

Ingvild Rustad Haugen

# Computer Vision for Automatic Assessment of Fish Mortality Causes in Commercial Fish Farms

Master's thesis in Cybernetics and Robotics

Supervisor: Martin Føre

Co-supervisor: Torfinn Solvang

June 2023



Norwegian University of  
Science and Technology



Ingvild Rustad Haugen

# **Computer Vision for Automatic Assessment of Fish Mortality Causes in Commercial Fish Farms**

Master's thesis in Cybernetics and Robotics  
Supervisor: Martin Føre  
Co-supervisor: Torfinn Solvang  
June 2023

Norwegian University of Science and Technology







## MASTER THESIS

Name of the candidate: Ingvild Rustad Haugen

Discipline: Engineering Cybernetics

Project title (Norwegian): Maskinsyn for automatisk analyse og kategorisering av dødsårsak hos oppdrettslaks i kommersielle merder

Project Title (English): Computer vision for automatic assessment of fish mortality causes in commercial fish farms

### Background:

Fish welfare is a key element in modern fish farming. Acceptable welfare is crucial to ensure that the animals are provided with a good enough quality of life to ensure ethical production and is also closely linked with production efficiency and the quality of the end-product, as good welfare tends to result in better growth and more healthy fish. In 2018, Nofima in collaboration with other research institutes, published a report outlining a series of visual/morphological features that can be used to assess fish welfare, so-called Operational Welfare Indicators (OWIs). While these indicators are useful to evaluate for free-swimming fish, they can also be highly relevant to analyse for fish that have already died and been collected by the dead fish collection system as elevated mortality is the ultimate expression for suboptimal welfare.

ScaleAQ is a major provider of technological solutions for fish farming who each year delivers several camera systems for fish monitoring at commercial fish farms that primarily are used as tools for assessing fish appetite during feeding. However, there is a large potential for also using these cameras to analyse fish welfare and health during production. While monitoring of free-swimming fish is relevant to assess the immediate fish welfare in the cage, analysing dead fish to identify and classify potential mortality causes could be very valuable inputs to assess how management and operations impact the fish as critically impaired welfare is often ultimately expressed as increased mortality in commercial fish farms. Identifying if the fish died to due factors such as external wounds due to operation and handling, disease outbreaks, parasite infestations or other causes can therefore be crucial information for improving fish farm management and operations to ensure acceptable fish welfare.

Computer vision methods based on machine learning has emerged to become an increasingly popular tool for processing of video and image material. Such approaches have proven very efficient at automatically quantifying specific features in images and distinguishing between images describing different objects/situations. While the required training of computer vision methods (particularly deep learning-based approaches) tends to require more computational power than conventional analysis methods, GPU-based technologies (e.g., TensorRT, CUDA) are typically used to improve the performance and convergence speed, often in combination with cloud services with large GPU resources. A long-term ambition for ScaleAQ is to develop a self-sustained “photobox” system that will use computer vision/machine learning to automatically analyse dead fish collected from sea-cages. This will enable categorisation of cause of death for all individual fish, and development of statistics describing the entire population. Realising such a solution will require efforts on both

developing the hardware (e.g., optics, an embedded system for control and storage) and software. The first step in developing the software side is to identify and develop methods from computer vision/machine learning that are capable of identifying and categorising mortality causes in fish, and also have the potential of being run at embedded systems for future implementation in edge computing.

This project will focus on exploring the potential of using computer vision methods and machine learning to automatically detect and label winter ulcers which is a common mortality cause in fish from commercial fish farms.

The project will contain the following elements:

- Literature study
  - Welfare monitoring in aquaculture
  - Previous efforts on using Computer Vision(CV) and Machine Learning (ML) for disease analyses
- Data collection
  - Collect real data on dead fish from two different commercial sites
  - Label and segment the data manually for winter ulcers
  - Augment acquired dataset to build a larger and more robust dataset for training, validating and testing
- Implementation, training, and testing
  - Implement CV/ML methods deemed suitable for disease analyses of dead fish
  - Train the model using selection of augmented dataset acquired in the field
  - Validate and test using the remainder of the data
  - Evaluate implemented methods based on:
    - Common metrics for computer vision methods
    - Visual assessment of their ability to segment winter ulcers

Start date: January 09, 2023

Delivery date: June 06, 2023

Conducted at the Department of Engineering Cybernetics, NTNU

Main supervisor: Martin Føre

Co-supervisor(s): Torfinn Solvang (ScaleAQ), Torstein Kristensen (ScaleAQ)

Trondheim, February 23, 2023

Supervisor



## Preface

This thesis is my final work before finishing my Master of Science degree at the Department of Engineering Cybernetics at the Norwegian University of Science and Technology. The thesis is developed with supervision from Dr. Martin Føre from NTNU and Torfinn Solvang from ScaleAQ. I would like to thank my supervisors for giving feedback and assistance rapidly throughout the project. I would also like to thank the company that gave me the possibility to collect the data needed for this project and for all the assistance and clarifications I got from collaborating with your employees during the field trip. I would also thank Gunnar Aske for providing me with a GPU machine to be able to train the networks in this thesis with more computing resources.

I would like to express my gratitude to the staff and students of the Master's program at NTNU, particularly those on the 4th floor of Hovedbygningen, for creating a great environment to work in and a motivational social area. I would also like to thank my cohabitant, Louis Linnerud, for his unwavering support and invaluable assistance with GitHub, ensuring the safety and accessibility of my code. Additionally, I want to thank Camilla Dybdal, which with her expertise and attention to detail has given me advice throughout the project. Their contributions have been instrumental in the completion of this thesis, and I am truly grateful for their support.

I had no prior knowledge or experience in machine learning and computer vision. However, recognizing the potential of machine learning techniques in addressing the problems of this task posed a requirement to learn and explore the field of machine learning. This thesis hopefully proves the amount of knowledge I have gained working on the project for the past five months.

The code implemented in this project can be shared by contacting me at [ingvild.ruha@hotmail.com](mailto:ingvild.ruha@hotmail.com).



## **Abstract**

This master thesis contributes to the development of a photobox designed for computer vision analysis of deceased salmon in aquaculture. It addresses a gap in the existing literature by providing deeper insights into the analysis of deceased fish, an area that has received limited attention in previous research.

Data were collected from two locations in the coastal area of Trøndelag, and computer vision techniques were used to analyze the acquired images. The prevalence of winter ulcers in a significant portion of the deceased fish prompted the decision to focus on capturing photographs of these cases. Winter ulcers pose a well-known challenge in aquaculture, particularly during periods of low ocean temperatures. The data was annotated, resulting in a complete dataset tailored for segmenting winter ulcers. A model to segment the ulcers was implemented, trained and tested.

The thesis begins by presenting the theoretical background of aquaculture and a review of the computer vision methods used for disease analysis in aquaculture. Subsequently, the concepts and techniques relevant to the project are introduced. The methods employed and the implementation details of the developed code are justified and described for reproducibility. The experiments to find the best model are presented along with the results of the best model. Finally, an evaluation of the project's choices is conducted, along with a discussion on further steps necessary to create a functional photobox for computer vision analysis of fish.

