, 2014)	7	80	30	11	1000x1000	2016
Biome (Foga et al., 2017)	4	96	30	11	9000x9000	2017
Inria (Maggiori et al., 2017)	2	360	0.3	RGB	5000x5000	2017
EvLab-SS (Zhang et al., 2017)	10	60	0.1 to 2	RGB	4500x4500	2017
RIT-18 (Kemker and Kanan, 2017)	18	3	0.05	6	9000x6000	2017
CITY-OSM (Kaiser et al., 2017)	3	1671	0.1	RGB	2500x2500	2017
Dstl-SIFD (Laboratory)	10	57	0.3	16	3350x3400	2017
IEEE GRSS Data Fusion Contest 2017	17	30	1	9	643x666;374x515	2017
DLRSD (Shao et al., 2018)	17	2100	0.3	RGB	256x256	2018
DeepGlobe Land Cover (Demir et al., 2018)	7	1146	0.5	RGB	2448x2448	2018
So2Sat LCZ42	17	400673	10	10	32x32	2019
SEN12MS (Schmitt et al., 2019)	33	180662	10	13	256x256	2019
ALCD Cloud Masks (Baetens et al., 2019)	8	38	10	RGB	1830x1830	2019
SkyScapes (Ruppert, 1988)	31	16	0.13	RGB	5616x3744	2019
DroneDeploy (Nicholas Pilkington)	7	55	0.1	RGB	12039x13854	2019
Slovenia LULC (Sentinelhub)	10	940	10	6	5000x5000	2019
LandCoverNet (Alemohammad and Booth, 2020)	33	1980	10	NIR,RGB	256x256	2019
GID (Tong et al., 2018)	15	150	0.8 to 10	4	6800x7200	2020
LandCover.ai (Boguszewski et al., 2021)	3	41	0.25,0.5	RGB	9000x9500;4200x4700	2020
Agriculture-Vision (Chiu et al., 2020)	3	41	0.25,0.5	RGB	9000x9500;4200x4700	2020
S2CMC (Francis et al.)	3	41	0.25,0.5	RGB	9000x9500;4200x4700	2020
LoveDA (Wang et al., 2021a)	7	5987	0.3	RGB	1024x1024	2021

Table 3.1: RS image semantic segmentation datasets.

Considering that they all have images of natural landscapes with high resolution. LandCover.ai is a semantic segmentation dataset of aerial images from rural areas across Poland with resolutions between 25 to 50 cm per pixel. This dataset contains *Building*, *Woodland*, *Water*, *Road* and *Background* classes. Agriculture-Vision is a large aerial image dataset for pattern analysis of agricultural lands which has 10 cm per pixel resolution. *Drydown*, *Nutrient deficiency*, *Weed cluster*, *Endrow*, *Double plant*, *Waterway*, *Storm damage*, *planter skip* and *Water* are existing classes of the dataset. DeepGlobe is a Satellite Image Understanding Challenge which consists of three challenges. Road extraction, building detection and land cover classification. Winner of each task had an oral presentation at the CPVR EarthVision workshop 2019 which was organized by DeepGlobe team. Even though DeepGlobe contains satellite images, it has high resolution of 50 cm per pixel. Dataset has 6 classes which are *Urban*, *Agriculture*, *Range land*, *Forest*, *Water*, *Barren* and *Unknown*. This dataset is used for transfer learning in this thesis. All of the aforementioned dataset contain RGB images except Agriculture-Vision which additionally contains Near-infrared channel. However, the images provided for the this thesis only contain grayscale channel. The work of Wang et al. (2021b) introduced a historical aerial image dataset with grayscale images. However, the authors formulated the land cover understanding to be a classification problem instead of semantic segmentation. Furthermore, none of these datasets cover the *gravel* class which is critical in analyzing the evolution of riverscapes.

Chapter 4

Methods

This chapter describes the methods used in this thesis. It starts off by describing the neural network architectures used in the experiments. Then continues to describe the model-centric and data-centric methods used to improve the performance of semantic segmentation. Finally, it describes the method which was used to provide the predictive uncertainty of the semantic segmentation in this work.

4.1 Deep Learning Architectures

In this section, the deep neural network architectures used in this thesis are presented.

4.1.1 ResNet50 Encoder

Residual neural network also known as ResNet is presented in the Background section 2.1.1. In this thesis, similar to many popular semantic segmentation works such as (Huynh et al., 2021; Wang et al., 2021a; Seferbekov et al., 2018b), ResNet50 is selected as encoder. As the name suggests, ResNet50 has 50 layers, 48 convolutional layers, 1 max pooling and one average pooling when it is used for classification. This architecture contains 4 residual layers, also known as *stages*, and each stage consists of 3 convolutional layers. Figure 4.1 illustrates the architecture of ResNet50, the left table shows the overall architecture and the right figure illustrates the first bottleneck layer of the first stage. Applying two 1×1 convolution filters in the bottleneck, reduces the number of parameters without notably degrading the performance. Another important property of ResNet is utilizing batch normalization. In all the experiments, ResNet50 model is pretrained on ImageNet dataset to speed up the training and help the generalization and robustness of the model.

4.1.2 DeepLab V3+

As described in section 3.1, the main components of DeepLabV3+ are Atrous Spatial Pyramid pooling (ASPP) and encoder-decoder architecture. Additionally, depthwise separable

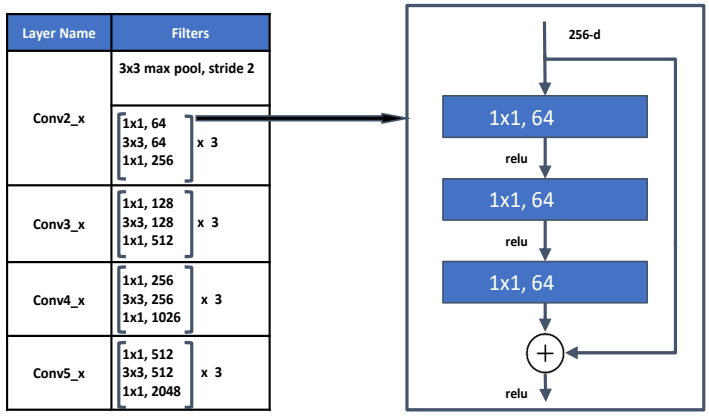


Figure 4.1: ResNet50 overall architecture.

convolution is applied to both ASPP module and decoder in order to achieve faster and stronger prediction. The encoder consists of a modified ResNet50 and ASPP layer. In the ASPP module, first 4 depthwise separable dilated convolution with dilation rate of 1, 6, 12 and 18 are applied to the input, these feature maps together with downsampled input are concatenated together and a 1×1 convolution layer with batch normalization is applied to make the output.

Figure 4.2 illustrates the architecture of DeepLabv3+ with ResNet50 that is used in this thesis. In the encoder, the output of final convolutional layer of ResNet50, the Conv5 in Figure 4.1, is fed into the ASPP layer while the output of first residual layer of ResNet and Conv2 is fed directly to the first layer of decoder. In the decoder, first a depthwise dilated 1×1 convolution layer is applied to the input which is then concatenated with the upsampled output of ASPP module. Afterwards 2 depthwise dilated 3×3 convolution layer is applied to the concatenated feature maps followed by final upsampling and 1×1 convolution layer to make the final prediction.

4.1.3 U-Net

U-Net is a very straight forward architecture. In this work two versions of U-Net are used. One is U-Net with VGG16 (Simonyan and Zisserman, 2014) as encoder as described in (Dalsgård, 2020). This architecture is also referred to as U-Net VGG16. The other architecture uses ResNet50 as the encoder and it is referred to as U-Net Resnet50 in this thesis. Beside residual layers, ResNet adopts batch normalization technique in its architecture, as a result, U-Net ResNet50 takes advantage of batch normalization.

In U-Net ResNet50 first input goes through all 4 stages of ResNet50 and then is upsampled in 4 stages of decoder while the output of each encoder stage is concatenated to the input of corresponding decoder as skip connection. Each decoder stage contains two

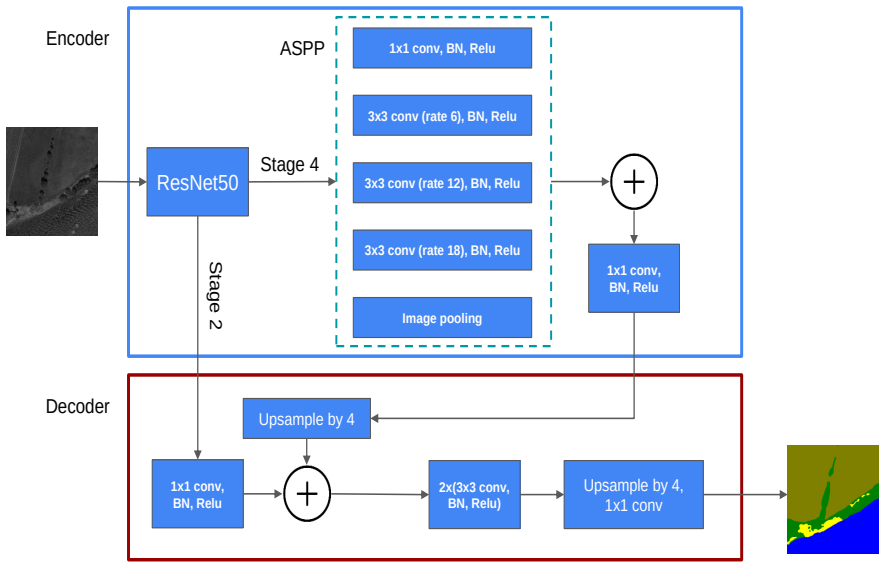


Figure 4.2: DeepLabV3+ architecture with ResNet50 encoder.

sequences of a 2d-convolutional layer, batch normalization and ReLU activation. Final output of the decoder is fed into a 1×1 convolutional layer followed by a sigmoid layer similar to (Alam et al., 2020). Figure 4.3 illustrates the architecture of U-Net ResNet50.

4.1.4 FPN

FPN consists of three components, bottom-up pathway, top-down pathway and lateral connection. In this work, similar to (Seferbekov et al., 2018b) ResNet50 is used as the bottom-up pathway of the architecture. Bottom-up component can be considered as the encoder, since it is used for extracting features. Figure 4.4 shows the architecture of FPN implemented in this thesis. At the last layer of the bottom-up component, a 1×1 convolution is applied to the output to make the O_1 which is the first input to the top-down component.

In the top-down component at the stage K, the previous input O_{K-1} is upsampled with the rate of 2 to make $O_{K-1}\prime$; a 1×1 convolution is applied to the corresponding bottom-up output B_K which has the same dimension as the $O_{K-1}\prime$; an element-wise addition is applied to B_K and $O_{K-1}\prime$ to make the O_K. A sequence of two 3×3 convolution layers are applied to the output of each top-down stage followed by upsampling. Afterwards the upsampled feature maps are concatenated together to make the stack of predictions. Ultimately spatial dropout is applied together with the upsampling and activation layer to make the final prediction map of the model. This helps to capture the multi-scale information about the image.

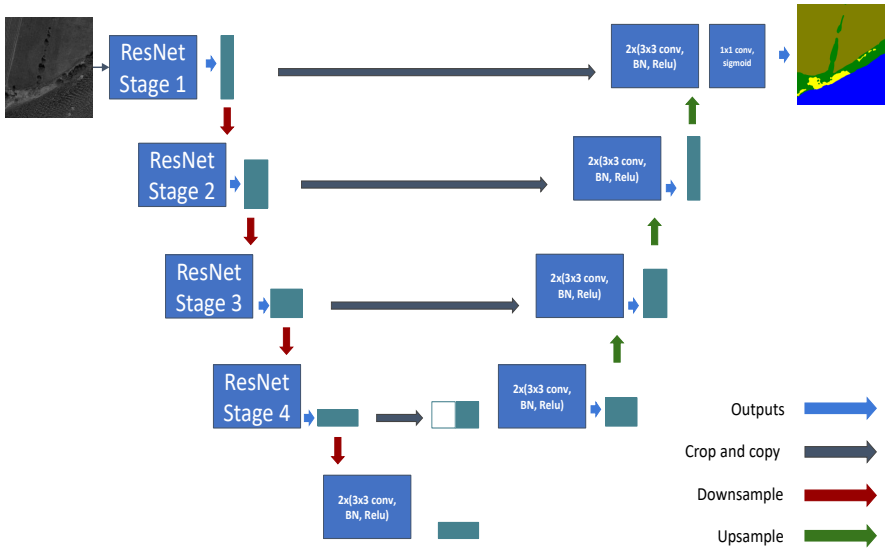


Figure 4.3: U-Net architecture with ResNet50 encoder.

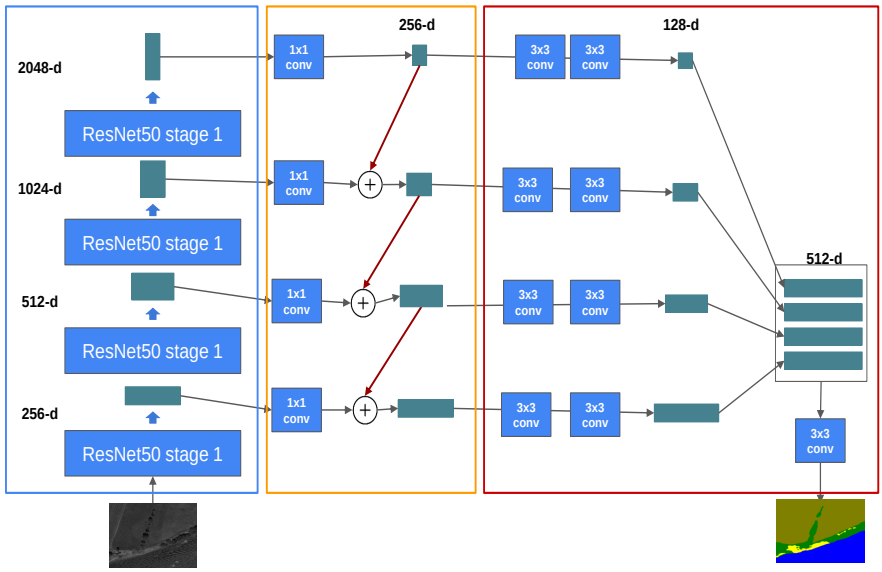


Figure 4.4: FPN architecture with ResNet50 encoder.

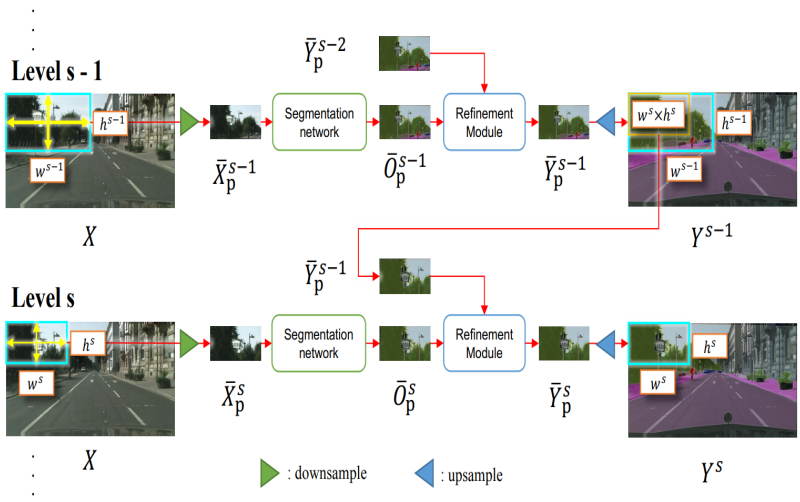


Figure 4.5: MagNet architecture with FPN-ResNet50 Backbone.

4.1.5 MagNet

MagNet module consists of two integral modules, refinement and segmentation modules. Refinement module can improve a segmentation output by employing another segmentation output.

MagNet is a multi-stage architecture, meaning that input is processed in multiple scales. These scales are set before training. Assuming there are 3 stages with scales of 612×612, 1224×1224 and 2448×2448 respectively. In the first stage, the coarsest scale image, which has the most covered area and least detail, is fed into the segmentation network to get the segmentation map O^1 with the size of $w^1 \times h^1$. In the first stage, the refinement module is not applied and O^1 is considered to be the output of this stage $Y^1 = O^1$. Afterwards the middle scale image, which covers less area but has better details, is fed into the segmentation module to achieve O^2. Subsequently, O^2 and those parts of Y^1 that overlap with O^2, fed into the refinement network to predict the Y^2. Same process is done in the last stage with the finest input to achieve O^3, refining O^3 and Y^2 results in Y^3 which is the final output. Figure 4.5 visualizes the described progressive mechanism.

The architecture of the refinement module used in the experiments is shown in Figure 4.6. This module has 3 convolutional layers and 2 residual blocks to process the $h \times w \times 2c$ size input and output $h \times w \times c$ size output. FPN with ResNet50 encoder is used as the segmentation module.

To train the network, first the backbone segmentation is trained on the images with different scales. Afterwards, the trained backbone model is used to train the refinement module.

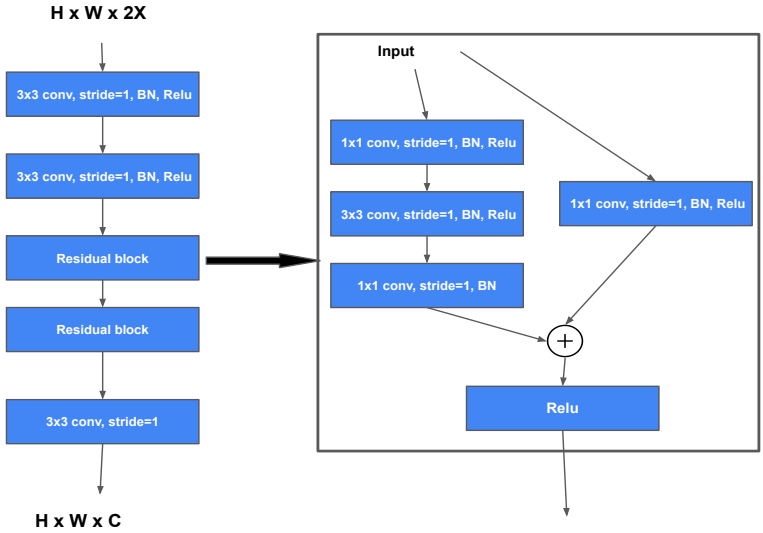


Figure 4.6: MagNet refinement module architecture.

4.1.6 Training details

In this section, the details of training the models are explained. For all the models, batch size is set to 16, except for MagNet which has a batch size of 8 for training the backbone FPN and 1 for training the refinement module. In order to tune the hyperparameters, the Hyperband algorithm (Li et al., 2018) was used to select the initial learning rate between (0.01, 0.001, 0.0001) and Dropout rate between (0.0, 0.1, 0.2). The objective of hyperband is the *value accuracy* with maximum 20 epochs and 10 hybrid iterations. All experiments used categorical cross entropy loss unless stated otherwise. For architectures such as DeepLabV3 and U-Net ResNet50 which do not contain any Dropout layers, only the initial learning rate was tuned. Moreover, the number of steps per epoch is manually tuned.