The result was a model which segmented winter ulcers, performing an IoU of 74,24% and an F1 score of 81,60%. The thesis concludes that the methods used in this thesis are suitable to segment winter ulcers in Atlantic Salmon.

## Sammendrag

Denne masteroppgaven er utarbeidet med hensikt å bidra til å lage en fotoboks for analyse av dødfisk ved hjelp av datasyn fra oppdrettsanlegg for laks. Dagens forskning har i liten grad viet oppmerksomhet til dypere datasynsanalyse av dødfisk. Derfor tar denne oppgaven sikte på å fylle dette gapet i den eksisterende litteraturen.

Data ble samlet inn ved hjelp av feltarbeid på to ulike anlegg langs kysten i Trøndelag og bildene ble analysert ved hjelp av datasyn. Individene som ble fotografert bar preg av vintersår, som er et kjent problem i vinterhalvåret i oppdrettsnæringen, da bakteriene som forårsaker plagen trives best ved lave havtemperaturer. Prosjektet ble derfor rettet mot å segmentere vintersår på død laks for å innhente videre informasjon om sykdommen. Dataen ble annotert for å danne et komplett datasett for trening av en modell for segmentering av vintersår. Modellen ble implementert, trent og testet.

Oppgaven begynner med å presentere teorien bak oppdrettsnæring og datasynmetoder brukt for analyse av sykdom i oppdrettsnæring belyst. Deretter presenteres konsepter og teori som er relevante for prosjektet. Følgende blir metodene og implementasjonen av koden presentert og valg blir begrunnet. Deretter blir eksperimentene for å finne den beste modellen og resultatene presentert. Til slutt diskuteres valgene som ble tatt under prosjektet og hvordan det kan jobbes videre for å utvikle en fungerende fotoboks for dataanalyse av dødfisk.

De beste resultatene i oppgaven var en IoU score på 74,24% og en F1 score på 81,60% for segmentering av vintersår. Oppgaven konkluderer med at metodene som ble brukt er passende for å segmentere vintersår hos laks fra oppdrettsanlegg og kan fungere som en basis for videre utvikling av den fremtidige løsningen.

## Abbreviations

Abbreviation	Description
EUS	Epizootic Ulcerative Syndrome
mAP	mean Average Precision
GeLu	Gaussian Error Linear Unit
HSV	Hue, Saturation and Value
CNN	Convolutional Neural Network
SVM	Support Vector Machine
PCA	Principal Components Analysis
SGD	Stochastic Gradient Descent
ReLU	Rectified Linear Unit
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
IoU	Intersection over Union
FPS	Frames Per Second
CPU	Central processing unit
GPU	Graphics processing unit
HMI	Human Machine Interface
RoI	Region of Interest

# Contents

Preface . . . . .	iv
Abstract . . . . .	v
Sammendrag . . . . .	vi
Abbreviations . . . . .	vii
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation and Background . . . . .	1
1.1.1 Aquaculture and Welfare Monitoring . . . . .	1
1.1.2 Previous Efforts on Computer Vision for Disease Analyses . . . . .	2
1.2 Problem Description . . . . .	5
1.3 Objectives and Scope . . . . .	6
1.4 Outline . . . . .	6
<b>2 Theory</b>	<b>7</b>
2.1 Deep Learning . . . . .	8
2.1.1 Neural Networks . . . . .	8
2.1.2 Convolutional Neural Networks . . . . .	11
2.1.3 U-net . . . . .	12
2.2 Metrics in Computer Vision . . . . .	14
2.2.1 Accuracy . . . . .	14
2.2.2 Recall . . . . .	15
2.2.3 Precision . . . . .	15
2.2.4 F1-score . . . . .	15
2.2.5 Intersection over Union . . . . .	16
2.2.6 Frames per Second . . . . .	16
<b>3 Methods and Implementation</b>	<b>17</b>
3.1 Technical Framework for Implementation . . . . .	17
3.1.1 Hardware . . . . .	17
3.1.2 Software . . . . .	17
3.1.3 Training With GPU . . . . .	18
3.2 Field Data Collection . . . . .	18
3.2.1 Image Acquisition . . . . .	18
3.2.2 Camera Settings . . . . .	20
3.2.3 The Raw Data Set from Data Acquisition . . . . .	21

3.3	Data Preparation to Segment Winter Ulcers . . . . .	22
3.3.1	Data Annotation of Winter Ulcers . . . . .	23
3.3.2	Creating Binary Masks for the Winter Ulcers . . . . .	24
3.3.3	Splitting the Dataset into Training, Testing and Validation Datasets . . . . .	25
3.3.4	Data Augmentation for the Winter Ulcer Dataset . . . . .	25
3.4	Preparing Data to Segment Fish Bodies . . . . .	27
3.4.1	Creating the Binary Masks for the Fish Bodies . . . . .	28
3.4.2	Creating Matching Datasets for the Two Models . . . . .	29
3.4.3	Data Augmentation for the Fish Body Dataset . . . . .	29
3.5	Neural Network Design . . . . .	29
3.5.1	U-net . . . . .	29
3.6	Combining the Two Models in Series . . . . .	31
<b>4</b>	<b>Training and Results</b>	<b>34</b>
4.1	Training U-net . . . . .	34
4.2	Experiments . . . . .	36
4.3	Metric Results From the Resulting Ulcer Model . . . . .	38
4.4	Results Visualization . . . . .	39
4.5	Results After Moving the Background . . . . .	41
4.6	The Ulcer Factor . . . . .	42
<b>5</b>	<b>Discussion and Further Work</b>	<b>44</b>
5.1	Discussion and Evaluation of the Model Performance . . . . .	44
5.2	The use of Neural Networks to Segment Winter Ulcers and the Choices for the Final Solution . . . . .	45
5.3	The Data Acquisition and the Dataset Used . . . . .	46
5.4	The Photobox as a Concept . . . . .	48
5.5	Software Transferability to the Photobox . . . . .	49
<b>6</b>	<b>Conclusion</b>	<b>51</b>
	<b>References</b>	<b>52</b>

# Chapter 1

## Introduction

The introduction chapter of this thesis begins by providing the motivation and background for the research problem under investigation. This includes a comprehensive overview of the context and significance of the topic, highlighting the limitations of existing research. Following the motivation and background, a clear problem description is presented, leading to the formulation of specific objectives for the study.

### 1.1 Motivation and Background

#### 1.1.1 Aquaculture and Welfare Monitoring

Fish farming is the practice of raising fish and comprises all life stages of the fish from brood-stock/eggs to fully grown adults [1]. While some species are raised in tanks until they have a marketable size, the Atlantic Salmon is moved to outdoor sea cages for the final growing phase. The volume of water required by a fish is dependent on its size, which makes it more convenient to meet the needs of fully grown fish in a sea cage. Fish farming is regarded as a pivotal approach in fostering the advancement of a more sustainable food production system and addressing the mounting global population's nutritional needs [2].