In MagNet, SGD is used for optimization with a weight decay of 0.9. For other experiments Adam (Kingma and Ba, 2015) is used for optimization with the ReduceLROnPlateau algorithm that reduces the learning rate by a factor of 0.5 if value loss does not decrease for more than 5 epochs. Finally $L2$ regularization is used for convolutional layers in all the models except MagNet. When Stochastic Weight Averaging (SWA) is applied, it is activated after the convergence with the constant learning rate of 0.0005. Early stopping is applied for all the models except MagNet. MagNet is trained for 484 epochs in normal experiments and 584 when SWA is applied. Other models stop training if value loss does not decrease for 20 epochs.

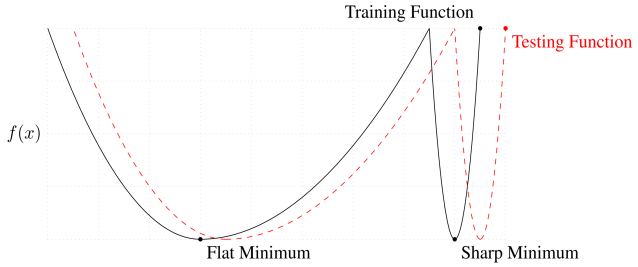


Figure 4.7: Comparison of generalization of sharp and broad minimas.

4.2 Stochastic Weight Averaging (SWA)

Averaging the network parameters in SGD optimization dates back to convex optimization and has been used in training deep neural networks. Louizos and Welling (2016) used Polyak-Ruppert averaging (Ruppert, 1988) to train their proposed variational Bayesian neural network. In the work of Nekrasov et al. (2018) this method is used to achieve a faster convergence in automatic search for neural network architecture. In Stochastic Gradient Descent optimization variants such as RMS Prop and Adam (Kingma and Ba, 2015) , the objective is to find the optima of the loss landscape. Keskar et al. (2016) claimed that batch gradient methods have the tendency to converge to the sharp minima. Compared to sharp minimas, broader minimas in the loss landscape are more likely to correspond to the solutions with better generalization. Figure 4.7 illustrates this effect by comparing the gap of training and test loss between sharp and wide minima of training loss in a simplified example. Izmailov et al. (2018) proposed a method, called Stochastic Weight Averaging (SWA), which will converge to the solution points that are wider optima compared to SGD variants and therefore will lead to better generalization. The main idea behind SWA is that by using high constant or cyclical learning rate and averaging the points which are traversed, it is possible to find a solution in the flatter area of the loss landscape.

To apply the SWA, the network is first trained to the convergence. Afterwards, the learning rate is set to a reasonably large value or assigned by a cyclical learning rate schedule (Garipov et al., 2018). At this stage, snapshots of network parameters are taken at a frequency which is set as a hyperparameter. After training, the average of saved parameters is assigned to be the parameters of the network. SWA (Izmailov et al., 2018) does not find the local optima, due to the fact that training loss for SWA is worse than SGD. It is shown to improve generalization with no overhead in multiple computer vision tasks such as CIFAR-10, CIFAR-100 and ImageNet. The only drawback of this approach is introducing new hyperparameters such as constant learning rate and the frequency of taking the snapshots.

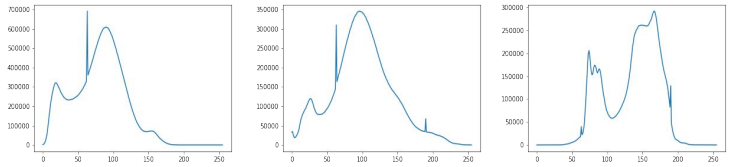


Figure 4.8: intensity histogram of three test sets. a) Gaula 1963, b) Nea 1962 and c) Gaula 1998

4.3 Online Data Augmentation (OA)

Image augmentation leads to improving the generalization of the model and mitigates the overfitting. In this thesis two online image augmentation methods are implemented as well as rotation augmentation which is an offline augmentation method. As the name suggests, online data augmentation means that the transformations are applied in real time. In another word, new data points are not being saved in the memory disk. To design the online augmentation methods, the difference between the distribution of training and test dataset is considered. For instance, comparison of histogram of historical images of 3 rivers is shown in Figure 4.8 that implies images have different brightness and contrast. This change in distribution is somehow expected since those images are taken from different landscapes, in different times using different technologies. All the augmentation methods are stochastic. It means each operation will be applied to the image with a given probability.

In previous work, it was proposed to adjust the average intensity of new images before feeding them into the network. This was done by decreasing the difference between the mean intensity of new images and training images. Since the difference needs to be set manually, it contradicts with having an out of the box model for large scale analysis. Moreover, average intensity depends on the coverage of the area of interest. For example, selected images of Gaula 1998 had average intensity of 133, however, the total set of images of this river has an intensity of 78. In this work, online-augmentation sought to solve the problem of variation in intensity of images by online-augmentation. In the following two augmentation methods are elaborated:

4.3.1 Online Image Augmentation version 1

In the following, augmentation methods applied to the image are briefly described. Reader is referred to (Gonzalez and Woods, 2008) for more detail about the transformations. In this version, images were *randomly flipped* and *transposed*. The *brightness* and *contrast* of images were randomly changed. In addition, *image compression*, *optical* and *grid distortion* along with *blurring filters* were applied to the image at random. The following elaborates the transformations with more details.

Robustness against spatial transformation is important in aerial photography. This is due to the properties of landscapes and the way aerial photography is taken. To elaborate, consider a river appearing in an image. Regardless of the direction and size of the river,

this entity is recognized as a river. To do so, transposition and flipping of the image were applied at random with probability of 10% for each operation.

Histograms Figure 4.8 manifests that robustness of model against light intensity and contrast is crucial. Consequently, random brightness and contrast augmentation were applied with probability of 20%. To cope with any noise that occurred in the images during the time the image was taken, scanned and published, Image Compression, Grid and Optical Distortion were applied to the image. All operations have a probability of 10%. To improve the robustness further, one of the Median blur filters, Random Gamma or Motion blur filter was applied with probability of 20%. Motion blur filter is applied with kernel size of 5 like Median filter. Random Gamma augmentation is Gamma (Power Law) transformation with γ is assigned at random.

4.3.2 Online Image Augmentation version 2

Second version of augmentation is less intensive compared to the first version. In this version, images were *randomly flipped* and *transposed*. The *brightness* and *contrast* of images were randomly changed. However, compared to previous version, brightness and contrast were changed less intensively. In addition, *blurring filters* were applied to the image at random.

The assumption for the moderation is to help the training process by making it a comparably easier task. Since historical images are grayscale, only texture can be used for interpretation of the image. The assumption is that these operations make it difficult for the model to learn the texture. For example Grid Distortion might lead to the illusion that a farm land is water due to the alteration of the texture of farmland. Limits of brightness and contrast changes are decreased so that images do not end up being too bright or dark.

4.4 Class Imbalance Mitigation

Learning algorithms are negatively affected by imbalance in the distribution of data classes (Krawczyk, 2016). Krawczyk defined three methods to address the class imbalance:

- *Data Level Methods* where data is modified or selected in a way that achieves better balance in the distribution of classes.
- *Algorithm Level Methods* where the model is designed in a way to counteract the imbalance of the dataset.
- *Hybrid Level Methods* that uses both Model and Data level methods together.

.

In this thesis, separate Data Level Method as well as Algorithm Level are utilized to balance the classes of the training data with more focus on more important classes, namely Gravel and Water. Rotation augmentation is proposed as a Data Level Method and Weighted Cross Entropy (WCE) is proposed to be used as a Model Level method. Both methods are elaborated in the following.

4.4.1 Rotation Augmentation (RA)

Relying on the domain knowledge, two assumptions can be considered about the aerial images:

- Gravel class is most likely found in the vicinity of Water class.
- If a pixel is gravel, the neighboring pixels are most likely gravel as well.

As a result of these assumptions, sampling more images with *gravel* as center can lead to mitigating the class imbalance by increasing the proportion of *gravel* and *water* in the dataset. In this work, rotation and sampling augmentation is proposed as a data level method to tackle the class imbalance. An overview of the method is mentioned in algorithm 1. To elucidate this method, consider a large image I, it is desired to sample small images from I with 10 degree rotation, $Angle$=10, with two constraints.

- Maximum 10% of the sampled image is Unknown class,
  $MaxUnknownPercentage$=10 .
- Minimum 70% of the area of the sampled image is new and not sampled before, $MaxOverlapPercentage$=30.

To do so, I rotated by $Angle$ and $SelectionLowerBound$ times images with *gravel* class in the center are sampled as random. Samples that satisfy two mentioned constraints get added to the dataset. This method will provide robustness against rotational invariants as well as improving the class imbalance. Rotation augmentation is a data-centric method. Figure 4.9 illustrates the rotation augmentation procedure.

4.4.2 Weighted Categorical Cross Entropy (WCE)

Weighted Cross Entropy is used as loss function in several studies to tackle the class imbalance. Özdemir and Sönmez (2020) used neural networks to diagnose COVID19 in COVIDX-Ray-5k dataset. The dataset is imbalanced and contains more Non-COVID samples compared to COVID samples, in order to mitigate this issue, Weighted Binary Cross Entropy is used as a loss function which led to state-of-the-art performance on the dataset.

Similarly, in this work Weighted Categorical Cross Entropy is used as a loss function as a means to tackle the class imbalance. The weighted cross entropy is defined as Equation 4.1 where similar to Categorical Cross Entropy at Equation 2.2, C is the number of classes, y_i is the ground truth of the i_{th} class and p_i is the softmax probability of i_{th} class. The only difference is w_i which is assigned weight for class i_{th}. z represents the output of the network.

$$WCE_{loss} = -\sum_{i}^{|C|}(w_i y_i log(p_i)) \tag{4.1}$$

$$p_i = \frac{exp(z_i)}{\sum_{j}^{|C|} exp(z_j)} \tag{4.2}$$

Algorithm 1: Rotation Algorithm for one large image.

Input : $Image_{M*N}$ (Large image with MxN dimension)
Input : P, Q (Dimension of sample images)
Input : $SelectionLowerBound$ (minimum number of sampling images)
Input : $SamplingClass$ (class of interest for sampling)
Input : $Angle$ (Rotation Angle)
Input : $MaxUnknownPercentage$
Input : $MaxOverlapPercentage$
Output: $ImageList$: list of rotated images

Initialize ImageList as empty list
Rotate $Image_{M*N}$ by $Angle^{\circ}$
$PotentialImageNumber \leftarrow (M * N)/(Q * N)$
$SelectionLowerBound \leftarrow max(SelectionLowerBound, PotentialImageNumber)$
for $i \leftarrow 0$ **to** $SelectionLowerBound$ **do**
    $SampleCandidate \leftarrow$ randomly sample an image with center of $SamplingClass$;
    if *less than* $MaxUnknownPercentage$ *of* $SampleCandidate$ *is UnknownClass* ***AND*** *less than* $MaxOverlapPercentage$ *of* $SampleCandidate$ *is already been sampled and added to* $ImageList$ **then**
        $ImageList$.append($SampleCandidate$)
    end
end

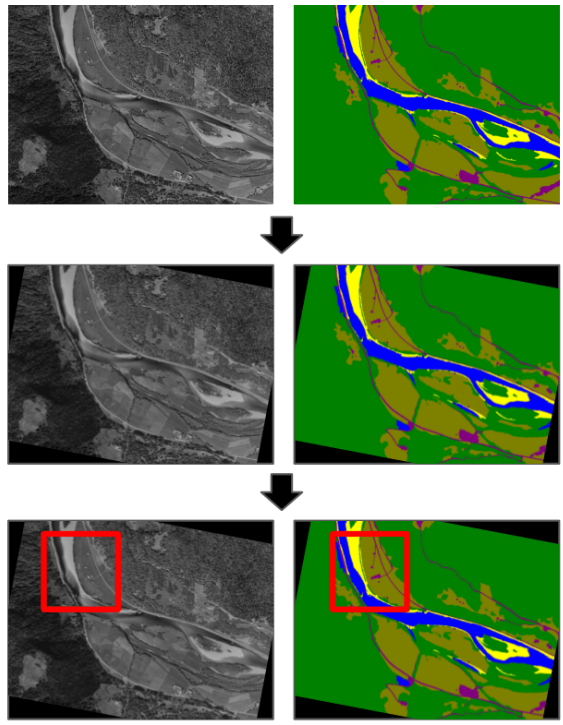


Figure 4.9: The process of sampling one image, marked in red, with the rotation augmentation method.

4.5 Data-Centric Method

The Figure 4.11 illustrates the data-centric as well as model-centric cycle. By following the data-centric cycle and analyzing the error of the baseline model, it was found that the labels of the initial dataset are noisy, inconsistent and in some places, incorrect.

To improve the quality of labels in the initial dataset, intuitively, the first idea was to correct the issues in the labels of existing dataset. However, the initial dataset consisted of scattered small images and did not cover a large connected spatial area. This means that images in the initial dataset, with 512×512 dimension, only contain a fraction of the original large images, with 8000×6000 dimension. Having a small segment of large area might make it difficult to understand the landscape due to lack of context information. For example, it is difficult to distinguish between a calm wide river and a vast farmland if the surrounding context information is not available. In addition, to make the initial dataset. Large images were partially annotated and divided into 512×512 patches. Then patches containing missing annotations, or including only one class, were removed from the dataset. As a result, it is not possible to combine the patches together to form a larger image with a wider view. Figure 4.10 illustrates the coverage of small patches in the initial dataset compared to large images. Not having broader context, hinders using methods that

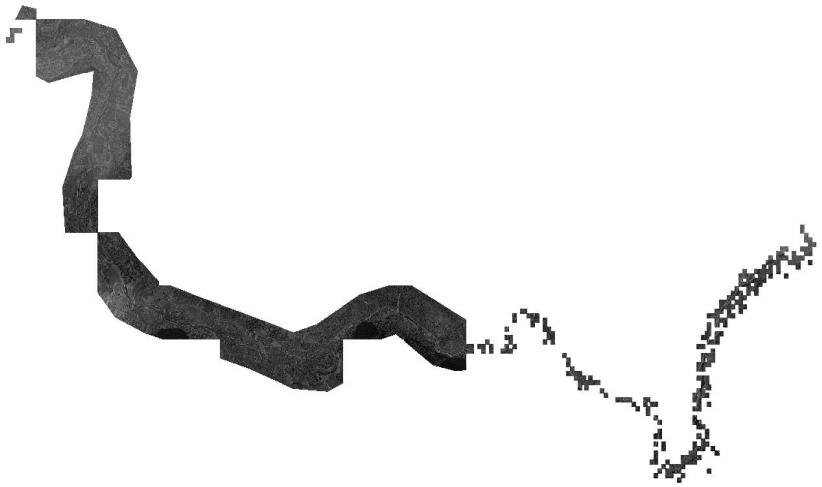


Figure 4.10: This figure shows the difference between the area covered by large images and initial dataset. In the left part, the large images of river Lærdal is illustrated, and the right half of image is the initial dataset placed in the correct geo-locations. This shows that initial dataset is scattered, and it is not possible to connect the small images to form a large, connected image..

incorporate the context information to predict the segmentation map such as (Huynh et al., 2021). Therefore, it was decided to annotate the existing large images from scratch.

The most common annotation tools for this purpose are GIS softwares with polygon editing tools such as QGIS (QGIS Development Team, 2009). These softwares, however, lack the precision required for the purpose of this work. To solve this issue, data pipeline was changed by adding additional steps to extracting the grayscale channel from the initial georeferenced data, processing the data and then attaching the geographic information to the processed image. This way, any image annotation too l could be used to label the images. Ultimately, Adobe Photoshop was selected to be the annotation tool since it enables having detailed annotation. Additionally it was possible to use an iPad to draw elaborated annotations as an extra layer on top of the image. For the annotation maps, 5 distinct colors were selected such that each color represents one class. Each image was loaded as a layer to Adobe Photoshop and on an extra layer on top of the image, each class was colored using the corresponding color of the class. Adobe Photoshop provided many utilities that facilitated the process of annotation in this work. Namely, Magic Wand was set up and used as a selection tool for most of the roads and Marching Ants algorithm (Viseras et al., 2016) was used to modify the edges of the objects.

Another important point in data labels is consistency of labels which becomes more important for uncertain areas or when there are multiple annotators labeling the data. To avoid any inconsistency in the labeling process, an annotation guideline was designed with the help of a domain-expert. Benefits of having a hard-coded and straightforward guideline are two-folded: In complex annotation tasks such as this work, where annotation uncertainty is high, guideline facilitates the labeling decisions for ambiguas cases. Guidelines

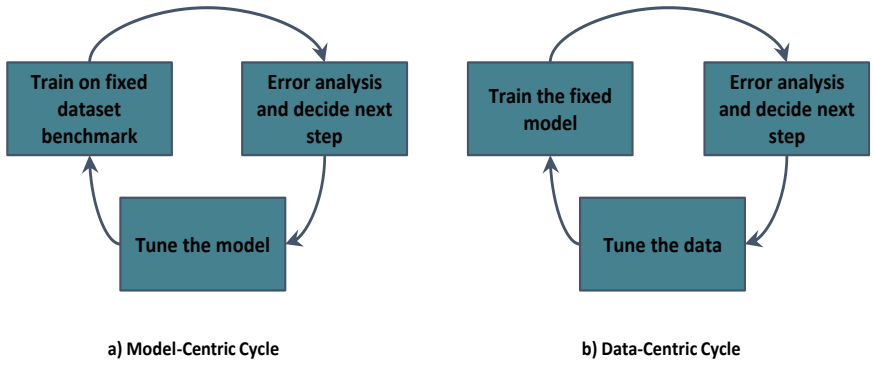


Figure 4.11: Data-centric and Model-centric cycles.

can pave the way for having collaborations on making or expanding datasets specifically when the task requires domain-expert. The guideline that was developed and followed in this work is presented below:

1. All the images should be annotated and areas with no image should be labeled as "Unknown".
2. Borders of classes should be as fine as possible. It is shown that deep learning based semantic segmentation models make their predictions mostly based on the shape and border of the objects (Wickstrøm et al., 2020).
3. Only what is visible is considered to be the true state of the map. For example if a road disappeared in the forest. It is not considered a road.
4. Dark shadows are considered to belong to the class that makes the shadow.
5. Main uncertainty challenges and how to solve them:
   (a) Confusion between human construction and gravel class: Due to the fact that images are historical, some roads, mines or even building constructions are very similar to gravel. In order to fix that, the current map should be checked and if there is a road or building in that place, it should be labeled as a human construction class.
   (b) Some areas that are not forest and not clearly farmland. These areas should be classified as vegetation.
6. In case of any uncertainty in the class, recent land cover maps of the uncertain area needs to be inspected to achieve more information about the area.
7. If recent maps did not help, cases should be reported to the domain-expert to resolve the uncertainty.

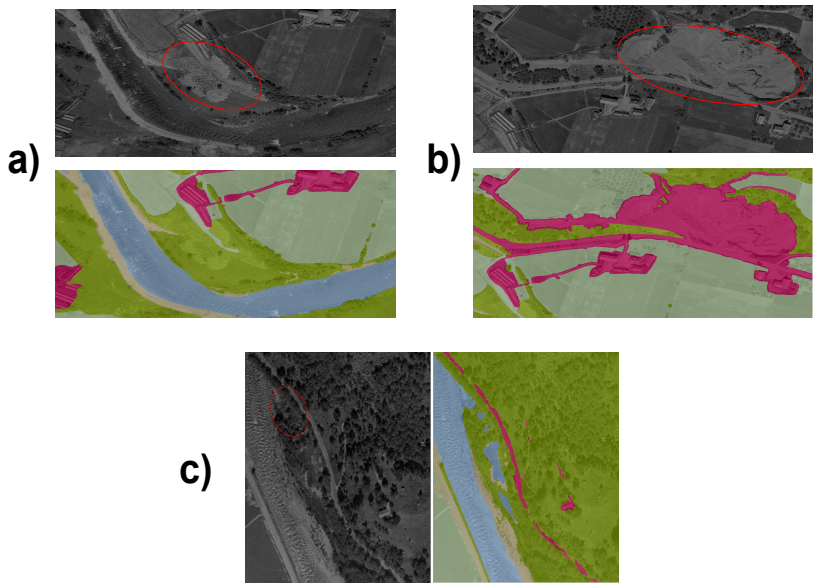


Figure 4.12: Example of the vague areas for annotation marked in red which were sent to the domain expert to help determining the underlying classes. a) and b) confusion between *gravel* and *human construction*. c) confusion between *vegetation* and *water*.

Figure 4.12 shows some of the cases that were referred to domain-expert due to being confusing, and asked for guidance.

4.6 Predictive Uncertainty Estimation of Semantic Segmentation

A brief introduction to uncertainty estimation in semantic segmentation is given in section 2.2. In this section we describe how predictive uncertainty of semantic segmentation models are estimated and used in this work. Similar to many deep learning tasks, recently the application of uncertainty estimation has increased in semantic segmentation. From identifying Out-of-distribution (OOD) and active learning (Papp et al., 2019) to improving the interpretability of colorectal polyps segmentation in medical images . In (Wickstrøm et al., 2020) uncertainty is estimated as running the model T times with Dropout layers activated to achieve T softmax output distribution. The standard deviation of these T samples are considered to be the estimated uncertainty.

However, in a study done by Dechesne et al. (2021) predictive uncertainty of semantic segmentation of satellite images is computed as predictive entropy of Monte Carlo Dropout as it is shown in the background section. In (Czolbe et al., 2021), the authors estimated

the uncertainty of semantic segmentation using Softmax entropy, Deep Ensemble, MC-Dropout and additionally Probabilistic U-Net (Kohl et al., 2018). It is shown that even the simplest estimation provides reliable indication of areas in the image that are ambiguous or have wrong predictions. This can specifically be very helpful for practitioners who use the prediction of semantic segmentation models.

In this work we estimate the uncertainty of the model to provide an insight into the performance of the model. Moreover, results of (Dechesne et al., 2021; Czolbe et al., 2021; Wickstrøm et al., 2020) showed that accurate predictions tend to have lower uncertainty whereas wrongly predicted areas have higher uncertainty one can say that models agree more on right answer but disagree more on the wrong one. Inspired by this, we compare the estimated predictive uncertainty of a fixed model trained on different versions of datasets. By fixing the model and only changing the training dataset, we see the difference in the uncertainty only related to the training data. Therefore, one can consider this comparison as comparison of validity of the different versions of the dataset.

Similar to (Dechesne et al., 2021), predictive uncertainty of the segmentation model is calculated using predictive entropy of Monte Carlo Dropout. Buy using the Equation 2.6, predictive entropy of T Monte Carlo samples can be calculated as Equation 4.3

$$\mathbb{H}[p(x)] = -\sum_{c}(\frac{1}{T}\sum_{t}p(y=c|x,\theta_t))log(\frac{1}{T}\sum_{t}p(y=c|x,\theta_t)) \tag{4.3}$$

Where c takes all the possible classes, p is the softmax probability of output of the network with activated dropout layers and θ_t refers to model parameters of the t^{th} sample.

Chapter 5

Experiment details

This section describes details related the datasets, models, features, data generation and experiments of the thesis. First, the chapter provides an overview of the datasets, models, features and transformations used, then the details of each experiment conducted is given.

5.1 Dataset details

In this work, the same training and validation set which was created in (Dalsgård, 2020) is considered as the *initial dataset*. The initial dataset is also referred to as the dataset V0. The dataset V0 contains 20328 images with the size of 512×512 pixels. In this work another dataset, dataset V1, is created. To make the dataset V1, large images with the size of 8000×6000 which were used to make the dataset V0 are retrieved from Kartvarket (Kartvarket, 2021). The details of making the dataset V1 are described in detail in the subsection 5.2.3. Table 5.1 shows the number of images with the size of 512×512 pixels in each dataset.

Additionally, to make it feasible to compare the performance of different methods, the test set used in this work is the same test set used in (Dalsgård, 2020). This test set contains a section from each of the river Gaula taken in 1998, 1963 and Nea taken in 1962. in the appendix the test sets are illustrated. In these three test sets, Gaula 1998 is the most

Dataset	**Rotation Augmentation (RA)**	**#Images** (512×512)	**Containing rivers**
Dataset v0		20328	Gaula 1963, Lærdal 1976, Surna 1963
Dataset v0	✓	31718	Gaula 1963, Lærdal 1976, Surna 1963
Dataset v1		13363	Gaula 1963, Lærdal 1976, Surna 1963
Dataset v1	✓	24300	Gaula 1963, Lærdal 1976, Surna 1963

Table 5.1: The summery of training and validation dataset used in this thesis. Dataset V0 is described in (Dalsgård, 2020), second row is explained in the first experiment (subsection 5.2.2) and the dataset V1 is described in E2 (subsection 5.2.3).

distinct set. This is due to changes in the technology used to take the images from 1962 to 1998. The Figure 4.8 shows the histogram of one of the images of each test set. The figure shows that Gaula 1998 has different contrast and intensity compared to the other two test sets. Moreover, by inspecting the images, it is visible that the older datasets are blurrier, and there are more distorted areas in them. All the images used in this thesis are grayscale and with a resolution of 20 cm per pixel.

5.2 Experiments

There where four experiments conducted in this thesis. Later in the thesis these will be referred to as **E1**, **E2** and **E3**.

5.2.1 Runtime Environment

All experiments were run on one node the Yoda cluster of Computer Science department of Norwegian University of Science and Technology. The information of this node is described below:

GPU: NVIDIA Tesla V100

CPU: Intel(R) Xeon-Gold 6240

Number of Cores: 18 cores @ 2.6 Ghz

RAM: 32 GiB

Models were implemented using Tensorflow (Abadi et al., 2015) and Pytorch (Paszke et al., 2019) packages. In addition, Albumentations library (Buslaev et al., 2020) was used for Online Augmentation, and SegmentModel (Yakubovskiy, 2019) liberiry was used in some of the implementations. In order to train the MagNet (Huynh et al., 2021), the script provided by the paper's github page was used.

5.2.2 Experiment 1

The objective of the first experiment **E1** was twofold. First goal was to provide a solid baseline for the data-centric method, and another objective was to determine the best way of feeding grayscale input to an encoder pretrained on RGB data. Firstly, the previous work done in (Dalsgård, 2020) was reproduced 5 times with 5 different random initializations to achieve the baseline performance.

As it is described in section 5.1, historical aerial images are grayscale. Additionally, the models that are used in this thesis have an encoder pretrained on ImageNet, which is RGB, to take advantage of the power of transfer learning. Grayscale images have 1 intensity channel whereas RGB has three. This leaves a 2-channel gap between the two data, and intuitively there are three way to fill this gap; assuming the input image is with the shape of $H \times W \times 1$:

1. **Convolution Layer:** Adding a 1×1 convolutional layer with 3 filters to the first layer of the network so that the transition between 1-channel input into 3-channel is learned by the network.

2. **Extra features:** Use grayscale images in the first channel and then add 2 extra channels with extra information that might help the learning process, i.e. extracted features, which can be helpful in the training process. Inspired by Ratajczak et al. (2019), the boundary of segments resulting from classical segmentation algorithms is assigned to the second channel and the feature extracted by arbitrary filters is assigned to the last channel.

3. **Copying:** Making three copy of the grayscale channel which lead to $H \times W \times 3$ shape.

To test which of the three proposed methods are more appropriate for this work, performance of the initial model which was trained on the initial dataset using these three methods are compared. The initial model, the U-Net VGG 16 proposed in the previous work (Dalsgård, 2020) as well as the datasetV0. Hyperparameters are selected exactly as described in the previous work. The implementation of the *convolution layer* method is straightforward since only one 1×1 layer with 3 kernels is added to the beginning of the encoder. The *copying* method is the one that was done in the original work. For the *extra features* method however, there was a need to choose a segmentation algorithm as well as a filter operation. To choose the segmentation algorithm, unsupervised algorithms with little tunable parameters, such as SLIC (Achanta et al., 2012), Quick shift (Vedaldi and Soatto, 2008) and Mean Shift (Comaniciu and Meer, 2002) were considered. After applying the algorithms on a sampled training image, algorithm parameters were manually tuned to maximize the area of segments that are semantically informative and minimize the number of segments with more than one label in the ground truth. After a visual inspection of the segments, Quick shift, with 0.9 ratio of color to image space proximity, was selected because of the quality of segments along with relatively fast run time. A Laplacian of gaussian filter with $sigma = 5$ was selected for the third channel. Going forward, the best performing method should be considered as the method of feeding the input image to the model.

In terms of creating the baseline, the work (Dalsgård, 2020) was first reproduced five times. This was done with the same hyperparameters as original work and different random initializations to determine the baseline performance. The performance of the reproduced baseline was further used for error analysis to discover which data-centric methods are more suitable for this application.

In order to assess the efficiency of the data-centric approaches, a set of common model-centric practices in deep learning research was tested additionally. The initial dataset suffers from class imbalance. This means that the distribution of the areas covered with different classes is not uniform. On the other hand, it can be fair to say, *gravel* and *water* classes have more importance to the purpose of this work. To mitigate this problem, the rotation augmentation method as described in subsection 4.4.1 was tested. Figure 5.1 shows the class distribution of the initial dataset before and after rotation augmentation. Afterwards, more advanced model architectures, namely FPN, DeepLabV3+ and U-Net

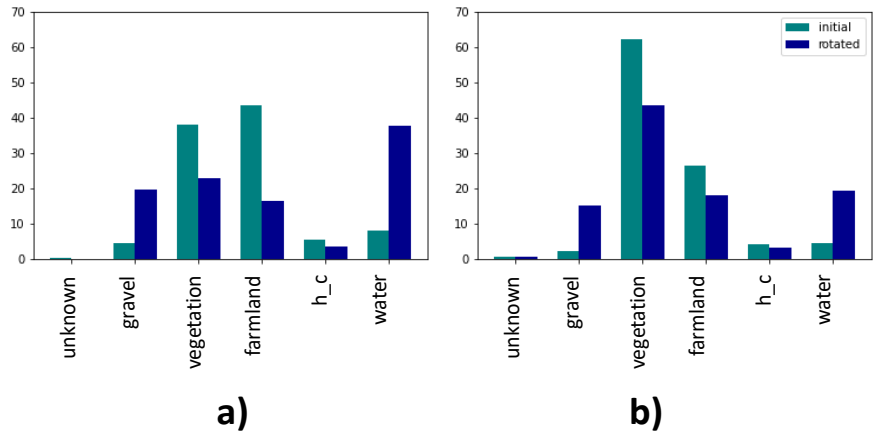


Figure 5.1: Distribution of classes in dataset v1 and v0, the light blue illustrates the datasets without rotation augmentation and dark blue show the distribution of the classes after rotation augmentation. a) new dataset (dataset v1), b) initial dataset (dataset v0)

with ResNet50, were used to train the initial dataset. The architectures and the training details are described in section 4.1.

5.2.3 Experiment 2

The objective of the second experiment, **E2**, was to investigate if the performance of the baseline semantic segmentation model could be improved by employing the data-centric method. After analyzing the errors of the baseline model, improving the label quality, aka dataset validity, was selected as the first data-centric method to tune the data. In data-centric AI, models are considered to be fixed in the training cycle Figure 4.11 and only the training data is modified. Following this paradigm, the test set of the initial dataset was not altered so that improvement, or lack thereof, can be detected. In **E2**, similar to previous work, images of sections of rivers Gaula, Laerdal and Surna were selected to create the training set. Selected training images were annotated using the method described in section 4.5. For annotation, the 6 classes of *water*, *gravel*, *vegetation*, *farmland*, *human construction* and *unknown* were considered, and *unknown* was only assigned to parts of a large image that did not contain any landscape and was completely back. It is a well-known fact that annotating the semantic segmentation data is a time-consuming and tedious task, and it becomes specifically more cumbersome if the quality and detail of the annotation is of great concern. Ultimately 87 large images with resolution of 8000×6000 pixels were manually annotated and a total amount of 270 hours were spent on this task which made it to be the most time-consuming part of this research. Table Table 5.2 shows the number of images belonging to each river in the initial dataset and in the dataset made in this experiment. For simplicity, the initial dataset is referred to as dataset V0 and the dataset made in this experiment is referred to as dataset V1.

River	Dataset	
	V0 partially annotated	V1 fully annotated
Gaula 1963	66	57
Lærdal 1978	38	24
Surna 1963	75	6
Sum	179	87

Table 5.2: Number of large images of 8000×6000 pixels in initial dataset (V0) and dataset developed in this work (V1).

Firstly, in order to see the performance of the dataset V1 on the baseline model, similar to the previous work, dataset-v1 was divided into 512×512 patches, and patches including only the *unknown* class were ignored. The results were 13363 small images and 80% of them were assigned to training and the rest were used for validation. The baseline model was trained on the patches without any data augmentation, and hyperparameters were selected as described in subsection 4.1.6. Considering the fact that a subset of large images used to make the initial dataset was utilized to make dataset V1, this dataset suffers from class imbalance as well. To mitigate this effect, similar to **E1**, rotation augmentation was applied by rotating the large image and sampling more *gravel* and *water* classes as described in the method section subsection 4.4.1. This increased the number of images in the training dataset to 24300. The distribution of classes before and after the rotation is shown in the figure Figure 5.1 which illustrates that rotation augmentation diminished the class imbalance. In order to account for randomness, the baseline model was trained on the expanded patched dataset with the same hyperparameters as above with the only difference being the random initialization. This way, the performance of the baseline model trained on dataset V0 and dataset V1 could be compared to determine if the selected data-centric method improved the performance of the segmentation or not.

5.2.4 Experiment 3

The final experiment attempts to confirm if model-centric methods can lead to further improvement of the new high-quality dataset. As stated in the method section, after reviewing the related work in semantic segmentation of RS in section 3.2, *new architectures*, *online data augmentation*, *weighted cross entropy* and *stochastic weight averaging* were selected as the potential model-centric methods that can lead to improvement of the semantic segmentation. Note that one can consider online augmentation as a data-centric method, however, since it requires changing the training pipeline by modifying the input mini-batches, in this thesis it is placed as a model-centric method. In **E3** an model comparison study of these 4 model-centric components were conducted to determine which of the proposed components could lead to improvement of the performance of semantic segmentation. Each component is explained further below:

In terms of *new architectures*, similar to **E1**, the architectures, DeepLabv3+, FPN and U-Net ResNet50 were selected. To train these architectures, 512×512 patches of dataset V1 with rotation augmentation were selected as the dataset, and the models trained as de-

scribed in section 4.1. Besides, the entire large images in dataset V1 were annotated, this enables the application of architectures that utilize the global information of high resolution images such as GLNet and MagNet. As MagNet is state-of-the-art on the DeepGlobe dataset, it was selected as the 4_{th} architecture for this experiment. The architecture of MagNet used in this work is described in subsection 4.1.5, and assuming that DeepGlobe dataset has similarity to the dataset of this thesis, the FPN-ResNet backbone segmentation model of MagNet was pretrained on DeepGlobe dataset. Afterwards, the last layer was changed to have a 6-dimensional output instead of 7. This is because DeepGlobe has 7 classes and this work contains 6 classes. For training the MagNet, large images were divided into 2448×2448 pixel images. Additionally, the same rotation augmentation method as **E1** and **E2** was done on the dataset V1, however, this time instead of sampling 512×512 pixel images, 2448×2448 pixel images were sampled. This led to a dataset of 2022 images with 2448×2448 pixels. To train the MagNet, same as the original paper, scales were set to be 612×612, 1224×1224 and 2448×2448. The FPN backbone model, that was pretrained on DeepGlobe, was fine tuned on the dataset V1. Afterwards, two refinement modules were trained to improve the segmentation with scales of 612×612 and 1224×1224 respectively. The training details are described in subsection 4.1.6. It is worth mentioning that freezing the parameters of the ResNet50 encoder during the training was tested. However, it was observed that it hindered the training process since the model struggled to decrease the training loss.

In terms of data augmentation, two *online data augmentation* methods were designed using the domain-knowledge of the task. First data augmentation applied more transformation on the input image and is referred to as online augmentation-v1, and the second method was less intensive in order to keep the training process easier. Both methods are explained in the method section section 4.3.

Regarding the *WCE*, As illustrated in the figure Figure 5.1, even after applying the rotation augmentation, the distribution of classes is not uniform. In order to overcome the imbalance further, and to encode the importance of *water* and *gravel* classes, WCE was used as the loss function and wegihts of each class was set as following:
unknown : 01.72%, *water* : 22.41%, *gravel* : 22.41%, *vegetation* : 17.24%, *farmland* : 17.24%, *human construction* : 18.97%.

It is shown that *SWA* leads to improvement of deep learning models including semantic segmentation models without adding much computational overhead. SWA was applied in E3 in order to confirm it could improve the performance of this work as well. The details of applying SWA is described in the method section section 4.2

Chapter 6

Results

This chapter presents the results of the experiments.

6.1 Experiment 1

The MIoU of all the models which were trained in **E1** is provided at the Table 6.1. The MIoU is calculated for each of the three test sets.

Model architecture	Encoder	id	RA	Feeding method	Test sets (MIoU)			
					Gaula 96	Nea 62	Gaula 98	Average
U-Net	VGG16	1		Convolution Layer	69.67	65.56	55.34	63.53
		2		Extra features	66.46	66.37	54.64	62.49
		3		Copying	69.12 ± 00.87	69.80± 04.62	54.18 ± 01.24	**64.60**
		4	✓	Copying	63.18	59.46	47.88	56.84
	ResNet50	5		Copying	70.1	67.94	**55.73**	64.59
FPN	ResNet50	6		Copying	**73.25**	**70.99**	40.06	61.43
DeepLabV3+	ResNet50	7		Copying	68.26	67.46	54.32	63.35

Table 6.1: MIoU of the models trained in E1 on all three test sets. RA is rotation augmentation.

The confusion matrix of one of the five baseline model which was reproduced in **E1** is provided in Table 6.2. The baseline model is reproduced model of previous work (Dalsgård, 2020). The confusion matrix is row normalized which means each row will some up to 1.

6.2 Experiment 2

The MIoU of models trained in **E2** is presented in Table 6.3. Baseline model trained on dataset V0 and dataset V1. If more than one model is trained, the result is presented as

Gaula 1963					
	W	**G**	**V**	**F**	**H**
W	**0.87**	0.01	0.04	0.08	0.00
G	0.08	**0.69**	0.14	0.06	0.03
V	0.02	0.01	**0.90**	0.04	0.04
F	0.01	0.02	0.08	**0.88**	0.01
H	0.03	0.03	0.11	0.13	**0.70**

Nea 1962					
	W	**G**	**V**	**F**	**H**
W	**0.91**	0.00	0.04	0.05	0.01
G	0.18	**0.46**	0.17	0.15	0.04
V	0.02	0.00	**0.86**	0.09	0.03
F	0.01	0.00	0.02	**0.96**	0.01
H	0.00	0.00	0.08	0.27	**0.64**

Gaula 1998					
	W	**G**	**V**	**F**	**H**
W	**0.90**	0.04	0.03	0.04	0.00
G	0.08	**0.62**	0.07	0.18	0.04
V	0.07	0.03	**0.63**	0.24	0.03
F	0.00	0.19	0.02	**0.76**	0.03
H	0.00	0.02	0.07	0.14	**0.77**

Table 6.2: Confusion matrix of reproduced baseline model trained in E1.