Salmon aquaculture was first developed in Norway during the early 1970s and has subsequently undergone significant global expansion [3]. The health benefits of seafood are increasingly being promoted by global health authorities, and aquaculture has expanded fish availability to regions and countries with limited or no access to cultured species [4]. At a global level, about 80% of the worldwide salmon harvest is farmed. Atlantic Salmon is relatively small in harvest volume when compared to other species of fish, but it is a very visible product in many markets due to a high level of industrialization.

According to a report from The Norwegian Marine Research Institute, Norway exported 1.26 million of tons farmed salmon for a value of 105.8 billion NOK in 2022, making salmon the biggest contributor to seafood export [5]. Norway's aquaculture industry represents the largest farmed production in the country, with over 500 million farmed fish located along the Norwegian coast. However, this large-scale production presents significant challenges concerning animal welfare.

In the realm of aquaculture, the well-being of the aquatic animals being raised is of paramount importance [6]. As these animals are living beings, there exists a moral obligation to ensure their welfare is upheld. This was the motivation behind the FISHWELL project, which has established a definition of fish welfare as the overall quality of life as perceived from the perspective of the fish themselves [6]. Ensuring fish welfare is not only of ethical interest but also benefits the producer economically due to higher quality and decreased mortality.

According to a report from 2022, the fish who dies in the fish cages constitutes the biggest loss in fish production of salmon and trout [7]. The Norwegian Veterinary Institute has identified three health challenges that contribute to significant mortality rates: injuries resulting from delousing operations, complex gill disease, and winter ulcers [8]. Particularly, the extent of winter ulcers poses challenges in terms of animal welfare. In 2022, 56,7 million salmon died in the sea cages, which represents 90 % of the loss in the production of Atlantic Salmon. The obligation for companies to solely report mortality numbers rather than mortality reasons render the precise data regarding diseases unattainable.

To optimize fish production and maximize resource utilization, it is necessary to implement monitoring and control measures for fish in cages. Currently, manual monitoring is conducted through the collection and examination of deceased fish from the cage floor. Automation of this process through the incorporation of cameras and machine learning techniques has the potential to greatly enhance and improve this aspect of fish production. The implementation of automated classification and analysis of deceased fish in a cage via machine learning techniques has the potential to greatly enhance disease and damage detection, allowing for earlier preventative measures to be taken. This can help to minimize the spread of disease and suffering among the fish population. Additionally, the reduction of loss from fish production that results from this improved monitoring can lead to increased revenue and better quality of the fish produced in fish farms.

### **1.1.2 Previous Efforts on Computer Vision for Disease Analyses**

Computer vision technology is an excellent and practical inspection technology that can contribute to more efficient solutions to industrial challenges [9]. Due to the rapid development of cameras and hardware, one has been able to use computer vision for an increasing volume of applications in the previous years. The interest in aquaculture combined with the search for sustainable food production has resulted in a considerable amount of related research motivated by the reduction of mortality and loss.

Fish farming practices worldwide exhibit variations in terms of species composition and the challenges encountered, leading to numerous studies and publications that can be partially transferable across contexts. However, it is important to note that certain cases are highly specific to particular species and diseases, necessitating tailored approaches and interventions.

A relevant research to this thesis was published by Ahmed et al. (2020) [10] who used feature extraction with statistical features and gray-level co-occurrence matrix features and a Super Vector Machine (SVM) to distinguish between healthy and infected fish. An SVM is a supervised classifier machine

learning model [11]. The research was motivated by the recognition of the significant impact that various diseases pose to both wild and farmed populations of fish and the objective was to propose a framework for fish disease detection based on an SVM. The dataset utilized in the study comprised salmon species, and while there was limited specification regarding the infected fish, the visual representations depicted in the paper closely resembled characteristics associated with skin diseases. In the process of developing the classifier, they used k-means clustering to separate the infected areas from the fish images. The features from these segmentations were then fed into the SVM. The classifier proposed achieved an accuracy of 94.12 % for their dataset with augmentation, which is considered good performance for the binary classification problem.

Automatic detection and classification of skin diseases appear to garner widespread attention due to their broad relevance and visible nature. Malik et al. (2017) [12] developed an algorithm to classify images of fish with and without Epizootic Ulcerative Syndrome (EUS). EUS is a skin disease occurring in regions far from Norway [13], but as mentioned earlier, the methods and techniques can be transferred to other cases. Like the research described in the previous section, the objective was to classify infected and non-infected fish. The research explored various combinations of feature extractors and processing techniques to find the best combination for classification. The combination of FAST-Principal-Component-Analysis (FAST-PCA) as a feature extractor and neural network gave an accuracy of 96 % for the binary classification.

Another disease classification study conducted by Waleed et al. [14] investigate different convolutional neural networks (CNNs) to classify three diseases in fish farms, respectively EUS, Ichthyophthirius (Ich) and Columnaris. The proposed system in the research is to utilize a combination of multiple sensors and cameras, integrated with a Raspberry Pi platform, to classify diseases. Upon detection, the system will communicate the results to the user through a mobile phone notification mechanism. The data used in the research was extracted from videos acquired by the authors. The study compares the performance of the CNN architectures ResNet18, ResNet50, ResNet101 and AlexNet on the dataset in different color spaces. The classification results made the authors conclude that Alexnet was the best deep neural network with a test accuracy of 99.04% for classifying the three diseases in the XYZ color space.

The classification of skin diseases is a popular research problem that has received attention in research. The detection of these infections may able the farmers to prevent further outbreaks and has motivated the research conducted by Gangyi et. al (2023) [15]. They used YOLOv4 in combination with MobileNetv3 as the feature extraction network and the GeLU (Gaussian error Linear Unit) activation function to detect and classify four different skin diseases. They achieved a mean Average Precision (mAP) of 99.64 %.

Although many researchers have used deep learning techniques to detect and classify skin diseases, Chakravorty et al. (2015) [16] explored how diseases can be identified by the use of image processing techniques. They were motivated by reducing mortality caused by EUS and gathered data from decreased fish infected with this skin disease. The research applies various image processing techniques like Principal Component Analysis (PCA), K-means Clustering, HSV and Morphological oper-



ations to measure the number of pixels in diseased areas. Despite being conducted eight years ago, this research showcased the possibility of simple segmentation techniques in addressing the problem. However, they conclude that more advanced techniques like pattern recognition by the use of neural networks or support vector machines may be advantageous.

Various papers of research have been conducted to classify skin diseases with the motivation of reducing losses in fish farming. Given the visible nature of skin diseases, using computer vision techniques becomes advantageous for the accurate identification and classification of such ailments. While certain studies have proposed comprehensive systems encompassing identification and notification functionalities for farmers, others have focused solely on the classification aspect. However, it can be generally observed that the systems proposed thus far primarily aim to classify the live fish within their respective aquaculture cages. The majority of studies focus on classification, separating fish by their diseases. There has been unexploited potential in the insights that can be derived from the analysis of dead fish and how meaningful information from this can conclude on fish health and prevent diseases.

A company that has attempted to exploit this is the Norwegian company Fishwell Technology AS which released a commercial product called GOODeye Detection which they claim can count dead fish in containers with a handheld tablet [17]. The diagnostics part of the system is said to be under development to deliver increased knowledge about the mortality reasons of fish from sea cages.

The existing studies primarily emphasize disease classification based on the presence or absence of diseases in images, focusing on classification or detection. However, the segmentation of diseased areas has received limited attention, and the study by Chakravorty et al. (2015) [16] does not utilize deep learning for this purpose. Consequently, there is a scarcity of research aimed at effectively segmenting infected areas and extracting additional information about the infections. The utilization of segmentation techniques holds promise in providing valuable insights into the localization, shape, and size of skin diseases.

The field of image segmentation has been of big interest in a broad field and has been applied in for example medical image analysis, autonomous vehicles and augmented reality [18]. One of the first significant techniques for image segmentation is thresholding, dividing pixels into groups by their values. Long et al. proposed in 2015 the Fully Convolutional Networks [19], which was a milestone in deep learning-based semantic image segmentation models. Since then, multiple deep-learning-based models for segmentation have been developed including U-net, Mask R-CNN and Deeplab.

Segmentation has not been frequently used in previous research of aquaculture, but a method article from Sintef by Kvæstad, Hansen and Davies (2021) describe how they used Mask R-CNN to segment morphometrics on microscopy images of cod larvae [20]. The researchers were able to accurately and efficiently analyze large amounts of microscopy images of cod larvae, providing valuable insights into the development and growth of these fish. The use of Mask R-CNN in combination with classical machine vision techniques achieved a significant improvement in accuracy and precision compared to previous methods, and the ability to automate the morphometric analysis of Atlantic Cod Larvae

has the potential to greatly enhance research in this field. Overall, this study is a noteworthy contribution to the advancement of computer vision and its applications in marine biology research. This study exemplifies the utilization of image segmentation to enhance comprehension of objects within images, with potential applicability to skin diseases.

## 1.2 Problem Description

As stated in Subsection 1.1.1, there is both ethical and economic motivation behind the investigation of diseases and mortality reasons in farmed salmon. The previous attempts at using computer vision presented in Subsection 1.1.2 emphasize the untapped possibilities for the analysis of deceased fish and introduce new avenues of research. However, there remains a distinct need to focus specifically on the segmentation of skin disease in deceased salmon. While previous attempts have explored classification methods, a deeper analysis of deceased fish has received comparatively limited attention in the existing literature. Hence, this research addresses this gap by emphasizing the significance of segmentation techniques for winter ulcers on deceased salmon.

ScaleAQ is an international provider of technological solutions for fish farming. The company *provide innovation, technology and equipment to customers globally* [21]. A long-term ambition for ScaleAQ is to develop a self-sustained photobox system that can analyze dead fish collected from sea cages with the help of computer vision. The idea of the desired solution with the photobox is to get more data and thereby learn from the dead fish analysis so that preventive measures can be taken in the future and prevent economic loss and animal suffering. This thesis aims to contribute to automating and digitizing the work of measuring disease and damage to fish in fish farming. A sketch of the photobox is shown in Figure 1.1.

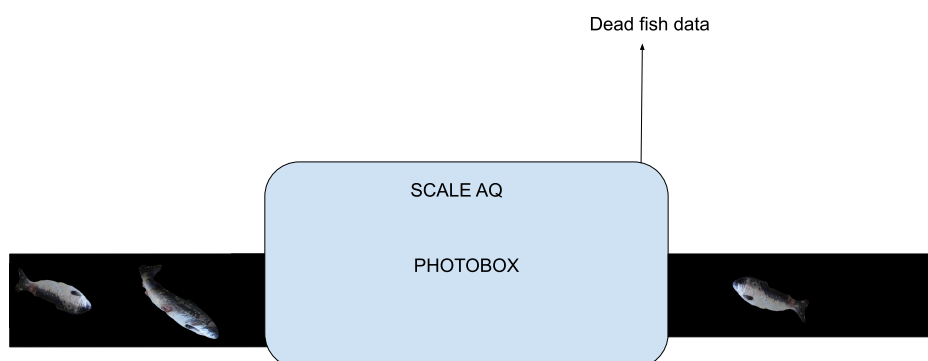


Figure 1.1: A sketch of the desired system.

### 1.3 Objectives and Scope

The objective of this thesis is to develop a program that will contribute to the development of a photobox for disease inspection for ScaleAQ and thereby contribute to fill the gap in previous research papers. The project description can be found in the ???. The thesis has a specific scope that concentrates on the software aspect of the photobox, with no consideration given to the hardware components. Consequently, the solution excludes any computational constraints that may arise during the actual implementation of the final product.

The process to create a program for the ScaleAQ photobox include data acquisition, image preprocessing, and the development of a neural network for analysis. The main objectives can be summarized as:

- Identify methods for automatic winter ulcer segmentation in salmon
- Collect and annotate real data from a fish farm
- Train and test a model for the automatic segmentation of winter ulcers

### 1.4 Outline

This report comprises six chapters that are structured as follows. Chapter 2 serves as an introduction to the concepts, presenting the theory of deep learning and identifying methods for image analysis and segmentation of winter ulcers. Chapter 3 focuses on the data acquisition process during the field trip and demonstrates how the techniques discussed in Chapter 2 were applied to implement the solution. Additionally, this chapter justifies the choices made during the implementation. Chapter 4 presents the experiments conducted to determine the optimal neural network with the highest performance. The results from the best model and the visualization of these results are showcased in Chapter 4, additionally. Chapter 5 presents a comprehensive discussion of the obtained results in this thesis and the methods used in the project. Furthermore, recommendations for further research and potential modifications are provided. Finally, the conclusion of the thesis can be found in Chapter 6.

## Chapter 2

# Theory

The chapter aims to provide the reader with a comprehensive understanding of the available known methods for the automatic inspection of images and focuses on the segmentation of winter ulcers in Atlantic Salmon, addressing the first of the three objectives described in Section 1.3. Different approaches to inspect diseases have been utilized in the literature presented in Subsection 1.1.2 and both deep learning techniques and basic image processing techniques have been attempted. However, deep learning techniques have been chosen to create a solution to this research problem as they have demonstrated good performance in image analysis tasks in [10], [12], [14] and [15]. In addition, the conclusion of [16] recommends using classical pattern recognition algorithms to improve the performance of disease detection, claiming that image processing is not sufficient. Recent methods for image segmentation propose deep learning models to solve the segmentation problem and have achieved notable performance [22]. Furthermore, considering that this research marks the initial phase in the development of the ScaleAQ photobox, employing a deep learning technique offers the potential for continuous improvement and adaptability as the project progresses. By selecting deep learning as the approach, this research aims to capitalize on its proven effectiveness in image analysis, aligning with the conclusions and recommendations of prior studies, and leveraging its potential for ongoing development and refinement within the context of the ScaleAQ photobox project.