$mean \pm std$ where $mean$ is the average of all the results and std is the standard deviation of the results.

Model architecture	Encoder	id	Dataset	RA	Test sets (MIoU)			
					Gaula 96	Nea 62	Gaula 98	Average
U-Net	VGG16	1	V0		69.12 ± 00.87	69.80± 04.62	54.18 ± 01.24	64.60
		2	V1		78.16	68.4	61.82	69.46
		3	V1	✓	**79.23± 00.49**	**73.11± 01.02**	**62.62± 00.86**	**71.65**

Table 6.3: MIoU of the models trained in E2 on all three test sets. RA stands for rotation augmentation.

6.3 Experiment 3

The result of ablation study conduced during **E3** is presented in Table 6.4.
The row normalized confusion matrix of best performing model in ablation study, id=15 is shown in Table 6.5. Furthermore, Table 6.6 shows the row normalized confusion matrix of best performing MagNet model on the test sets.

Model architecture	Encoder	id	SWA	RA	WCE	OA		Test sets (MIoU)			
						V1	V2	Gaula 96	Nea 62	Gaula 98	Average
U-Net	VGG16	1						78.16	68.4	61.82	69.46
		2	✓					79.29	71.20	61.34	70.61
		3		✓				79.23	73.11	62.62	71.65
		4	✓	✓				79.13	73.43	64.75	72.43
		5		✓	✓			80.42	72.63	57.88	70.31
		6	✓	✓	✓			**80.55**	**74.27**	60.8	71.88
		7	✓	✓		✓		74.76	65.23	69.95	69.98
		8	✓	✓			✓	79.26	72.78	64.76	72.27
		9	✓	✓	✓		✓	79.56	72.96	64.76	**72.46**
	ResNet50	10		✓				79.84	72.34	65.9	72.69
		11	✓	✓				78.85	72.85	67.04	72.91
		12	✓	✓	✓			79.34	74.04	67.32	73.57
		13	✓	✓		✓		77.44	69.76	64.33	70.51
		14	✓	✓			✓	79.11	74.59	66.77	73.49
		15	✓	✓	✓		✓	77.59	71.25	**73.55**	**74.13**
FPN	ResNet50	16		✓				79.15	70.51	66.51	72.06
		17	✓	✓				79.63	73.01	64.57	72.40
		18	✓	✓	✓			79.57	71.8	67.59	72.99
		19	✓	✓		✓		76.77	67.84	67.55	70.72
		20	✓	✓			✓	76.43	70.21	61.11	69.25
		21	✓	✓	✓		✓	79.30	72.10	67.61	**73.00**
DeepLabV3+	ResNet50	22		✓				79.59	71.8	63.3	71.56
		23	✓	✓				79.45	71.98	63.35	**71.59**
		24	✓	✓	✓			77.54	68.9	62.54	69.66
		25	✓	✓		✓		69.00	64.82	68.11	67.31
		26	✓	✓			✓	77.87	66.78	59.48	68.04
		27	✓	✓	✓		✓	74.91	67.55	58.09	66.85
MagNet	FPN-ResNet50	28		✓				76.60	64.25	47.64	62.83
		29	✓	✓				77.62	64.94	49.44	64.00
		30	✓	✓	✓			79.08	68.61	56.16	67.95
		31	✓	✓	✓	✓		76.54	68.98	65.83	70.45
		32	✓	✓	✓		✓	79.36	72.01	63.14	**71.50**

Table 6.4: MIoU of the models trained in E3 on all three test sets. In the table, RA stands for rotation augmentation, OA is online augmentation, WCE stands for weighted cross entropy and SWA is stochastic weight averaging. Each of these configurations are referred to by the corresponding id value. The overall best performing configuration is colored with yellow.

Gaula 1963					
	W	G	V	F	H
W	**0.96**	0.01	0.02	0.00	0.00
G	0.03	**0.90**	0.06	0.01	0.00
V	0.00	0.03	**0.94**	0.03	0.00
F	0.00	0.00	0.13	**0.86**	0.00
H	0.01	0.01	0.22	0.13	**0.65**

Nea 1962					
	W	G	V	F	H
W	**0.93**	0.01	0.04	0.02	0.00
G	0.09	**0.72**	0.07	0.12	0.02
V	0.01	0.01	**0.89**	0.08	0.00
F	0.00	0.00	0.05	**0.94**	0.01
H	0.00	0.01	0.09	0.20	**0.69**

Gaula 1998					
	W	G	V	F	H
W	**0.96**	0.01	0.02	0.02	0.00
G	0.04	**0.33**	0.43	0.19	0.02
V	0.01	0.00	**0.97**	0.02	0.01
F	0.00	0.01	0.03	**0.96**	0.00
H	0.00	0.00	0.15	0.28	**0.57**

Gaula 1963					
	W	G	V	F	H
W	**0.94**	0.03	0.03	0.00	0.00
G	0.03	**0.91**	0.06	0.00	0.00
V	0.01	0.03	**0.94**	0.02	0.00
F	0.01	0.00	0.13	**0.85**	0.01
H	0.01	0.04	0.24	0.09	**0.62**

Nea 1962					
	W	G	V	F	H
W	**0.93**	0.01	0.04	0.02	0.00
G	0.07	**0.71**	0.08	0.11	0.02
V	0.01	0.01	**0.92**	0.06	0.01
F	0.00	0.00	0.07	**0.91**	0.01
H	0.00	0.01	0.12	0.15	**0.72**

Gaula 1998					
	W	G	V	F	H
W	**0.96**	0.03	0.01	0.01	0.00
G	0.04	**0.59**	0.25	0.06	0.05
V	0.02	0.00	**0.93**	0.03	0.02
F	0.00	0.03	0.03	**0.92**	0.02
H	0.00	0.02	0.06	0.15	**0.79**

Table 6.5: Confusion matrix of the best performing model on average all test sets in E3, which has the id=15 in Table 6.4 (bottom). Along with the same configuration only without weighted cross entropy which has the id=14 (top).

Gaula 1963					
	W	G	V	F	H
W	**0.94**	0.02	0.04	0.00	0.00
G	0.04	**0.88**	0.07	0.01	0.00
V	0.00	0.03	**0.94**	0.01	0.00
F	0.01	0.00	0.10	**0.88**	0.01
H	0.00	0.01	0.27	0.06	**0.66**

Nea 1962					
	W	G	V	F	H
W	**0.90**	0.03	0.07	0.00	0.00
G	0.06	**0.81**	0.11	0.01	0.01
V	0.01	0.01	**0.95**	0.03	0.01
F	0.00	0.00	0.11	**0.86**	0.02
H	0.00	0.01	0.13	0.11	**0.75**

Gaula 1998					
	W	G	V	F	H
W	**0.97**	0.01	0.01	0.01	0.00
G	0.03	**0.87**	0.05	0.01	0.05
V	0.02	0.13	**0.76**	0.03	0.06
F	0.00	0.22	0.03	**0.66**	0.09
H	0.00	0.08	0.03	0.02	**0.87**

Table 6.6: Confusion matrix of MagNet. row 15 in Table 6.4

Chapter 7

Evaluation

This chapter starts off by evaluating each research question based on the results presented in the previous chapter, discussing each research question in turn. The results are then evaluated in light of the related work presented in chapter 3, before the chapter discusses the contributions listed in section 1.5. Finally, the chapter evaluates the objective of the thesis.

7.1 Evaluation of Research Questions

The first research question of this thesis was concerned with how data-centric AI could improve the semantic segmentation of historical aerial images of riverscapes. This question was answered by first reviewing the literature as presented in chapter 3. It was found that not much research is done on data-centric AI methods for semantic segmentation which resulted in attempting to adapt similar approaches done on similar domains. Inspired by research conducted on object detection and image classification, a full cycle of data-centric AI was performed in the experiments **E1** and **E2**. The first research question was formulated as:

RQ1: *How can Data-Centric AI be used to improve the semantic segmentation results?*

To see the improvement, it is important to start with a baseline. In **E1**, a baseline for data-centric methods was provided. The average MIoU of baseline on all three test is 64.60%. This was done by first reproducing the baseline model of the previous work. It is important to mention that the results differed slightly from the previous work. To verify that this is not due to the runtime environment, sequences of the validation-loss history of the reproduced models and the original work were compared, and it was observed that they are similar. The reproduced baseline models were used for qualitative error analysis.

After reproducing the methods, common model-centric approaches were implemented. This provides an observation of how effective vanilla model-centric AI can be compared to data-centric AI. Moreover, it makes it possible to investigate the effect of applying data-centric methods on model-centric methods. The best way of feeding grayscale images into

encoders pretrained on RGB images was investigated as described in **E1**. Looking at the Table 6.1, it is obvious that simply copying the grayscale channel to make 3-channel input data is the best way to feed the data into the network. This is consistent with the work of Xie and Richmond (2019), where it was found that color features are not crucial for natural image classification. Therefore, feeding a grayscale image into a network, pretrained on colored images, without modifications, is an effective approach.

The reason for the relatively worse performance when having a convolutional layer in the beginning, is the magnitude of gradients. This layer is at the beginning of a very deep neural network and it makes it very difficult to train the parameters of that layer. When using the *convolution layer*, compared to the *copying channel*, the model performed better on the Gaula 1963, which is the most in-sample test set, and performed worse on the others. This is an indication that this approach is more prone to the overfitting problem. Ultimately, when extra features were placed in the extra channels, the overall performance of the model decreased on all the test sets. This can be explained with the notion of transfer learning. As stated before, the encoder of the model is pretrained on ImageNet. It means the weights of the encoder layers are intialized to extract low level features from the 3-channel image data. Once the nature of input is altered, and the data is filled with information other than image input, the encoder layers need to learn different parameters to extract low level features from the new data. This makes the process of learning less efficient.

Moreover, the algorithms used to fill the input channel, like Quickshift, require setting the algorithm parameters manually. These parameters were set with respect to the training set which makes them susceptible to overfitting.

Afterwards, rotation augmentation and new architectures were tested as described in **E1**. Surprisingly, rotation augmentation led to worse performance. This effect is assumed to be due to the quality of labels. Since there are noises and inconsistencies in the label, by sampling more data points using transformations such as rotation, these inconsistencies and noises result in more confusion for the training algorithm. Similarly, as it is shown in the Table 6.1, none of the more advanced models resulted in consistently improved performance on the three test sets. FPN and U-Net ResNet50 had a better performance on Gaula 1963 and Nea 1962, and worse on Gaula 1998. This can be due to the fact that Gaula 1963 and Nea 1963 are more similar to the training data and that these models are not able to generalize better than U-Net VGG16. These results show that common Model-Centric methods are not effective to achieve the objective of this work.

In the second experiment **E2**, two data-centric methods of improving the quality of annotations as well as rotation augmentation were tested. Table 6.3 shows the results of training the baseline model on both dataset V0, the initial dataset, and dataset V1, the dataset with higher quality of labels. The first row represents the results of dataset V1 without any data augmentation, and the model trained on dataset V1 outperforms the one trained on dataset V0 on all three test sets. This demonstrates that improving the validity of the dataset alone, was an effective method to improve the performance of the segmentation task. Moreover, the table also shows the results of training the baseline model on dataset V1 with the rotation augmentation. Since the model was trained five times with different random initializations, the average $\pm$ standard deviation of prediction MIoU is presented in the table. As can be seen in the table, it is clear that rotation augmentation improves the

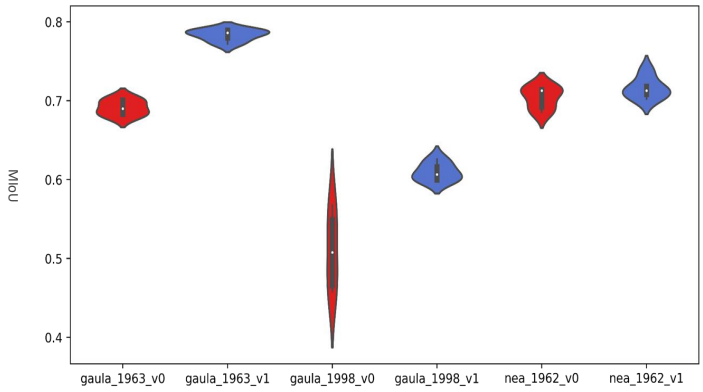


Figure 7.1: The violin diagram of test sets MIoU of initial model trained on dataset V0 and dataset V1 five times during E1 and E2. Model trained on dataset V0 is depicted in red and the one trained on dataset V1 is blue.

performance for all the test sets. The vanilla diagram of five training on dataset V1 with rotation annotation and initial dataset is illustrated in Figure 7.1. This diagram shows that data-centric methods not only improved the performance of the model, but also diminish the variance of the performance on different test sets. By looking at the variance of performance of the initial baseline, the difficulty of reproducing the old dataset becomes more clear.

To complete the evaluation, predictive uncertainty of the initial model trained on dataset V0 as well as the same model trained on dataset V1 with rotation augmentation are compared. To generate the uncertainty map of the test sets, one of the five trained models was selected for both datasets. Afterwards, the entropy of Monte Carlo Dropout with 20 samples was used to generate the uncertainty maps as described in the method chapter section 4.6. As it is mentioned previously, in all three test sets, only a subset of large images in the vicinity of rivers were annotated and considered as ground truth. To improve the visualization, one large image from each of the test sets was selected and the entire image was corrected to make the new ground truth. Uncertainty maps of the three images can be seen in Figure 7.2, where the green color corresponds to correct prediction and red represents the errors. It is observed that utilizing the data-centric methods led to decrease in both uncertainty and errors. One can perceive that correct predictions tend to have comparably less uncertainty as opposed to errors. Table 7.1 quantifies the Figure 7.2. It is shown that incorrect areas have higher uncertainty in both dataset V0 and dataset V1. When using dataset V0 for training, the average uncertainty of correct areas is 0.03 units lower and this number for dataset V1 is 0.039 units. The Table 7.1 confirms the findings of (Wickstrøm et al., 2020; Czolbe et al., 2021). Moreover, for all the test sets, the data-centric method decreased the predictive uncertainty.

As the evaluation shows, by applying the two data-centric methods which were selected by analyzing the error of the reproduced baseline, performance of the semantic segmentation considerably improved. The average of MIoU of all test sets became 71.65% when using the data-centric methods, where the same MIoU for baseline was 64.60%. It

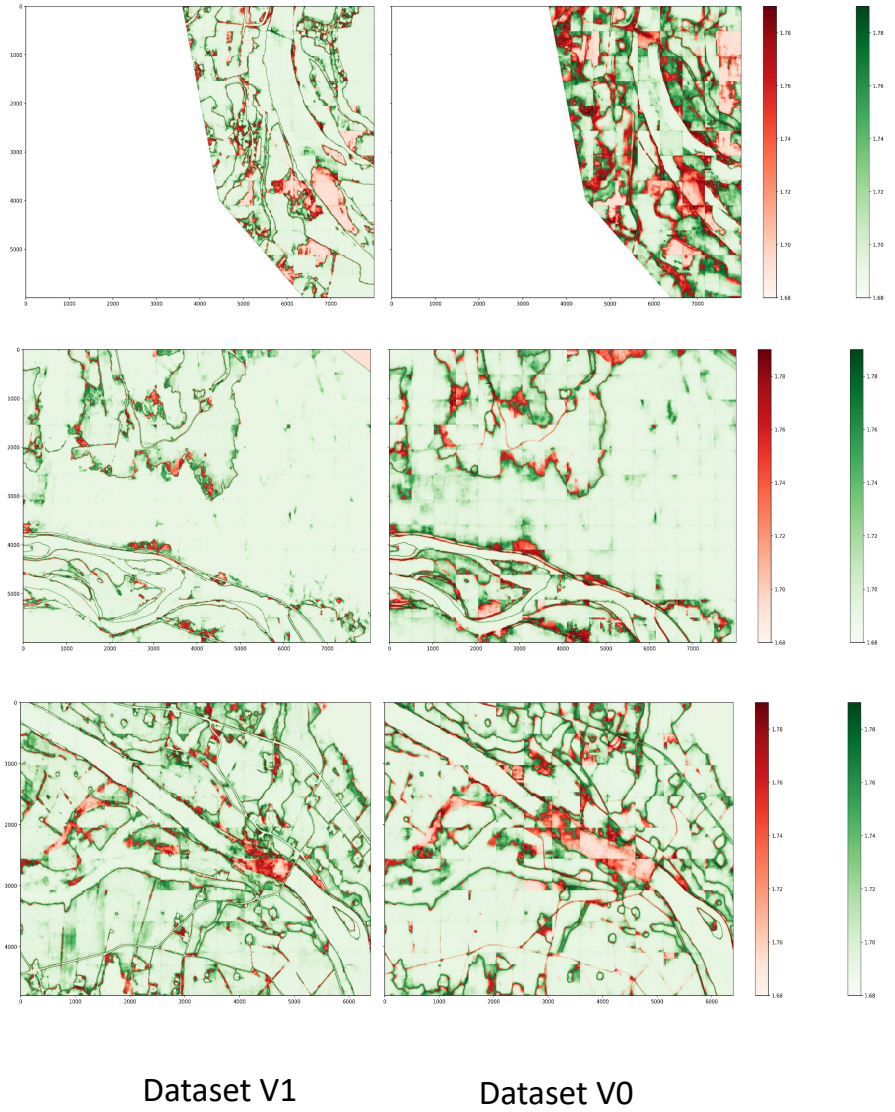


Figure 7.2: The predictive uncertainty of the initial model trained on datasets V0 and V1. The green color indicates the correct prediction and red areas are the errors.

River name	Dataset V0			Dataset V1		
	Correct average	Incorrect average	Total average	Correct average	Incorrect average	Total average
Gaula 1963	1.70	1.74	1.71	1.69	1.74	1.69
Nea 1962	1.70	1.73	1.71	1.70	1.75	1.71
Gaula 1998	1.72	1.74	1.73	1.70	1.72	1.71
Average	1.707	1.737	1.716	1.697	1.736	1.704

Table 7.1: The predictive uncertainty of model in E1 and E2. Correct and incorrect averages are the average uncertainty of correct and incorrect predictions, respectively. Total average is the average uncertainty of the whole image. Lower value means less uncertainty.

means that by improving the quality of label of the dataset and generation of new data points in a way that mitigates the class imbalance by rotation augmentation, the performance of semantic segmentation of historical aerial images of riverscapes can be improved. We found that data-centric methods in the form of improving the quality of labels and rotation augmentation, reduced the error rate of the baseline model by 19.84% which answered **RQ1**.

The second research question is formulated as:

RQ2: *Can Model-Centric methods improve the results of semantic segmentation further?*

E3 attempted to answer this question by conducting a model comparison study of model-centric methods selected in the section 3.2. *SWA*, *WCE*, two different online augmentations *OA* and new advanced deep learning architectures were the model-centric methods used in this experiment. New architectures are *FPN*, *U-Net ResNet50*, *DeepLabV3+* and *MagNet*. To clarify, as described in **E3**, first online augmentation, *OA V1*, applied intensive stochastic transformations and *OA V2* applied comparatively less transformations. Table 6.4 shows the results of the model comparison study where each row represents the configuration of the corresponding training procedure.