The objective of this thesis is to address the existing research gap by focusing on the segmentation of winter ulcers, a skin disease prevalent in aquaculture-reared Atlantic Salmon. The literature review has identified an absence in the segmentation of winter ulcers, and by conducting this segmentation, valuable information regarding deceased salmon can be extracted. Additionally, the segmentation process can also serve as a means of solving the binary classification problem, as it allows for the determination of the presence or absence of the disease.

To address the research objectives outlined in this thesis and bridge the existing research gap, the acquisition of data from infected fish using a camera is crucial. The process of distinguishing between infected and non-infected fish necessitates the extraction of pertinent features from the captured images, followed by training a model to discern the distinctive patterns present in the infected fish. To do pixel-wise segmentation of the ulcers, the features need to be fed into a binary classifier to classify if the pixel belongs to the ulcer or not. The theory behind the components used in the implemented solution will be presented in the following sections.

## 2.1 Deep Learning

Deep learning is a subset of the field of machine learning which has proved to provide advanced analytical tools for understanding enormous amounts of data [23]. As deep learning involves training neural networks to learn patterns and data representations, this is a suitable method to make a computer process images and videos. Convolutional neural networks have gained significant popularity as a technique employed for the processing of video or images to extract information. Section 2.1.1 will present the theory of neural networks and how they learn and Subsection 2.1.2 will present the theory behind convolutional neural networks.

### 2.1.1 Neural Networks

Neural networks are a class of machine learning algorithms that use multiple layers to learn and make predictions from data. These networks are a set of algorithms inspired by the neurons in the human brain, designed to recognize patterns [24]. In the context of supervised learning, a neural network can be conceptualized as a black box that takes in data along with their corresponding true labels and then computes the relationship between them. This can be seen in Figure 2.1. The goal is that by updating the models' internal states, it can use these internal states to predict data when presented with data it has never seen before.

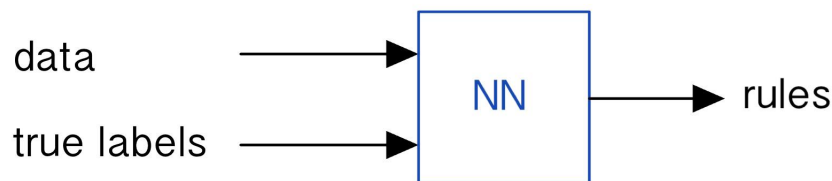


Figure 2.1: Neural network as a black box.

The inputs to a neural network is a set of data points of patterns represented as feature vectors. In image recognition tasks, the input can be a matrix of pixel values [25]. Computations takes place in the nodes, which can take multiple inputs and give a single output [26]. A neural network node is seen in Figure 2.2 and the functions will be explained below.

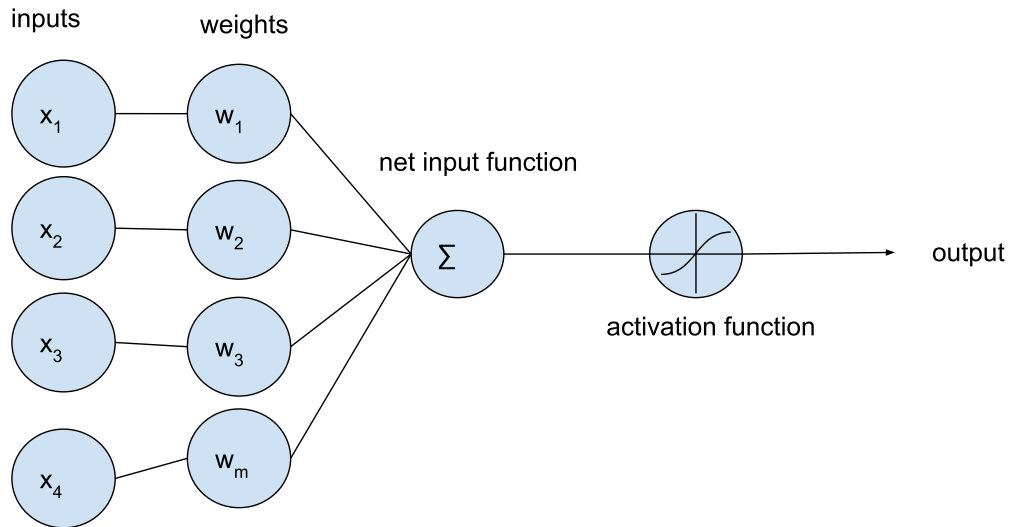


Figure 2.2: A diagram of a node in neural networks.

The inputs,  $[x_1, x_2, x_3, \dots, x_m]$ , will in the case of image recognition be pixel values, and the weights,  $[w_1, w_2, w_3, \dots, w_m]$ , determine the significance of the inputs. The net input function gives the weighted sum of the inputs:

$$\Sigma = (x_1 \times w_1) + (x_2 \times w_2) + \dots + (x_m \times w_m) + b \quad (2.1)$$

The  $b$  in Equation 2.1 denotes the bias, which can be seen as an offset and is necessary to move the entire activation function horizontally to generate the required outputs [27]. The activation function scales the output and depends on the design of the network. Activation functions prevent linearity and enable the model to solve complex problems [28]. A popular choice is the nonlinear Rectified Linear Unit, ReLu [29].

A neural network consists of layers of nodes processing inputs and outputs. The initial input layer received the data and the output layer gives the prediction. The output on one node is the input to a node in the next layer. This builds the neural network as seen in Figure 2.3.

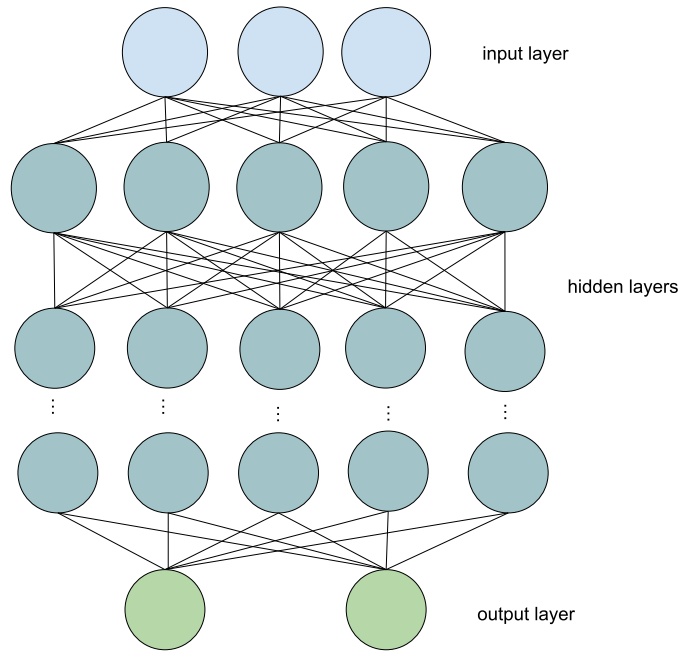


Figure 2.3: A deep NN.

The objective of the training is to find the optimal weights and biases, that is the network parameters. For a neural network to find these parameters, a cost function is used to get an estimation of how far from the desired solution the model is. The cost function is dependent on what type of problem the neural network solves [27]. An example of a cost function is the mean squared error (MSE), which is the average of the error between the predicted labels and the true labels,

$$C = MSE = \frac{1}{m} \sum_{i=1}^m (y_i - \hat{y}_i)^2 \quad (2.2)$$

where  $y_i$  denotes the true values,  $\hat{y}_i$  denotes the predicted values and  $m$  denotes the number of samples. It is desired to minimize the cost function, as this yields a minimization of the error. The process for minimizing the error by updating weights and biases is called back-propagation [29]. This is done by taking the partial derivative with respect to the weights and biases, which gives the gradient of the cost function. To estimate the optimal weights and biases, an optimization algorithm is used. This algorithm is to be chosen by the designer of the neural network, popular choices are Gradient Descent and Stochastic Gradient Descent [30]. To choose the step for the change of the weights and biases to minimize the cost, one can set a learning rate,  $\alpha$ . The updates of the weights and biases, when using Gradient Descent, are given as follows [27]:

$$\begin{aligned} w_i &= w_i - \left( \alpha \times \frac{\partial C}{\partial w_i} \right) \\ b &= b - \left( \alpha \times \frac{\partial C}{\partial b} \right) \end{aligned} \quad (2.3)$$

The objective when training a neural network is to minimize the generalization error, which reflects the network's ability to make accurate predictions for new, unseen data that were not encountered during training[31]. Overfitting is a problem that prevents the model to generalize, which means it

fits too well to the training data [32]. There are several techniques to avoid overfitting the training data including early stopping and batch normalization which will be used in this thesis. Batch normalization is a technique used during the training of deep neural networks to accelerate the training process and enhance the network's performance [33]. The objective is to avoid the phenomena of internal covariance shift, referring to the change in the distribution of each layer's inputs during training.

### 2.1.2 Convolutional Neural Networks

Convolutional neural networks are algorithms that take an image as input and learn the features and characteristics of the image. The CNNs have three main layers; a convolutional layer, a pooling layer and a fully connected (FC) layer [34].

The convolutional layer takes an image as input and performs multiple convolutions on the image. The objective is to extract patterns and information from the image. The convolutional layers utilize kernels to convolve the image and extract features [35]. The deeper the convolutional layer is in the network, the more advanced pattern it learns as each convolutional kernel specializes in learning different features. An example of one step of convolution can be found in Figure 2.4. The kernel acts as a filter and can be seen as a small-size matrix [36]. The filter slides across the image pixels and determines specific features from the image, depending on the kernel design. Figure 2.4 shows the kernel applied to the first 3x3 pixels of the image. The convolved feature is the result of the filter's convolution for this first step. The next step will involve a convolution with the 3x3 next pixels and the pixel step size for the filter is to be set as the stride. Adding multiple convolutional layers in a NN allows the network to learn different features.

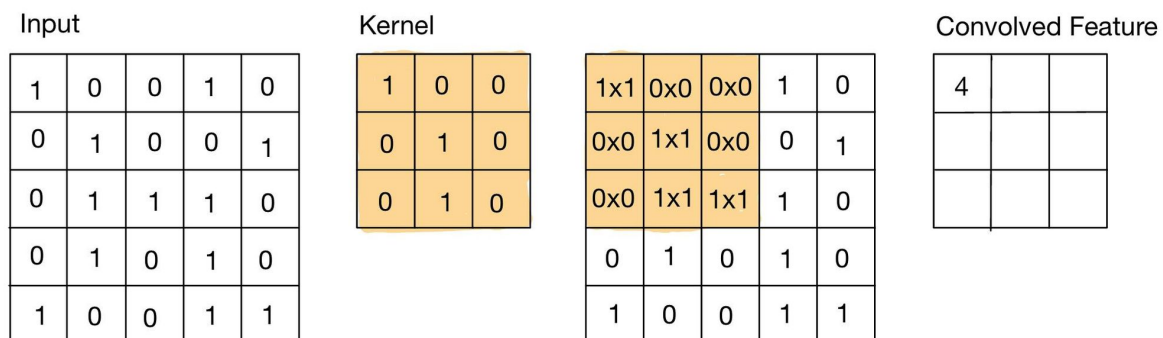


Figure 2.4: Convolution.

The pooling layer is responsible for performing pooling operations to the image, to mainly reduce computational cost and make the network more generic [35]. The pooling layer reduces the spatial dimensions, that is the width and height of the input. The operation leads to a loss of information but, however, the loss is beneficial to the network as the computational cost decrease. The pooling operations extract the most important features in the data. Figure 2.5 demonstrates a visual example of max pooling. The spatial dimension is reduced from 4x4 pixels to 2x2, and the maximum value of the pixels is kept.



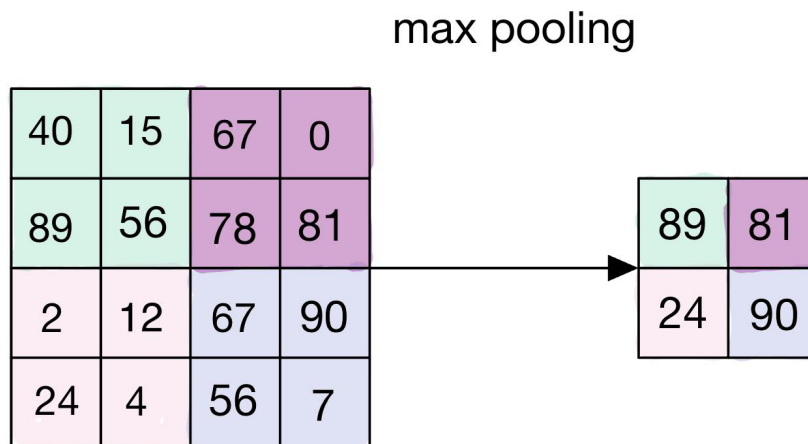


Figure 2.5: Max pooling.

The convolutional layers and the pooling layers can be repeated multiple times before the final fully-connected layer [35].

The fully-connected layers take in a one-dimensional flattened image vector and find a probability score for each of the labels.

Convolutional neural networks have been applied to a variety of computer vision problems including image classification, object detection, semantic segmentation or instance segmentation [37]. Given the objective of segmenting winter ulcers in a dataset of infected salmon, CNNs have demonstrated their potential and efficiency for addressing this specific problem. Semantic segmentation aims to segment an unknown image into different parts [38]. As the project aims to segment winter ulcers, the pixels will be assigned to belong to the ulcer or the background, hence the problem to solve is a binary segmentation problem. Notably, numerous studies have showcased remarkable accomplishments in semantic segmentation by leveraging convolutional neural network architectures such as U-net and its variants [37], [39], [40]. Therefore, the subsequent section presents the theoretical foundations of U-net to provide a comprehensive understanding of its principles and techniques.