For simplicity the *id* of each configuration is used for reference. For example *id=1*, refers to the first row where U-Net VGG16 was trained without applying any extra model-centric methods. The table shows that using SWA improved the average performance of all five architectures, and when applied to MagNet, it increased the prediction MIoU of all three test sets. However, in the rest of the models, it resulted in improvement of the performance on two test sets and slight degradation of MIoU on the other test set. With the exception of *id=1*, SWA resulted in improvement of the results for Nea 62 and Gaula 98, which were not represented in the training set. This indicates that generally speaking, SWA led to better generalization.

As it is shown in the Table 6.4, using WCE as loss function improved the average MIoU of U-Net ResNet, FPN and MagNet by 0.22%, 0.34% and 1.17% respectively. However, this was not the case for DeepLabV3+ and when it was used in U-Net VGG16 without online augmentation. Not being able to improve when using VGG16 as encoder

might be due to the fact that VGG16 lacks the complexity to capture the representation of data when loss is changed. It is noteworthy that using WCE in U-Net ResNet, FPN and MagNet resulted in improvement of prediction of the Gaula 98 which is the most different dataset when compared to the training set.

The effect of online augmentation, OA, is more complex; in UNet VGG16, OA improved the performance of Gaula 98 but decreased the average MIoU. On the contrary, when it was used in UNet ResNet, the performance on Gaula 98 was not better but overall MIoU was better when OA V2 was applied. When comparing the performance of OA V1 and OA V2, it was observed that, except for DeepLabV3+, for all other architectures, OA V2 performed better. Looking at the combination of methods, with the exception of DeepLabV3+, the best performing configuration for all the deep learning architectures were by using SWA together with OA V2 and assigning WCE as loss function.

Additionally, DeepLabV3+ did not perform better than U-Net or FPN. This is consistent with other similar studies in which DeepLabV3+ was applied for semantic segmentation of remote sensing images. Studies show that DeepLabV3+ performed worse than architectures such as U-Net (Wang et al., 2021a), FCN (Chen et al., 2019) or FPN (Huynh et al., 2021). Unexpectedly, despite great performance of MagNet on DeepGlobe dataset, the average MIoU of this architecture over the test sets was not better than the others. Two reasons could have caused this issue. First, MagNet consists of two modules, segmentation model, which was the FPN, and the refinement module, thus it has more trainable parameters which might lead to overfitting and also more hyperparameters which requires more tuning. Second, MagNet was trained on larger patches of images. The images used for training other architecture had the dimension of 512×512 but MagNet used 2048×2048. This means MagNet had less data points for training.

Confusion matrices of the best performing model, *id=15*, along with the same configuration only without WCE loss function *id=14* are shown in Table 6.5. Compared to the baseline model Table 6.2, prediction accuracy of *water* and *vegetation* were improved for all test sets. Moreover, improvement can be observed for the *gravel* class of the Gaula 63 and Nea 62 test sets, however, this is not observed in Gaula 1998. Looking at *human constructions* and *farmland*, it is not possible to draw any conclusion about them since there are no consistent patterns. When comparing the two, U-Net ResNet50, when using WCE, improved the accuracy of *gravel* and *human construction* remarkably in the Gaula 98 test set despite that no pattern could be found. The Confusion matrix of MagNet is presented in Table 6.6. The prediction of *gravel* is markedly better in Gaula 98 and Nea 62 but it is slightly worse in Gaula 63 when compared to the best performing model. It can be observed that in general, performance of MagNet is consistently good when it comes to *gravel* and *water*. This might be due to the fact that this model incorporates more global information and does not rely solely on small patches of images.

To summarize, by comparing **E1** and **E3**, it can be concluded that data-centric methods are not only able to improve the performance of the semantic segmentation, but also can pave the way to use model-centric methods more effectively. From section 3.2, four model-centric methods were selected and a model comparison study was conducted to test if these methods could improve the performance. The results of this study is presented in Table 6.4 which shows that the answer to this research question is yes. Using U-Net

ResNet50 architecture with applying SWA and OA V2, along with having WCE loss function, achieved an average of 74.13 % MIoU over all three test sets. This means a 9.53 % improvement compared to the baseline model and a 2.48 % improvement compared to when only Data-Centric methods were applied in **E2**.

7.2 Discussion

In this section, findings and limitations of this work in terms of both performance and reproducibility are discussed.

As described in the section 3.1, not much research is done on semantic segmentation of grayscale images. In this thesis, it was found that simply copying the grayscale channel to create a 3-channel input is the best way to feed the 1-channel grayscale image into the encoder pretrained on 3-channel images. Additionally, not much work is done on data-centric methods for the task of semantic segmentation. This work showed that data-centric methods can improve the performance of semantic segmentation models. After following the data-centric AI cycle, model-centric methods proved to be more effective at improving the performance. Using both the model-centric and data-centric methods in this thesis, the existing semantic segmentation of historical aerial images was improved on all the test sets.

Intuitively, quality and consistency of labels of the dataset is more important in small scale dataset compared to the larger ones, since in large datasets, the chances that label errors and noises cancel each other out is higher. The dataset of the thesis is a relatively small dataset and therefore it is important to have high quality annotations. The findings of this thesis showed that improving the data quality not only led to decrease in the error rate of the predictions, but also reduced the predictive uncertainty. Furthermore, better label quality led to reduction of the variance of prediction of models which were trained on the same dataset multiple times with random initialization.

However, this work has limitations. First of all, as mentioned in the section 7.1, after training the baseline, the prediction results were not exactly the same as mentioned in the previous work, despite the fact that all the hyperparameters were set to be exactly the same. Additionally, the history of the validation loss of the baseline models trained in this work, were compared to the history of the validation loss of the original model. Since these histories were very similar, and performance of the baseline models on the test sets varied notably. It was decided to consider the results of reproduced models as the baseline of this thesis. Moreover, the best performing model still has 25.87 % MIoU error on average on all the test sets. It is important to consider the error rate when this model is being used for real-world applications.

It was shown in this work that predictive uncertainty is higher for areas with incorrect prediction. This finding is consistent with (Wickstrøm et al., 2020; Czolbe et al., 2021). Therefore, the predictive uncertainty of the results can provide a good indication of potential errors and can help the researchers and practitioners who will use this model to have an insight into potential errors in the prediction maps.

In terms of reproducibility of the work done in this thesis, Gundersen et al. (2022)

identified six groups of factors leading to *irreproducibility* in machine learning projects. Even though the data and codes of this work is accessible, still other factors can lead to irreproducibility of the results of this thesis. These factors are described below:

1. **Algorithmic factors:**
   (a) **Stochastic layers**: Having stochasticity in the model, like having Dropout, leads to variance in the results of different training processes.
   (b) **Random initialization, data shuffling and batch ordering**: All result in variance in the training procedure.
2. **Implementation factor:**
   (a) **Initialization seeds**: For all the experiments, except **E2**, seed values were selected to be 0. In **E2**, when the models are trained for five times, five seeds were selected at random.
   (b) **Processing unit**: If the models are trained on different processing units, results might vary (Nagarajan et al., 2018).
3. **Observation factors:**
   (a) **Data-augmentation**: Considering the stochastic nature of rotation and online augmentations used in this work. Results of models which used these augmentations might vary if the training process is repeated.
   (b) **Data split**: Data split is done randomly in this thesis and difference in data split might cause different results.
   (c) **Environment properties**: Training and testing the models in different environments could change the outcome of the models.
4. **Evaluation factor:**
   (a) **Error estimation**: For most models, due to high demand of computation, training was done only once and the test results were based on that one-time training of the model. Having a model which was trained once, makes it impossible to account for variance and have any confidence in the results.

To sum up, this thesis improved the performance of existing semantic segmentation of historical aerial images of riverscapes on all three test sets. However, it has its limitations which need to be considered for utilization.

7.3 Qualitative Error Analysis

To the best of our knowledge, no previous work conducted a thorough qualitative analysis to systematically assess the misprediction of the model. This study will provide a better insight into the performance of the model for experts who will use the predictions in real applications and makes it possible to qualitatively assess the improvement. To conduct the

analysis for each model, first the model is used to provide the predictions of all three test sets. Afterwards, each of the test sets are manually inspected and most prevalent cases of errors are selected and are referred to as *error cases*. Each error is described as (correct label:prediction error). For example (water:farmland) means the cases where *water* is misclassified as *farmland*.

First, the prediction of reproduced model of (Dalsgård, 2020), as described in **E1**, was used for analysis. Afterwards, other architectures trained on dataset V1 was used to qualitatively investigate the improvement for prediction of models presented in Table 6.4. For each test sets, a set of error cases are presented. each case is divided into two parts, first explanation of the reproduced model and afterwards the performance of other model are discussed. For simplicity, new methods are referred to by pointing at their *id* in Table 6.4. For example, MagNet id=32 referres to the last row of Table 6.4 which is MagNet with SWA, RA and OA V2 trained with WCE loss function.

7.3.1 Gaula 1963 Test Set

The following describes the most pervasive error cases found in the Gaula 1963 test set.

Water:Farmland

- **Error in the reproduced model:**

  One of the most pervasive errors that existed in predictions was mislabeling water as farmland. Model only had access to the grayscale value of the image and therefore had to predict based on only the texture of the image. Considering the similarities in texture of placid water and farmland, it was expected that these areas were difficult to predict. Figure 7.3 illustrates examples of this case.

- **Evaluation of new approaches on the problem:**

  This was mostly solved by training on dataset V1. However tiny farmland segments could be found in slow currents. These segments shrank when RA was used and were completely solved when more complex architectures were used. For example, U-Net ResNet50, DeeplabV3+ and MagNet performed very well. However, using FPN led to observing tiny fragments of farmland in the water.

Noisy human construction segments

- **Error in the reproduced model:**

  In vegetation there were some noisy human construction segments. When training annotation was checked, similar noises were found in the labels. Therefore it was assumed that this problem would be solved by improving the annotations of training data. Figure 7.4 shows the case.

- **Evaluation of new approaches on the problem:**

  This issue was resolved by training on the dataset V1. However, some dark areas of forest were mislabeled as water. The issue of dark forest being mislabeled as water

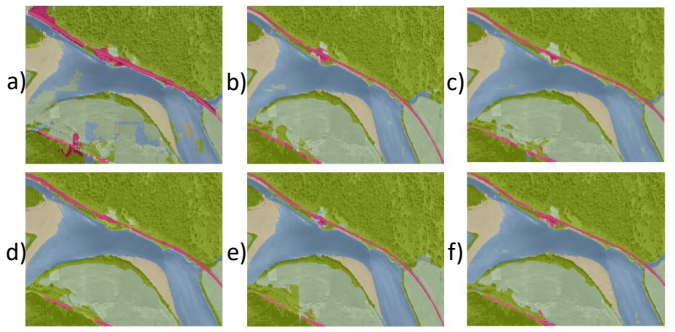


Figure 7.3: a) Reproduced model, b) U-Net VGG16 [id=3], c) U-Net VGG16 [id=9], d) MagNet [id=32], e) DeepLabV3+ [id=26], f) FPN [id=21]. Where id refers to id of configuration in the Table 6.4.

remained even when SWA, OA and new architectures were used. It is important to mention that forest area far from the river is not an area of interest for the purpose of this work and this error can be overlooked.

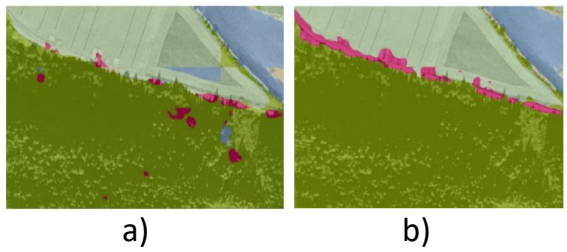


Figure 7.4: a) Reproduced model, b) U-Net VGG16 [id=3]. Where id refers to id of configuration in the Table 6.4.

Issue with the roads

- **Error in the reproduced model:**

  It is noticeable that in many places, roads were ignored. This issue could be traced back into the training data since the same errors could be seen in the training labels. This case is illustrated in Figure 7.5.

- **Evaluation of new approaches on the problem:**

  Simply by training on dataset V1 this error was solved.

New issues on Gaula 1963

The following describes the issues observed in the new predictions and not the baseline predictions.

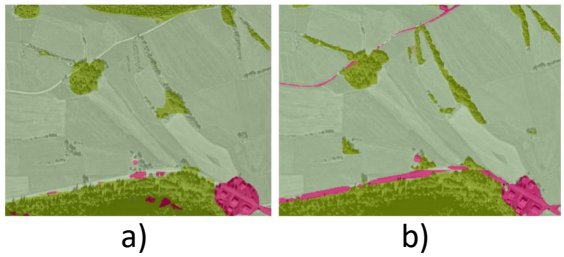


Figure 7.5: a) Reproduced model, b) U-Net VGG16 [id=3]. Where id refers to id of configuration in the Table 6.4.

- **Evaluation of new approaches on the problem:**

  Some flat areas that were neither clearly farmland nor forest are difficult to classify. It was also reflected through the predictive uncertainty. Models which were trained on dataset V1, were more uncertain in these flat areas. This issue was more visible when these areas were at the border of the image and the input image did not provide much context to the model. Figure 7.6 illustrates this issue. One explanation can be under-representation of the farmland class in dataset V1 compared to V0, meaning. In dataset V1, the balance between farmland area and vegetation is not as good as dataset V0. The reason for this imbalance can be keeping images with only vegetation class in dataset V1 which were removed in dataset V0.

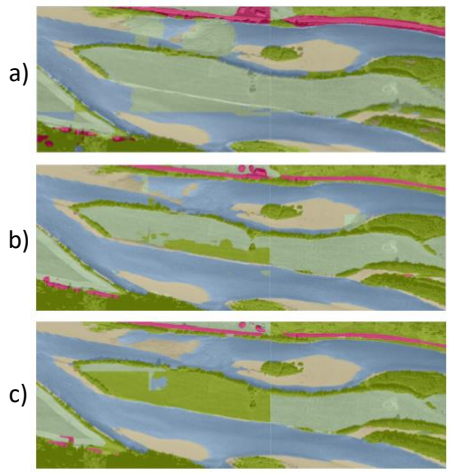


Figure 7.6: a) Reproduced model, b) U-Net VGG16 [id=3], c) DeepLabV3+ [id=26]. Where id refers to id of configuration in the Table 6.4.

7.3.2 Gaula 1998 Test Set

The most prevalent error cases found in the Gaula 1998 test set are presented in the following.

Water:Farmland

- **Error in the reproduced model:**

  Similar to Gaula 1963, it was observed that when the river is calm and broad, due to similarity of the texture of the river to farmland, water was frequently misclassified as farmland. This case is shown in the Figure 7.7.

- **Evaluation of new approaches on the problem:**

  Dataset V1 helped to alleviate the problem. Using RA, OA, SWA and new models such as FPN also led to improvement.

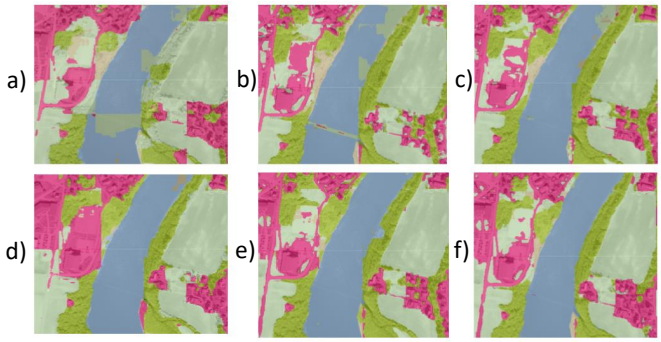


Figure 7.7: a) Reproduced model, b) U-Net VGG16 [id=3], c) U-Net VGG16 [id=9], d) DeepLabV3+ [id=26], e) FPN [id=21] e) U-Net ResNet50 [id=15]. Where id refers to id of configuration in the Table 6.4.

Vegetation:Water

- **Error in the reproduced model:**

  It was pervasive that light forest was mispredicted as water. the same type of noises was found in the training labels. This led to assumption that this issue might be solved when training labels are improved. Figure 7.8 illustrates this case.

- **Evaluation of new approaches on the problem:**

  This issue was almost completely fixed when dataset V1 was used for training.

Water:Vegetation

- **Error in the reproduced model:**

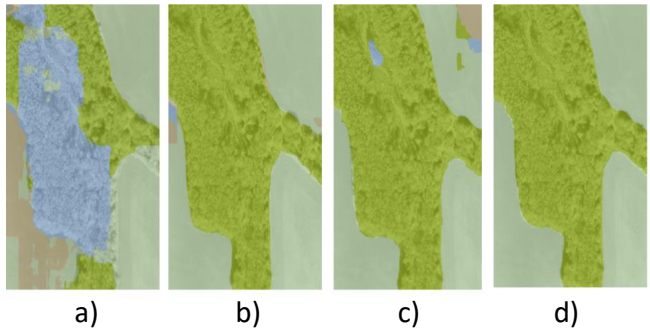


Figure 7.8: a) Reproduced model, b) U-Net VGG16 [id=1], c) U-Net VGG16 [id=4], d) U-Net VGG16 [id=8]. Where id refers to id of configuration in the Table 6.4.

Unlike Gaula 1963, misclassified segments of vegetation in the rivers as well as farmland were observed. This was more common in calm sections of the river. Figure 7.9 illustrates this case.

- **Evaluation of new approaches on the problem:**

  Dataset V1 minimized this error, It is observed that the issue was mostly fixed with using SWA and OA.

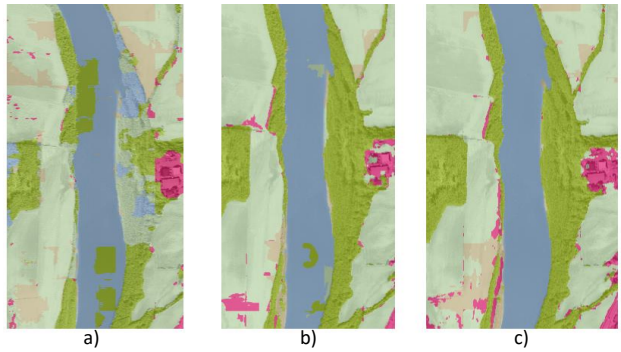


Figure 7.9: a) Reproduced model, b) U-Net VGG16 [id=1], c) U-Net VGG16 [id=8]. Where id refers to id of configuration in the Table 6.4.

Water:Gravel

- **Error in the reproduced model:**

  One part of shallow water was usually confused with gravel. This area was inspected with the presence of a domain-expert and the area is considered to be an ambiguous area where can easily be confused with gravel due to the depth of water. Figure 7.10 illustrates this case.

- **Evaluation of new approaches on the problem:**

  This problem did remain when new models were used. It was still visible when OA or SWA were applied. Moreover new architectures did not resolve the issue. It was conclude that to solve this issue more images with shallow calm water is needed to make this case less ambiguous for the model. Looking at the predictive uncertainties of this area, it was clear that more images with shallow water and slow streams are needed for training.