### 2.1.3 U-net

U-net is a deep neural network designed for image segmentation [41]. The release of U-net provided an 11% improvement in accuracy over the second best approach, contributing to a significant improvement in the problem of image segmentation [42]. The architecture of the net makes it belong to the encoder-decoder-based models as the net consists of an encoding stage and a decoding stage, denoting a U. The U-net architecture is presented in Figure 2.6. The net is symmetric, where the first contracting path captures context, classifying the objects and the expanding path enables precise localization [22].

The encoder aims to determine the type of objects in the image and their localization [42], depicted in the left part in Figure 2.6. The encoder extract features from the image, creating compact representations of the input image in a lower dimension. The extraction is done using convolutional layers and pooling layers. The convolutional layer is a kernel passing through each pixel of the image and learning the features. Each encoder block exists of two 3x3 convolutions, seen as the blue arrows in Figure 2.6. Each convolution is followed by a ReLU activation function. The ReLU makes the network non-linear, which contributes to the generalization of the training data. The output of the ReLU acts as a skip connection for the corresponding decoder block, illustrated by the gray arrows in Figure 2.6. The following layers are pooling layers which reduce the dimensions of the output. U-net has a 2x2 max-pooling where the spatial dimensions are reduced, leading to a reduction of the computational cost due to the reduction of trainable parameters. The max pooling is denoted by the red arrows in Figure 2.6. Combining these layers able for the extraction of detailed information. During training, the network learns what features are important for semantic segmentation and extract these to create a compact representation of the image.

The skip connections between the encoder and decoder layers, illustrated by gray arrows, aid in improving the flow of gradients during backpropagation [42]. Due to the symmetry of the U-net, these opposite layers have the same dimensions. The skip connections connect the encoder and decoder at the corresponding convolution and deconvolution blocks. This allows the decoder to combine high-level features from the encoder with the low-level details that were lost during downsampling.

The decoder reconstructs the image from the compact representation output of the encoder [42]. It contains deconvolutional layers to increase the dimension of the image. Every block in the decoder starts with a 2x2 deconvolution, seen as green arrows in Figure 2.6, which is concatenated with the corresponding convolution layer in the encoder. Next follows two 3x3 convolutions followed by the ReLU activation function. The output of the last decoder passes the image through a 1x1 convolution with sigmoid activation. The sigmoid activation creates the segmentation mask, representing the pixel-wise classification. The classification appears as probabilities for each pixel to belong to the positive class.

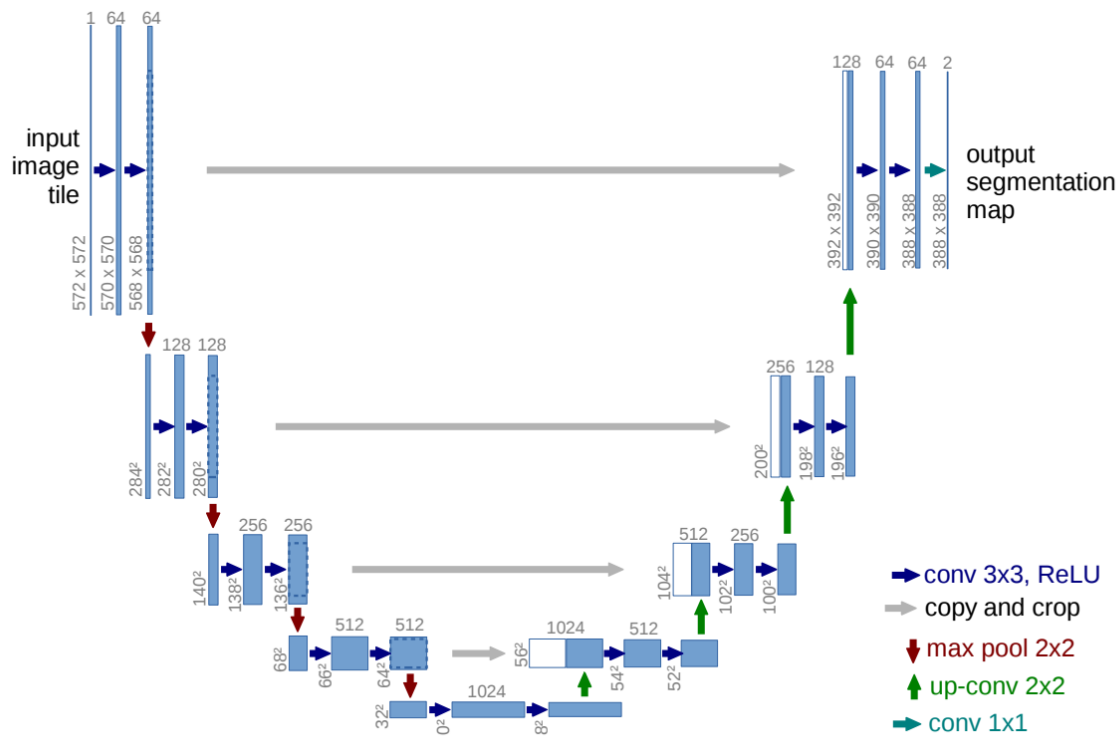


Figure 2.6: U-net as illustrated in the original paper [39]. Reproduced with permission.

U-net was made to perform biomedical image segmentation. One of the main advantages of U-net is the reliance on data augmentation to use the available dataset more efficiently [39]. The network has been applied to various biomedical segmentation problems.

## 2.2 Metrics in Computer Vision

The evaluation of a model is crucial in the search for the best solution for image segmentation of winter ulcers. This section will present the metrics used in the project for measurements of performance in the developed model. Several metrics for performance measurements in machine learning exist but some are more relevant than others, dependent on the specific case. Since this thesis is centered around image segmentation, it is crucial to consider a comprehensive set of metrics that collectively assess the performance of the segmentation. Instead of relying solely on a single metric, evaluating the segmentation results using multiple measures allows for a more holistic and thorough analysis of the overall segmentation quality. The metrics to be presented will be the ones used for the binary segmentation task done in this thesis.

### 2.2.1 Accuracy

Pixel accuracy is a widely used metric to evaluate the accuracy of image classification tasks by measuring the percentage of correctly classified pixels in an image. However, this metric may not be a suitable measure for binary segmentation tasks due to the presence of imbalanced class distribution in the image. In such scenarios, a high pixel accuracy score may be achieved even if the segmentation output assigns all the pixels to the background class, rendering this metric inadequate for assessing

the performance of binary segmentation models [43]. A more suitable metric for image segmentation is balanced accuracy. Balanced accuracy is a metric used when the problem has a class imbalance and the formula is given by [44]

$$\text{Balanced Accuracy} = \frac{\text{Sensitivity} + \text{Specificity}}{2} \quad (2.4)$$

where

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (2.5)$$

and

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad (2.6)$$

### 2.2.2 Recall

Recall is a metric given by

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (2.7)$$

which can be seen as the classifier's ability to find all the positive samples [44]. In the context of image segmentation of winter ulcers, the true positive predictions refer to the pixels that belong to the ulcers. The false negatives belong to the pixels that were predicted to not belong to the winter ulcers, but truly belonged.