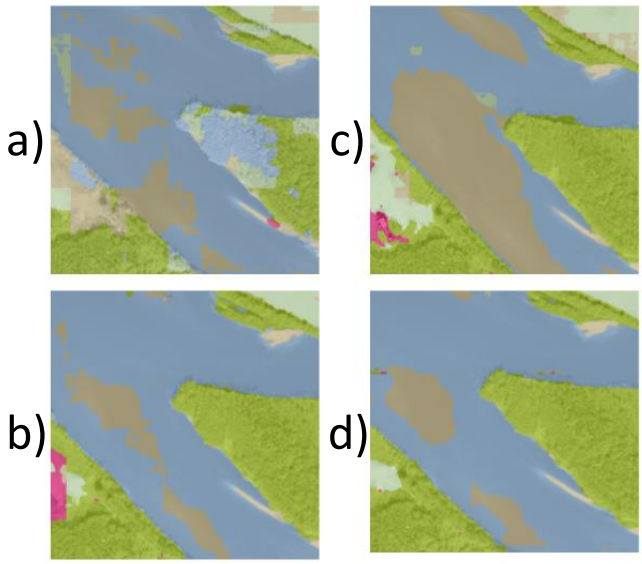


Figure 7.10: a) Reproduced model, b) U-Net VGG16 [id=1], c) FPN [id=16], d) U-Net VGG16 [id=8]. Where id refers to id of configuration in the Table 6.4.

7.3.3 Nea 1962 Test Set

The most common error cases of Nea 1962 are presented below.

Farmland:Water

- **Error in the reproduced model:**

  It was noticeable that some farmlands were mislabeled as water. This was more common at the edge of the prediction window. Figure 7.11 illustrates this case.

- **Evaluation of new approaches on the problem:**

  It was mostly solved by training on dataset V1. However, in some architectures such as DeepLabv3+, this error could be found.

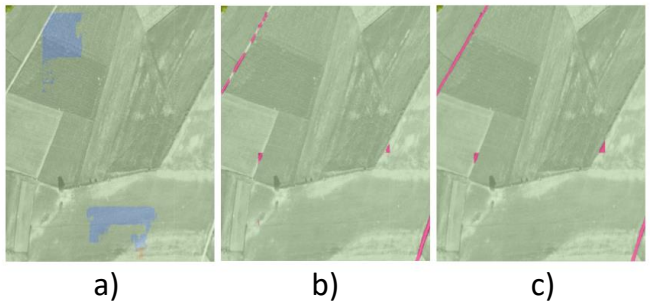


Figure 7.11: a) Reproduced model, b) U-Net VGG16 [id=3], c) U-Net ResNet [id=11]. Where id refers to id of configuration in the Table 6.4.

Issue with the roads

- **Error in the reproduced model:**

  Similar to G63 and G98, roads were frequently ignored. Figure 7.12 illustrates this case.

- **Evaluation of new approaches on the problem:**

  It was fixed with new data. Human construction seemed to have more detail which reflects the more detailed and consistent human construction annotation in the training labels. However, Some minor errors in MagNet predictions were still visible. This is because MagNet combines global and local information and therefore might overlook some details such as roads.

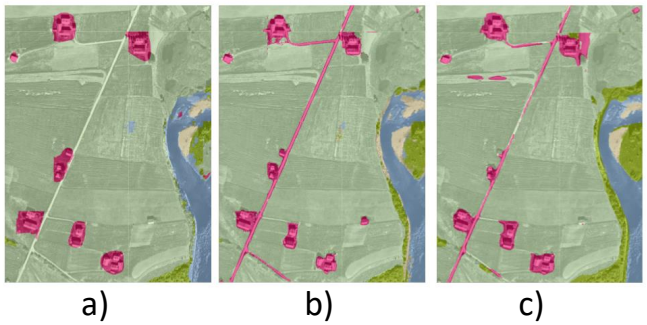


Figure 7.12: a) Reproduced model, b) U-Net VGG16 [id=3], c) MagNEt [id=32]. Where id refers to id of configuration in the Table 6.4.

Water:Farmland

- **Error in the reproduced model:**

In the main river, some parts were segmented as farmland. Figure 7.13 illustrates this case.

- **Evaluation of new approaches on the problem:**

  This problem was mitigated with dataset V1 which was then improved more when using new architectures such as U-Net ResNet50 and FPN were used. DeepLabV3+ and MagNet were very effective in terms of fixing this issue. Using DeepLabV3+ it can be observed that some part of river are now mislabeled as vegetation instead of farmland. River parts which were completely dark often mislabeled as Unknown class by FPN or U-Net. This was due to the fact that borders of images that contain no class and only plain black color were labeled as "Unknown" during the training. However, this issue is not observed in DeepLabV3+ or MagNet.

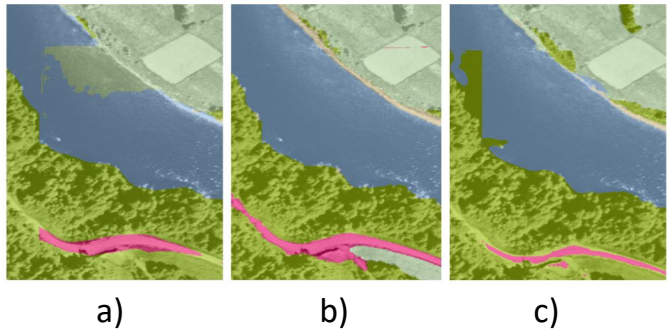


Figure 7.13: a) Reproduced model, b) U-Net VGG16 [id=3], c) DeepLabV3+ [id=26]. Where id refers to id of configuration in the Table 6.4.

Shimmering water

- **Error in the reproduced model:**

  Due to different weather conditions, speed of flow and water depth, some area of images contain shimmering water which was difficult to predict for the model. Figure 7.14 illustrates this case.

- **Evaluation of new approaches on the problem:**

  The problem was not solved by training on dataset V1. However, using Magnet and U-Net ResNet50 fixed the problem.

7.4 Contributions

In this section the contributions of the thesis are evaluated. These contributions are described as:

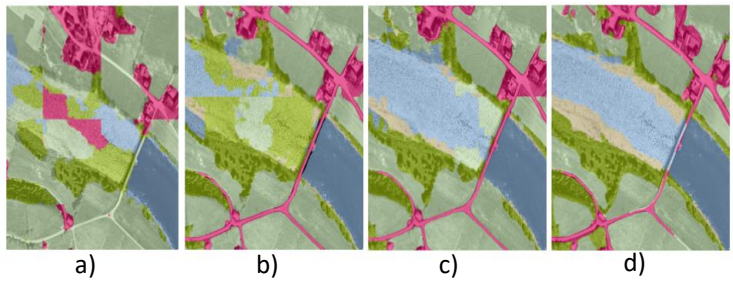


Figure 7.14: a) Reproduced model, b) U-Net VGG16 [id=3], c) U-Net ResNet50 [id=15], d) MagNet [id=32]. Where id refers to id of configuration in the Table 6.4.

C1: *Demonstrating a data-centric approach which improves the performance of existing semantic segmentation of historical aerial images of riverscapes and increased the average MIoU of test sets by 7.05%, meaning the error rate is reduced by 19.84%.*

Observing that commonly used model-centric methods were not able to improve the performance of the baseline model on all three test sets, as described in **E1**, suggested that an alternative approach was needed to improve the performance. Since the community has started to shift their focus from model-centric to data-centric AI, it was assumed that data-centric AI could lead to improvement of the performance. This assumption was formulated as a research question, RQ1. To answer this research question the full cycle of data-centric learning Figure 4.11 was followed. Two data-centric methods were selected as a result of error analysis and tested in this work, and as described in the evaluation, the results indicated that data-centric methods improved the performance of semantic segmentation on all the test sets. When both data-centric methods are employed, average test sets MIoU percentage increases from 64.60% to 71.65%.

C2: *Creating a fully annotated semantic segmentation dataset of historical aerial images of riverscapes in Norway with detailed manual annotations.*

As a result of improving the quality of the dataset. A new dataset including 87 high resolution aerial images of riverscapes was created. In this dataset aerial images were fully annotated and all the annotations were done manually with attention to details. 83 images have the resolution of 8000×6000 and the resolution of other 4 images are 6400×4800 pixels. To the best of our knowledge, this is the first semantic segmentation dataset of historical grayscale aerial images of riverscapes with manual annotations of high resolution images.

C3: *Improving the performance of semantic segmentation further by employing a set of model-centric methods after applying the data-centric methods. This improved the average MIoU of baseline on all test sets by 9.53%, which means the error rate reduced by 26.84%.*

First, a set of model-centric candidate methods were selected by studying the related research and using a model comparison study, the best perming method on the

average of all three test sets were selected. Additionally, this study indicated that by applying the data-centric methods in **E2**, model-centric methods became more effective. The model-centric methods improved the performance of the baseline model from 64.60% to 74.13%.

7.5 Evaluation of Objective

The objective of the thesis being directly is defined as:

O: *Find out how performance of existing semantic segmentation of historical aerial images of riverscapes of Norway can be improved in order to provide an out of the box tool for large scale analysis of evolution of rivers in Norway.*

In order to achieve the objective of the research, two research questions were asked. By answering **RQ1**, in the contributions **C1** and **C2**, and **RQ2**, in the contribution **C3**, the objective was achieved.

Chapter 8

Conclusion and Future Work

This chapter concludes the thesis and lists possible directions for future work.

8.1 Conclusion

The research conducted to improve the existing semantic segmentation of historical riverscapes in Norway, is presented in this thesis. After disability of improving the performance using the common model-centric methods. This thesis focused on the recent data-centric methodology. Two data-centric methods were performed which led to improvement of the performance. Data-centric methods not only improved the performance but paved the way for model-centric methods to be more effective on the performance. This work manifests the importance of data-centric ai on image semantic segmentation. It is shown that improving the quality of the dataset not only improves the performance of the model on similar test sets, but also leads to better generalization. Additionally this work manifests a case in which AI can help to have a better understanding of the ecology which is crucial for sustainable development.

In conclusion this thesis improved the performance of existing models with the help of data-centric and model-centric AI.

8.2 Future work

As it is stated in the introduction, the main goal of this research was to provide a reliable tool to be used for assessment of evolution of riverscapes through time. For that reason the primary next step of this work is to use the best performing method to conduct the assessment. Additionally, other future works are listed below:

Estimating the gain of each image annotation:

In semantic segmentation, annotation is an exceptionally laborious and time consuming procedure. As a result it is crucial to have an estimation of gain of adding

one image to the dataset in terms of performance achievement.

Correlation between dataset validity and predictive uncertainty:

Looking at the Table 7.1 Figure 7.2 there is a correlation between the quality of the training labels and the predictive uncertainty. However, there is a need for more study to investigate the correlation between the confidence of the model and validity of the data.

More investigation for feeding grayscale to RGB pretrained encoder:

In subsection 5.2.2, three methods were to investigate the best method to feed a grayscale input image into an encoder pretrained on ImageNet. One of these methods was to add a convolutional layer before the encoder to change the number of channels of the input or in other words learn the transformation from 1-d grayscale data to 3-d RGB. However, it was shown that this method did not perform better than simply copying the grayscale channel 3 times to have a 3-channel input. It was assumed that gradient vanishing might cause this issue. One direction of study can be implementing a residual connection for this layer and investigate whether this leads to any improvement or not.

Other architecture and augmentation methods:

With the promising performance of transformer-based architecture in computer vision tasks. It can be a good idea to investigate their performance on the developed dataset of this work. Additionally, AutoAugment (Cubuk et al., 2018) can be utilized to learn an augmentation policy specific to the dataset.

Investigation of the difference between Data-Centric methods for different tasks:

As described in the section 3.3, in image classification, only one label is assigned to an image. Having only one label per image makes it easier to deal with anomalies in the data by using clustering methods. However, semantic segmentation is a per-pixel classification task which makes it difficult to use algorithms such as clustering to deal with anomalies. Overall, one can assume that data-centric methods designed to be used for semantic segmentation should be more sophisticated. An investigation is required to determine whether this assumption is correct.

Bibliography

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. URL: https://www.tensorflow.org/. software available from tensorflow.org.

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2012. Slic superpixels compared to state-of-the-art superpixel methods. IEEE Transactions on Pattern Analysis and Machine Intelligence 34, 2274–2282. doi:10.1109/TPAMI.2012.120.

Alam, S., Tomar, N.K., Thakur, A., Jha, D., Rauniyar, A., 2020. Automatic polyp segmentation using u-net-resnet50. URL: https://arxiv.org/abs/2012.15247, doi:10.48550/ARXIV.2012.15247.

Alemohammad, H., Booth, K., 2020. Landcovernet: A global benchmark land cover classification training dataset. CoRR abs/2012.03111. URL: https://arxiv.org/abs/2012.03111, arXiv:2012.03111.

Alfredsen, K., Dalsgård, A., Shamsaliei, S., Halleraker, J.H., Gundersen, O.E., 2021a. Towards an automatic characterization of riverscape development by deep learning. River Research and Applications 38, 810–816. URL: https://doi.org/10.1002/rra.3927, doi:10.1002/rra.3927.

Alfredsen, K., Dalsgård, A., Shamsaliei, S., Halleraker, J.H., Gundersen, O.E., 2021b. Towards an automatic characterization of riverscape development by deep learning. River Research and Applications 38, 810–816. URL: https://doi.org/10.1002/rra.3927, doi:10.1002/rra.3927.

Andriluka, M., Uijlings, J.R.R., Ferrari, V., 2018. Fluid annotation: a human-machine collaboration interface for full image annotation. CoRR abs/1806.07527. URL: http://arxiv.org/abs/1806.07527, arXiv:1806.07527.

Aroyo, L., Lease, M., Paritosh, P.K., Schaekermann, M., 2021. Data excellence for AI: why should you care. CoRR abs/2111.10391. URL: https://arxiv.org/abs/2111.10391, arXiv:2111.10391.

Baetens, L., Desjardins, C., Hagolle, O., 2019. Validation of copernicus sentinel-2 cloud masks obtained from maja, sen2cor, and fmask processors using reference cloud masks generated with a supervised active learning procedure. Remote Sensing 11. URL: https://www.mdpi.com/2072-4292/11/4/433, doi:10.3390/rs11040433.

Berman, M., Blaschko, M.B., 2017. Optimization of the jaccard index for image segmentation with the lovász hinge. CoRR abs/1705.08790. URL: http://arxiv.org/abs/1705.08790, arXiv:1705.08790.

Boguszewski, A., Batorski, D., Ziemba-Jankowska, N., Dziedzic, T., Zambrzycka, A., 2021. Landcover.ai: Dataset for automatic mapping of buildings, woodlands, water and roads from aerial imagery, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pp. 1102–1110.

Bre, F., Gimenez, J.M., Fachinotti, V.D., 2018. Prediction of wind pressure coefficients on building surfaces using artificial neural networks. Energy and Buildings 158, 1429–1441. URL: https://doi.org/10.1016/j.enbuild.2017.11.045, doi:10.1016/j.enbuild.2017.11.045.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D., 2020. Language models are few-shot learners, in: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc.. pp. 1877–1901. URL: https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Buslaev, A., Iglovikov, V.I., Khvedchenya, E., Parinov, A., Druzhinin, M., Kalinin, A.A., 2020. Albumentations: Fast and flexible image augmentations. Information 11. URL: https://www.mdpi.com/2078-2489/11/2/125, doi:10.3390/info11020125.

Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A.L., Zhou, Y., 2021. Transunet: Transformers make strong encoders for medical image segmentation. CoRR abs/2102.04306. URL: https://arxiv.org/abs/2102.04306, arXiv:2102.04306.

Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2014. Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv preprint arXiv:1412.7062 .

Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2017a. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. arXiv:1606.00915.

Chen, L.C., Papandreou, G., Schroff, F., Adam, H., 2017b. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 .

Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European conference on computer vision (ECCV), pp. 801–818.

Chen, W., Jiang, Z., Wang, Z., Cui, K., Qian, X., 2019. Collaborative global-local networks for memory-efficient segmentation of ultra-high resolution images, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE. URL: https://doi.org/10.1109/cvpr.2019.00913, doi:10.1109/cvpr.2019.00913.

Chiu, M.T., Xu, X., Wei, Y., Huang, Z., Schwing, A.G., Brunner, R., Khachatrian, H., Karapetyan, H., Dozier, I., Rose, G., Wilson, D., Tudor, A., Hovakimyan, N., Huang, T.S., Shi, H., 2020. Agriculture-vision: A[jenssen] large aerial image database for agricultural pattern analysis, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2825–2835. doi:10.1109/CVPR42600.2020.00290.

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

Cireşan, D.C., Meier, U., Gambardella, L.M., Schmidhuber, J., 2010. Deep, big, simple neural nets for handwritten digit recognition. Neural Computation 22, 3207–3220. URL: https://doi.org/10.1162/neco_a_00052, doi:10.1162/neco_a_00052.

Comaniciu, D., Meer, P., 2002. Mean shift: a robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 603–619. doi:10.1109/34.1000236.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Machine learning 20, 273–297.

Cubuk, E.D., Zoph, B., Mané, D., Vasudevan, V., Le, Q.V., 2018. Autoaugment: Learning augmentation policies from data. CoRR abs/1805.09501. URL: http://arxiv.org/abs/1805.09501, arXiv:1805.09501.

Czolbe, S., Arnavaz, K., Krause, O., Feragen, A., 2021. Is segmentation uncertainty useful?, in: Feragen, A., Sommer, S., Schnabel, J., Nielsen, M. (Eds.), Information Processing in Medical Imaging, Springer International Publishing, Cham. pp. 715–726.

Dalsgård, A.S., 2020. TSegmentation of river ecology using deep learning on aerial images. Master's thesis. Norwegian University of Science and Technology.

Data-Centric, 2021. Data centric website. https://datacentricai.org/, accessed: 16.03.2022.

Dechesne, C., Lassalle, P., Lefèvre, S., 2021. Bayesian u-net: Estimating uncertainty in semantic segmentation of earth observation images. Remote Sensing 13. URL: https://www.mdpi.com/2072-4292/13/19/3836, doi:10.3390/rs13193836.

Demir, I., Koperski, K., Lindenbaum, D., Pang, G., Huang, J., Basu, S., Hughes, F., Tuia, D., Raskar, R., 2018. DeepGlobe 2018: A challenge to parse the earth through satellite images, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE. URL: https://doi.org/10.1109/cvprw.2018.00031, doi:10.1109/cvprw.2018.00031.

Devries, T., Taylor, G.W., 2017. Improved regularization of convolutional neural networks with cutout. CoRR abs/1708.04552. URL: http://arxiv.org/abs/1708.04552, arXiv:1708.04552.

Fei-Fei, L., Fergus, R., Perona, P., 2004. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories, in: 2004 Conference on Computer Vision and Pattern Recognition Workshop, pp. 178–178. doi:10.1109/CVPR.2004.383.

Foga, S., Scaramuzza, P.L., Guo, S., Zhu, Z., Dilley, R.D., Beckmann, T., Schmidt, G.L., Dwyer, J.L., Joseph Hughes, M., Laue, B., 2017. Cloud detection algorithm comparison and validation for operational landsat data products. Remote Sensing of Environment 194, 379–390. URL: https://www.sciencedirect.com/science/article/pii/S0034425717301293, doi:https://doi.org/10.1016/j.rse.2017.03.026.

Francis, A., Mrziglod, J., Sidiropoulos, P., Muller, J.P., . Sentinel-2 cloud mask catalogue. URL: https://zenodo.org/record/4172871#.Yml0mRxBwUF. accessed: 2022-05-27.

Gal, Y., Ghahramani, Z., 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning, in: international conference on machine learning, PMLR. pp. 1050–1059.

Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D., Wilson, A.G., 2018. Loss surfaces, mode connectivity, and fast ensembling of dnns, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA. p. 8803–8812.

Gebrehiwot, A., Hashemi-Beni, L., Thompson, G., Kordjamshidi, P., Langan, T.E., 2019. Deep convolutional neural network for flood extent mapping using unmanned aerial vehicles data. Sensors 19. URL: https://www.mdpi.com/1424-8220/19/7/1486, doi:10.3390/s19071486.

Gerke, M., Rottensteiner, F., Wegner, J.D., Gunho Sohn, 2014. Isprs semantic labeling contest URL: http://rgdoi.net/10.13140/2.1.3570.9445, doi:10.13140/2.1.3570.9445.

Gonzalez, R.C., Woods, R.E., 2008. Digital image processing. Prentice Hall, Upper Saddle River, N.J. URL: http://www.amazon.com/Digital-Image-Processing-3rd-Edition/dp/013168728X.

Goodfellow, I.J., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press, Cambridge, MA, USA. http://www.deeplearningbook.org.

Grill, G., Lehner, B., Thieme, M., Geenen, B., Tickner, D., Antonelli, F., Babu, S., Borrelli, P., Cheng, L., Crochetiere, H., Macedo, H.E., Filgueiras, R., Goichot, M., Higgins, J., Hogan, Z., Lip, B., McClain, M.E., Meng, J., Mulligan, M., Nilsson, C., Olden, J.D., Opperman, J.J., Petry, P., Liermann, C.R., Sáenz, L., Salinas-Rodríguez, S., Schelle, P., Schmitt, R.J.P., Snider, J., Tan, F., Tockner, K., Valdujo, P.H., van Soesbergen, A., Zarfl, C., 2019. Mapping the world's free-flowing rivers. Nature 569, 215–221. URL: https://doi.org/10.1038/s41586-019-1111-9, doi:10.1038/s41586-019-1111-9.

Grizzetti, B., Pistocchi, A., Liquete, C., Udias, A., Bouraoui, F., van de Bund, W., 2017. Human pressures and ecological status of european rivers. Scientific Reports 7. URL: https://doi.org/10.1038/s41598-017-00324-3, doi:10.1038/s41598-017-00324-3.

Gundersen, O.E., Coakley, K., Kirkpatrick, C., 2022. Sources of irreproducibility in machine learning: A review. arXiv preprint arXiv:2204.07610 .

Guo, Y., Shi, H., Kumar, A., Grauman, K., Rosing, T., Feris, R.S., 2018. Spottune: Transfer learning through adaptive fine-tuning. CoRR abs/1811.08737. URL: http://arxiv.org/abs/1811.08737, arXiv:1811.08737.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

Hoyer, L., Dai, D., Chen, Y., Koring, A., Saha, S., Van Gool, L., 2021. Three ways to improve semantic segmentation with self-supervised depth estimation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11130–11140.

Hughes, M.J., Hayes, D.J., 2014. Automated detection of cloud and cloud shadow in single-date landsat imagery using neural networks and spatial post-processing. Remote Sensing 6, 4907–4926. URL: https://www.mdpi.com/2072-4292/6/6/4907, doi:10.3390/rs6064907.

Huynh, C., Tran, A.T., Luu, K., Hoai, M., 2021. Progressive semantic segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021, Computer Vision Foundation / IEEE. pp. 16755–16764. URL: https://openaccess.thecvf.com/content/CVPR2021/html/Huynh_Progressive_Semantic_Segmentation_CVPR_2021_paper.html.

Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D.P., Wilson, A.G., 2018. Averaging weights leads to wider optima and better generalization. CoRR abs/1803.05407. URL: http://arxiv.org/abs/1803.05407, arXiv:1803.05407.

Jadon, S., 2020. A survey of loss functions for semantic segmentation, in: 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), IEEE. pp. 1–7.

Kaiser, P., Wegner, J.D., Lucchi, A., Jaggi, M., Hofmann, T., Schindler, K., 2017. Learning aerial image segmentation from online maps. IEEE Transactions on Geoscience and Remote Sensing 55, 6054–6068. doi:10.1109/TGRS.2017.2719738.

Kartvarket, 2021. Norge i bilder. https://www.norgeibilder.no/, accessed: 12.10.2021.

Kemker, R., Kanan, C., 2017. Deep neural networks for semantic segmentation of multispectral remote sensing imagery. CoRR abs/1703.06452. URL: http://arxiv.org/abs/1703.06452, arXiv:1703.06452.

Keskar, N.S., Mudigere, D., Nocedal, J., Smelyanskiy, M., Tang, P.T.P., 2016. On large-batch training for deep learning: Generalization gap and sharp minima. CoRR abs/1609.04836. URL: http://arxiv.org/abs/1609.04836, arXiv:1609.04836.

Kiela, D., Bartolo, M., Nie, Y., Kaushik, D., Geiger, A., Wu, Z., Vidgen, B., Prasad, G., Singh, A., Ringshia, P., Ma, Z., Thrush, T., Riedel, S., Waseem, Z., Stenetorp, P., Jia, R., Bansal, M., Potts, C., Williams, A., 2021. Dynabench: Rethinking benchmarking in NLP. CoRR abs/2104.14337. URL: https://arxiv.org/abs/2104.14337, arXiv:2104.14337.

Kim, J., Sangjun, O., Kim, Y., Lee, M., 2016. Convolutional neural network with biologically inspired retinal structure. Procedia Computer Science 88, 145–154.

Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization, in: Bengio, Y., LeCun, Y. (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings. URL: http://arxiv.org/abs/1412.6980.

Kirillov, A., Wu, Y., He, K., Girshick, R., 2020. Pointrend: Image segmentation as rendering, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9799–9808.

Kohl, S.A.A., Romera-Paredes, B., Meyer, C., Fauw, J.D., Ledsam, J.R., Maier-Hein, K.H., Eslami, S.M.A., Rezende, D.J., Ronneberger, O., 2018. A probabilistic u-net for segmentation of ambiguous images, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA. p. 6965–6975.

Kotaridis, I., Lazaridou, M., 2021. Remote sensing image segmentation advances: A meta-analysis. ISPRS Journal of Photogrammetry and Remote Sensing 173, 309–322. URL: https://www.sciencedirect.com/science/article/pii/S0924271621000265, doi:https://doi.org/10.1016/j.isprsjprs.2021.01.020.

Krawczyk, B., 2016. Learning from imbalanced data: open challenges and future directions. Progress in Artificial Intelligence 5, 221–232.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, in: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

LabelImg, T., 2015. labelimg. https://github.com/tzutalin/labelImg.

Laboratory, D.S.T., . Dstl satellite imagery feature detection. URL: https://www.kaggle.com/c/dstl-satellite-imagery-feature-detection.

Lakshminarayanan, B., Pritzel, A., Blundell, C., 2017. Simple and scalable predictive uncertainty estimation using deep ensembles, in: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2017/file/9ef2ed4b7fd2c810847ffa5fa85bce38-Paper.pdf.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. nature 521, 436–444.

Lecun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE 86, 2278–2324. doi:10.1109/5.726791.

Lee, H., Grosse, R., Ranganath, R., Ng, A.Y., 2011. Unsupervised learning of hierarchical representations with convolutional deep belief networks. Commun. ACM 54, 95–103. URL: https://doi.org/10.1145/2001269.2001295, doi:10.1145/2001269.2001295.

Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., Talwalkar, A., 2018. Hyperband: A novel bandit-based approach to hyperparameter optimization. Journal of Machine Learning Research 18, 1–52. URL: http://jmlr.org/papers/v18/16-558.html.

Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S., 2017. Feature pyramid networks for object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2117–2125.

Liu, P., Choo, K.K.R., Wang, L., Huang, F., 2017. Svm or deep learning? a comparative study on remote sensing image classification. Soft Computing 21, 7053–7065.

Liu, W., Rabinovich, A., Berg, A.C., 2015. Parsenet: Looking wider to see better. arXiv preprint arXiv:1506.04579 .

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swin transformer: Hierarchical vision transformer using shifted windows. CoRR abs/2103.14030. URL: https://arxiv.org/abs/2103.14030, arXiv:2103.14030.

Liu, Z., Mao, H., Wu, C., Feichtenhofer, C., Darrell, T., Xie, S., 2022. A convnet for the 2020s. CoRR abs/2201.03545. URL: https://arxiv.org/abs/2201.03545, arXiv:2201.03545.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE. URL: https://doi.org/10.1109/cvpr.2015.7298965, doi:10.1109/cvpr.2015.7298965.

Long, Y., Xia, G., Li, S., Yang, W., Yang, M.Y., Zhu, X.X., Zhang, L., Li, D., 2020. Dirs: On creating benchmark datasets for remote sensing image interpretation. CoRR abs/2006.12485. URL: https://arxiv.org/abs/2006.12485, arXiv:2006.12485.

Louizos, C., Welling, M., 2016. Structured and efficient variational deep learning with matrix gaussian posteriors. CoRR abs/1603.04733. URL: http://arxiv.org/abs/1603.04733, arXiv:1603.04733.

Lu, H., Ma, L., Fu, X., Liu, C., Wang, Z., Tang, M., Li, N., 2020. Landslides information extraction using object-oriented image analysis paradigm based on deep learning and transfer learning. Remote Sensing 12. URL: https://www.mdpi.com/2072-4292/12/5/752, doi:10.3390/rs12050752.

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

Minaee, S., Boykov, Y.Y., Porikli, F., Plaza, A.J., Kehtarnavaz, N., Terzopoulos, D., 2021. Image segmentation using deep learning: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence , 1–1doi:10.1109/TPAMI.2021.3059968.

Mnih, V., 2013. Machine Learning for Aerial Image Labeling. Ph.D. thesis. CAN. AAINR96184.

Motamedi, M., Sakharnykh, N., Kaldewey, T., 2021. A data-centric approach for training deep neural networks with less data. arXiv preprint arXiv:2110.03613 .

Mountrakis, G., Im, J., Ogole, C., 2011. Support vector machines in remote sensing: A review. ISPRS Journal of Photogrammetry and Remote Sensing 66, 247–259.

Murphy, K.P., 2022. Probabilistic Machine Learning: An introduction. MIT Press. URL: probml.ai.

Nagarajan, P., Warnell, G., Stone, P., 2018. The impact of nondeterminism on reproducibility in deep reinforcement learning .

Nekrasov, V., Chen, H., Shen, C., Reid, I.D., 2018. Fast neural architecture search of compact semantic segmentation models via auxiliary cells. CoRR abs/1810.10804. URL: http://arxiv.org/abs/1810.10804, arXiv:1810.10804.

Ng, A., Laird, D., He, L., 2021. Data-centric ai competition, 2021. https://https-deeplearning-ai.github.io/data-centric-comp/, accessed: 16.03.2022.

Nicholas Pilkington, Stacey Svetlichnaya, T.H., . Dronedeploy machine learning segmentation benchmark. URL: https://github.com/dronedeploy/dd-ml-segmentation-benchmark. accessed: 2022-05-27.

Northcutt, C., Jiang, L., Chuang, I., 2021a. Confident learning: Estimating uncertainty in dataset labels. Journal of Artificial Intelligence Research 70, 1373–1411.

Northcutt, C.G., Athalye, A., Mueller, J., 2021b. Pervasive label errors in test sets destabilize machine learning benchmarks URL: https://arxiv.org/abs/2103.14749, doi:10.48550/ARXIV.2103.14749.

Olsson, V., Tranheden, W., Pinto, J., Svensson, L., 2020. Classmix: Segmentation-based data augmentation for semi-supervised learning. CoRR abs/2007.07936. URL: https://arxiv.org/abs/2007.07936, arXiv:2007.07936.

Papp, D., Szűcs, G., Knoll, Z., 2019. Difference based query strategies in active learning, in: 2019 IEEE 17th International Symposium on Intelligent Systems and Informatics (SISY), pp. 35–40. doi:10.1109/SISY47553.2019.9111587.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. Pytorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems 32. Curran Associates, Inc., pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-libra
pdf.

Phillips, E., 2010. How to get a PhD a handbook for students and their supervisors /. 5th ed. rev. and updated. ed., McGraw-Hill Open University Press, Maidenhead.

Piégay, H., Arnaud, F., Belletti, B., Bertrand, M., Bizzi, S., Carbonneau, P., Dufour, S., Liébault, F., Ruiz-Villanueva, V., Slater, L., 2020. Remotely sensed rivers in the anthropocene: state of the art and prospects. Earth Surface Processes and Landforms 45, 157–188. URL: https://doi.org/10.1002/esp.4787, doi:10.1002/esp.4787.

Priyanka, N, S., Lal, S., Nalini, J., Reddy, C.S., Dell'Acqua, F., 2022. DIResUNet: Architecture for multiclass semantic segmentation of high resolution remote sensing imagery data. Applied Intelligence URL: https://doi.org/10.1007/s10489-022-03310-z, doi:10.1007/s10489-022-03310-z.

QGIS Development Team, 2009. QGIS Geographic Information System. Open Source Geospatial Foundation. URL: http://qgis.osgeo.org.

Rakhlin, A., Davydow, A., Nikolenko, S., 2018. Land cover classification from satellite imagery with u-net and lovász-softmax loss, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 262–266.

Ratajczak, R., Crispim-Junior, C.F., Faure, E., Fervers, B., Tougne, L., 2019. Automatic land cover reconstruction from historical aerial images: An evaluation of features extraction and classification algorithms. IEEE Transactions on Image Processing 28, 3357–3371. doi:10.1109/TIP.2019.2896492.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer. pp. 234–241.

Roth, A., Wüstefeld, K., Weichert, F., 2021. A data-centric augmentation approach for disturbed sensor image segmentation. Journal of Imaging 7. URL: https://www.mdpi.com/2313-433X/7/10/206, doi:10.3390/jimaging7100206.

Ruder, S., 2016. An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747 .

Ruppert, D., 1988. Efficient estimations from a slowly convergent robbins-monro process .

Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T., 2007. LabelMe: A database and web-based tool for image annotation. International Journal of Computer Vision 77, 157–173. URL: https://doi.org/10.1007/s11263-007-0090-8, doi:10.1007/s11263-007-0090-8.

Samy, M., Amer, K., Eissa, K., Shaker, M., ElHelw, M., 2018. NU-net: Deep residual wide field of view convolutional neural network for semantic segmentation, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE. URL: https://doi.org/10.1109/cvprw.2018.00050, doi:10.1109/cvprw.2018.00050.

Schmarje, L., Liao, Y.H., Koch, R., 2021. A data-centric image classification benchmark URL: https://datacentricai.org/neurips21/papers/64_CameraReady_camera-readydcaiv2.pdf.

Schmitt, M., Hughes, L.H., Qiu, C., Zhu, X.X., 2019. SEN12ms – a CURATED DATASET OF GEOREFERENCED MULTI-SPECTRAL SENTINEL-1/2 IMAGERY FOR DEEP LEARNING AND DATA FUSION. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences IV-2/W7, 153–160.

URL: https://doi.org/10.5194/isprs-annals-iv-2-w7-153-2019, doi:10.5194/isprs-annals-iv-2-w7-153-2019.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 .

Seferbekov, S., Iglovikov, V., Buslaev, A., Shvets, A., 2018a. Feature pyramid network for multi-class land segmentation, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 272–2723. doi:10.1109/CVPRW.2018.00051.

Seferbekov, S., Iglovikov, V., Buslaev, A., Shvets, A., 2018b. Feature pyramid network for multi-class land segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 272–275.

Sentinelhub, . Example dataset of eopatches for slovenia 2019. URL: http://eo-learn.sentinel-hub.com/?prefix=. accessed: 2022-05-27.

Shao, Z., Yang, K., Zhou, W., 2018. Performance evaluation of single-label and multi-label remote sensing image retrieval using a dense labeling dataset. Remote Sensing 10. URL: https://www.mdpi.com/2072-4292/10/6/964, doi:10.3390/rs10060964.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 .

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research 15, 1929–1958.

Stewart, A.J., Robinson, C., Corley, I.A., Ortiz, A., Lavista Ferres, J.M., Banerjee, A., 2021. TorchGeo: deep learning with geospatial data. arXiv preprint arXiv:2111.08872 URL: https://github.com/microsoft/torchgeo.

Strudel, R., Garcia, R., Laptev, I., Schmid, C., 2021. Segmenter: Transformer for semantic segmentation, in: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE. URL: https://doi.org/10.1109/iccv48922.2021.00717, doi:10.1109/iccv48922.2021.00717.

Sun, K., Xiao, B., Liu, D., Wang, J., 2019. Deep high-resolution representation learning for human pose estimation, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5686–5696. doi:10.1109/CVPR.2019.00584.

Takase, S., Kiyono, S., 2021. Lessons on parameter sharing across layers in transformers. ArXiv abs/2104.06022.

Tang, Z., Gao, Y., Karlinsky, L., Sattigeri, P., Feris, R., Metaxas, D., 2020. Onlineaugment: Online data augmentation with less domain knowledge, in: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (Eds.), Computer Vision – ECCV 2020, Springer International Publishing, Cham. pp. 313–329.

Terzi, R., Azginoglu, N., Terzi, D.S., 2021. False positive repression: Data centric pipeline for object detection in brain MRI. Concurrency and Computation: Practice and Experience URL: https://doi.org/10.1002/cpe.6821, doi:10.1002/cpe.6821.

Tong, X., Xia, G., Lu, Q., Shen, H., Li, S., You, S., Zhang, L., 2018. Learning transferable deep models for land-use classification with high-resolution remote sensing images. CoRR abs/1807.05713. URL: http://arxiv.org/abs/1807.05713, arXiv:1807.05713.

Tritrong, N., Rewatbowornwong, P., Suwajanakorn, S., 2021. Repurposing gans for one-shot semantic part segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4475–4485.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention is all you need. CoRR abs/1706.03762. URL: http://arxiv.org/abs/1706.03762, arXiv:1706.03762.

Vedaldi, A., Soatto, S., 2008. Quick shift and kernel methods for mode seeking, in: Forsyth, D., Torr, P., Zisserman, A. (Eds.), Computer Vision – ECCV 2008, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 705–718.

Viseras, A., Losada, R.O., Merino, L., 2016. Planning with ants. International Journal of Advanced Robotic Systems 13, 172988141666407. URL: https://doi.org/10.1177/1729881416664078, doi:10.1177/1729881416664078.

Volpi, M., Ferrari, V., 2015. Semantic segmentation of urban scenes by learning local class interactions, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1–9. doi:10.1109/CVPRW.2015.7301377.

Wang, J., Zheng, Z., Ma, A., Lu, X., Zhong, Y., 2021a. LoveDA: A remote sensing land-cover dataset for domain adaptive semantic segmentation, in: Vanschoren, J., Yeung, S. (Eds.), Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks. URL: https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/file/4e732ced3463d06de0ca9a15b6153677-Paper-round2.pdf.

Wang, J., Zheng, Z., Ma, A., Lu, X., Zhong, Y., 2021b. Loveda: A remote sensing land-cover dataset for domain adaptive semantic segmentation. CoRR abs/2110.08733. URL: https://arxiv.org/abs/2110.08733, arXiv:2110.08733.

Whang, S.E., Lee, J.G., 2020. Data collection and quality challenges for deep learning. Proceedings of the VLDB Endowment 13, 3429–3432. URL: https://doi.org/10.14778/3415478.3415562, doi:10.14778/3415478.3415562.

Wickstrøm, K., Kampffmeyer, M., Jenssen, R., 2020. Uncertainty and interpretability in convolutional neural networks for semantic segmentation of colorectal polyps. Medical Image Analysis 60, 101619. URL: https://www.sciencedirect.com/science/article/pii/S1361841519301574, doi:https://doi.org/10.1016/j.media.2019.101619.

Wohl, E., 2019. Forgotten legacies: Understanding and mitigating historical human alterations of river corridors. Water Resources Research 55, 5181–5201. URL: https://doi.org/10.1029/2018wr024433, doi:10.1029/2018wr024433.

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P., 2021. Segformer: Simple and efficient design for semantic segmentation with transformers, in: Ranzato, M., Beygelzimer, A., Dauphin, Y.N., Liang, P., Vaughan, J.W. (Eds.), Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, pp. 12077–12090. URL: https://proceedings.neurips.cc/paper/2021/hash/64f1f27bf1b4ec22924fd0acb550c235-Abstract.html.

Xie, Y., Richmond, D., 2019. Pre-training on grayscale imagenet improves medical image classification, in: Leal-Taixé, L., Roth, S. (Eds.), Computer Vision – ECCV 2018 Workshops, Springer International Publishing, Cham. pp. 476–484.

Yakubovskiy, P., 2019. Segmentation models. https://github.com/qubvel/segmentation_models.

Yang, K., Qinami, K., Fei-Fei, L., Deng, J., Russakovsky, O., 2020. Towards fairer datasets: Filtering and balancing the distribution of the people subtree in the imagenet hierarchy, in: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, Association for Computing Machinery, New York, NY, USA. p. 547–558. URL: https://doi.org/10.1145/3351095.3375709, doi:10.1145/3351095.3375709.

Yuan, Y., Chen, X., Wang, J., 2019a. Object-contextual representations for semantic segmentation. CoRR abs/1909.11065. URL: http://arxiv.org/abs/1909.11065, arXiv:1909.11065.

Yuan, Y., Chen, X., Wang, J., 2019b. Object-contextual representations for semantic segmentation. CoRR abs/1909.11065. URL: http://arxiv.org/abs/1909.11065, arXiv:1909.11065.

Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y., 2019. Cutmix: Regularization strategy to train strong classifiers with localizable features. CoRR abs/1905.04899. URL: http://arxiv.org/abs/1905.04899, arXiv:1905.04899.

Zhai, X., Kolesnikov, A., Houlsby, N., Beyer, L., 2021. Scaling vision transformers. CoRR abs/2106.04560. URL: https://arxiv.org/abs/2106.04560, arXiv:2106.04560.

Zhang, H., Cissé, M., Dauphin, Y.N., Lopez-Paz, D., 2018. mixup: Beyond empirical risk minimization, in: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, OpenReview.net. URL: https://openreview.net/forum?id=r1Ddp1-Rb.

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.Y., 2022. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. URL: https://arxiv.org/abs/2203.03605, doi:10.48550/ARXIV.2203.03605.

Zhang, M., Hu, X., Zhao, L., Lv, Y., Luo, M., Pang, S., 2017. Learning dual multi-scale manifold ranking for semantic segmentation of high-resolution images. Remote Sensing 9. URL: https://www.mdpi.com/2072-4292/9/5/500, doi:10.3390/rs9050500.

Zhang, Y., Ling, H., Gao, J., Yin, K., Lafleche, J.F., Barriuso, A., Torralba, A., Fidler, S., 2021. Datasetgan: Efficient labeled data factory with minimal human effort, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10145–10155.

Zhang, Y., Qin, J., Park, D.S., Han, W., Chiu, C.C., Pang, R., Le, Q.V., Wu, Y., 2020. Pushing the limits of semi-supervised learning for automatic speech recognition. URL: https://arxiv.org/abs/2010.10504, doi:10.48550/ARXIV.2010.10504.

Zhao, S., Wu, B., Chu, W., Hu, Y., Cai, D., 2019. Correlation maximized structural similarity loss for semantic segmentation. CoRR abs/1910.08711. URL: http://arxiv.org/abs/1910.08711, arXiv:1910.08711.

Zhou, Y., Ren, Y., Xu, E., Liu, S., Zhou, L., 2022. Supervised semantic segmentation based on deep learning: a survey. Multimedia Tools and Applications URL: https://doi.org/10.1007/s11042-022-12842-y, doi:10.1007/s11042-022-12842-y.

Zlateski, A., Jaroensri, R., Sharma, P., Durand, F., 2018. On the importance of label quality for semantic segmentation, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE. URL: https://doi.org/10.1109/cvpr.2018.00160, doi:10.1109/cvpr.2018.00160.

Özdemir, , Sönmez, E.B., 2020. Weighted cross-entropy for unbalanced data with application on covid x-ray images, in: 2020 Innovations in Intelligent Systems and Applications Conference (ASYU), pp. 1–6. doi:10.1109/ASYU50717.2020.9259848.

Appendix

Test Sets

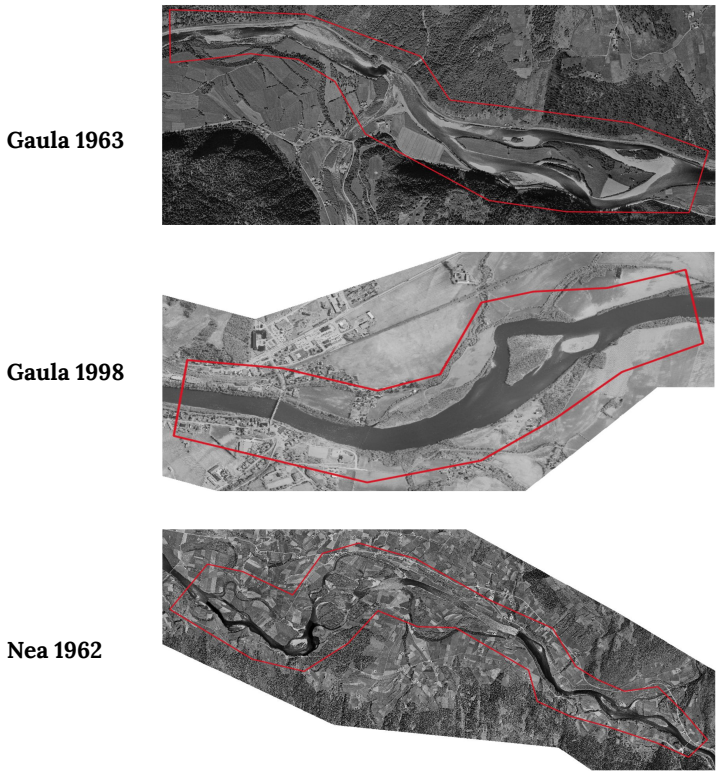


Figure 8.1: The test sets used for all the experiments. Only area inside the depicted boundery is considered to be the test set.

Models Disagreements

test sets. For each pixel in the test set images, all 5 different predictions were checked, and the number of different predictions assigned to that point is considered to be the number of disagreements. Hence, if for one point, all five predictions are the same, the number of disagreement is 1. If all the same except one of the 5 model, the disagreement is 2. Models are baseline model trained on dataset V1 and dataset V0 as described in the **E2** in subsection 5.2.3.

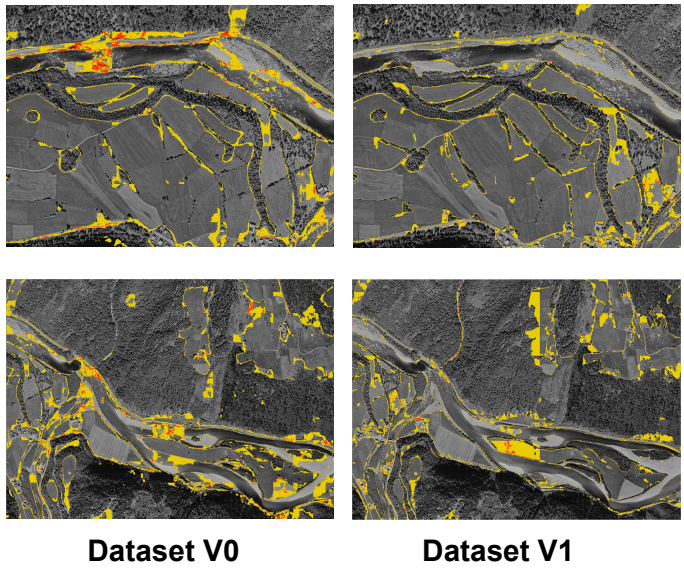


Figure 8.2: The disagreement of the five models on the Gaula 1963 test set. The disagreements are illustratd as *yellow*:2, *orange*:3, *brown*:4, *black*:5

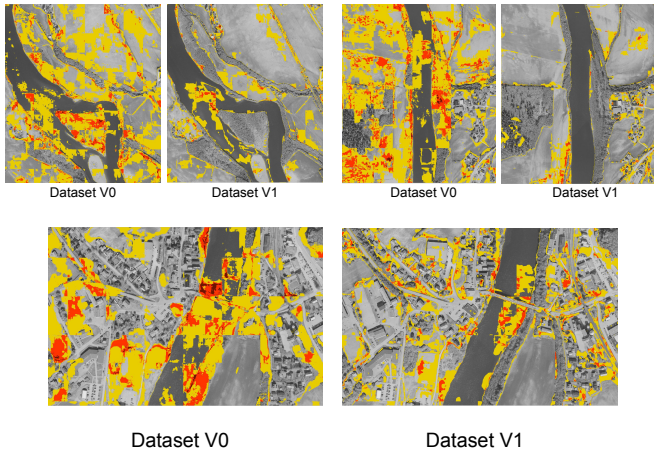


Figure 8.3: The disagreement of the five models on the Gaula 1998 test set. The disagreements are illustratd as *yellow*:2, *orange*:3, *brown*:4, *black*:5

MC Dropout Entropy

Models are baseline model trained on dataset V1 and dataset V0 as described in the **E2** in subsection 5.2.3.

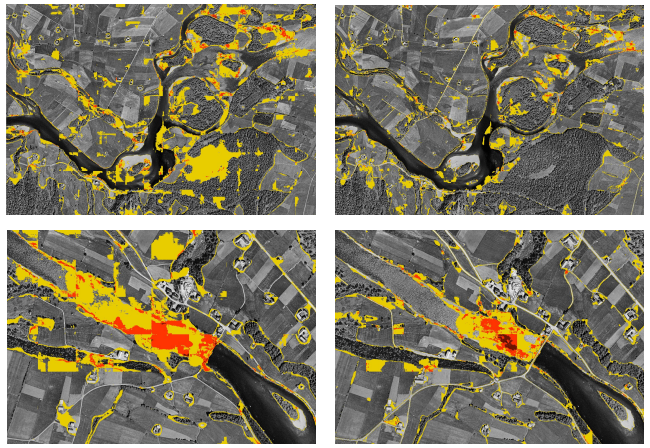


Figure 8.4: The disagreement of the five models on the Nea 1962 test set. The disagreements are illustratd as *yellow*:2, *orange*:3, *brown*:4, *black*:5

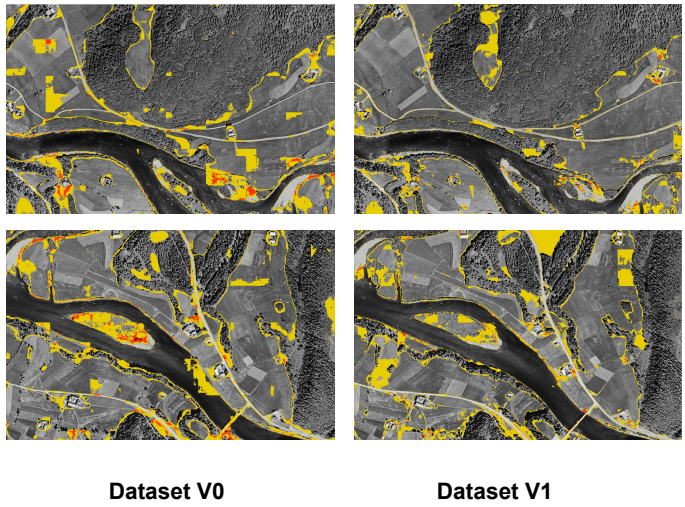


Figure 8.5: The disagreement of the five models on the Nea 1962 test set. The disagreements are illustratd as *yellow*:2, *orange*:3, *brown*:4, *black*:5

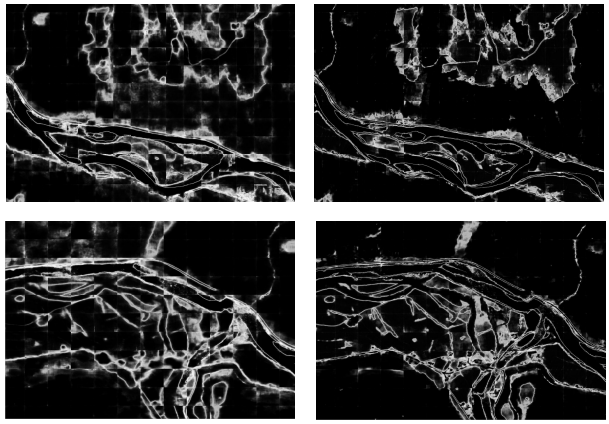


Figure 8.6: The MC dropout entropy of the models on Gaula 1963. Whiter areas are more uncertain.

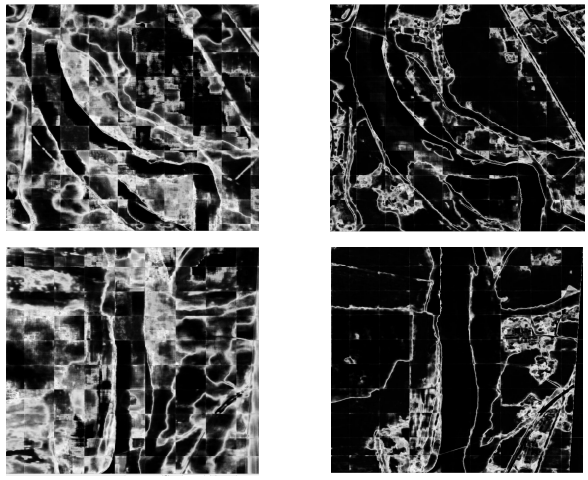


Figure 8.7: The MC dropout entropy of the models on Gaula 1998. Whiter areas are more uncertain.

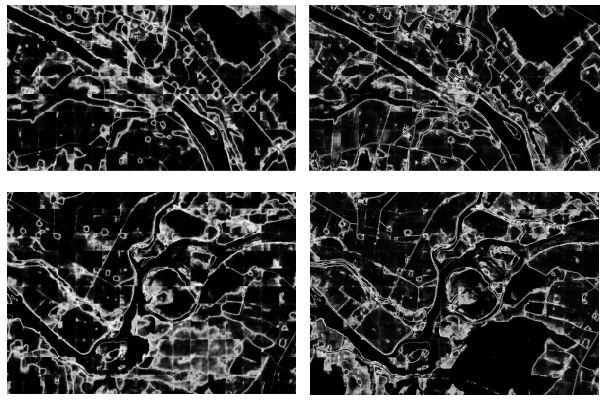
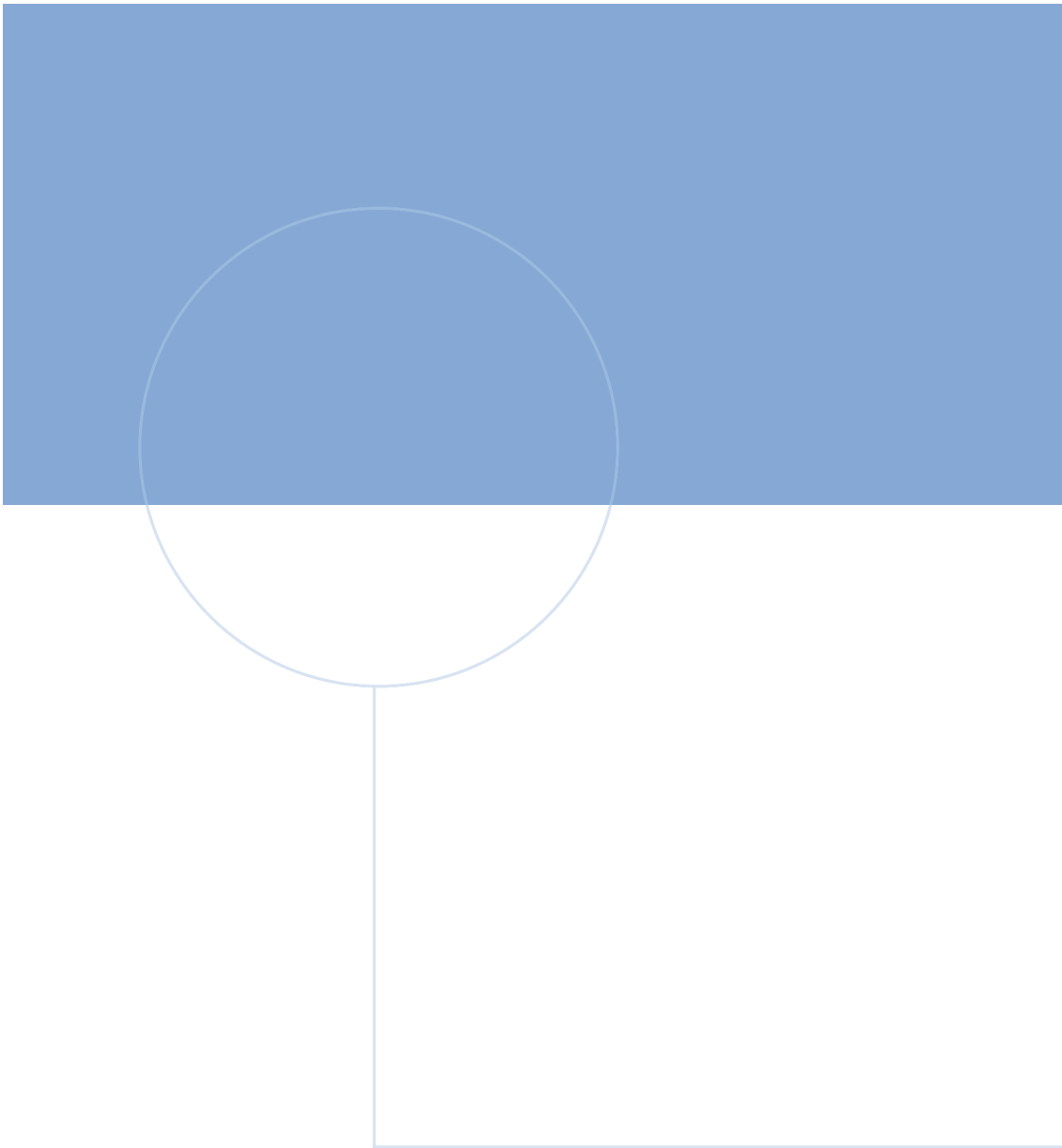


Figure 8.8: The MC dropout entropy of the models on Nea 1962. Whiter areas are more uncertain.



NTNU

Norwegian University of
Science and Technology