### 2.2.3 Precision

The precision is the ratio given by

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (2.8)$$

and can be seen as the ability of the classifier to accurately identify positive samples at the pixel level. It quantifies the number of true positive predicted pixels overall positively predicted pixels [44].

### 2.2.4 F1-score

The F1 score is a common performance metric used to evaluate the accuracy of binary classification models [43]. The F1 score is given by

$$\text{F1} = \frac{2 * |A \cap B|}{|A| + |B|} \quad (2.9)$$

where A and B denote the ground truth segment and the predicted segment [43]. The F1 score is a good evaluation for binary segmentation tasks as it can be seen as a measure of overlap between the ground truth and the predicted class. As this thesis will solve a binary segmentation problem, the F1 score will be the same as the Dice coefficient. For this thesis, the dice coefficient metric is used in the loss function, which will be later explained and justified. The dice loss is given similar to the F1 score, but with a 1 added to the denominator and numerator so that the function is defined when  $A = B = 0$  [45].

$$\text{DL}(A,B) = 1 - \frac{2 * |A \cap B| + 1}{|A| + |B| + 1} \quad (2.10)$$

### 2.2.5 Intersection over Union

A good evaluation for binary segmentation tasks is the Intersection over Union (IoU), which is given by the formula

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \quad (2.11)$$

where  $A$  and  $B$  denote the ground truth segment and the predicted segment. This metric is considered as a good measurement for semantic segmentation tasks where the classes are imbalanced distribution in the images. The IoU determines the area of overlap between the predicted segmentation and the ground truth, divided by the union between these [43]. A visual representation of IoU is to be found in Figure 2.7.

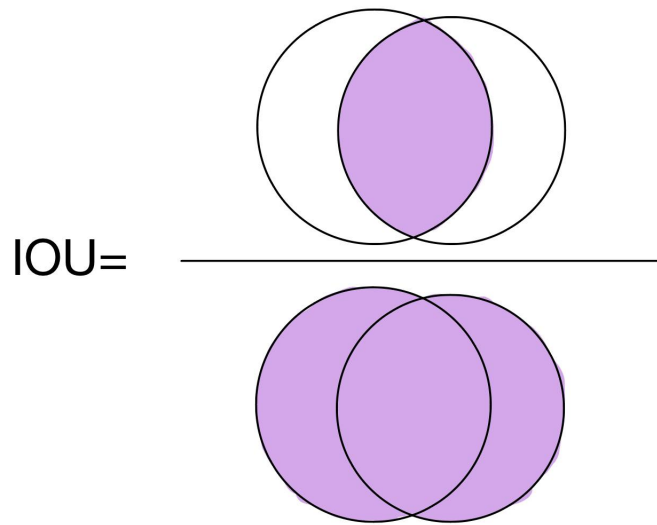


Figure 2.7: IoU - Intersection over Union.

### 2.2.6 Frames per Second

Frames per second (FPS) is a key metric used to evaluate the performance of computer vision algorithms [46]. It measures the rate at which a system can process video frames per second and is critical for real-time applications. A higher FPS indicates that the algorithm can process more frames in a shorter amount of time, which is important for achieving real-time performance in applications such as autonomous driving or robotics. In the context of the photobox, the FPS will evaluate how fast the algorithm can process images, i.e. how fast the photobox can be presented for new fish individuals to analyze [47].

$$\text{FPS} = \frac{1}{t} \quad (2.12)$$

# Chapter 3

## Methods and Implementation

This chapter presents the practical application of the theoretical concepts and techniques outlined in the previous chapter to address the objectives of the master's thesis. Firstly, the hardware tools and software frameworks employed in the study are presented. Subsequently, the data acquisition process is described, followed by the preprocessing steps and the necessary preparations of the data to train a neural network. Lastly, the implementation details of the U-net architecture are presented.

### 3.1 Technical Framework for Implementation

#### 3.1.1 Hardware

As the training of neural networks might be computationally intensive and can be time-consuming on CPUs, NTNU provided a GPU machine that was accessed remotely through the Windows software program *Remote Desktop*. The GPU used was NVIDIA GeForce RTX 3070. The hardware used for the data acquisition was a CANON XA15 video camera along with a tripod and a tripod extension.

#### 3.1.2 Software

This section provides information about the software set-up in the project and justifications for the choices made regarding software.

The programming language chosen for this project was Python 3.10 due to the availability of open-source libraries tools and resources for computer vision projects. These libraries have a lot of resources and documentation online, making them easy to use.

Pytorch 2.0.1 was chosen as the deep learning framework as it is a popular tool in deep learning research [48]. Pytorch is easy and intuitive to use and provides strong support for GPU acceleration, allowing faster training times. It also integrates seamlessly with other libraries like for example NumPy and OpenCV. Another alternative is TensorFlow, but Pytorch was understood to have better community support for research. The additional libraries used in this project will be presented and justified further in the following sections of this chapter.

Visual Studio Code was chosen as the code editor as it is designed to be lightweight, fast, and versatile.

The choice of the operating system for this project was Windows 10, as it was readily available on the provided computer and the candidate possessed prior experience with this operating system.

### 3.1.3 Training With GPU

The computational demands of training machine learning models can be substantial, which made it advantageous to use Graphics Processing Units (GPUs) due to their capacity to execute parallel computations. NVIDIA's CUDA 11.7 software platform has facilitated the development of code that can be executed on NVIDIA GPUs, rendering it an appealing choice in combination with PyTorch for its user-friendliness.

## 3.2 Field Data Collection

As indicated in the literature review discussed in subsection 1.1.2, limited efforts have been made to analyze diseases specifically in Atlantic Salmon, resulting in a limited availability of data including diseased salmon. Due to the lack of available data, it was necessary to undertake a field data collection.

### 3.2.1 Image Acquisition

On March 29th and 30th, 2023, image acquisition was carried out at sea-based fish farms located in a coastal region of Trøndelag. The data were collected from two separate facilities, each containing salmon in varying stages of growth. A third party agreed to provide a visit to do data acquisition for the thesis. The locations were decided by the company but were chosen as two locations with high mortality, providing more fish samples to capture. To ensure confidentiality for the company that facilitated the visit, the precise locations of the fish farms will not be disclosed, and will henceforth be referred to as location 1 and location 2. Location 1 where visited on 29th of March while location 2 where visited on 30th of March.

The first part of both data acquisition days was used to pump the dead fish from the bottom of the cages up in a boat, which can be seen in Figure 3.1. A camera placed at the bottom of the cages was used to observe that all the dead fish was removed from the cages. The collection of deceased fish from the sea cages was carried out by the third-party company.

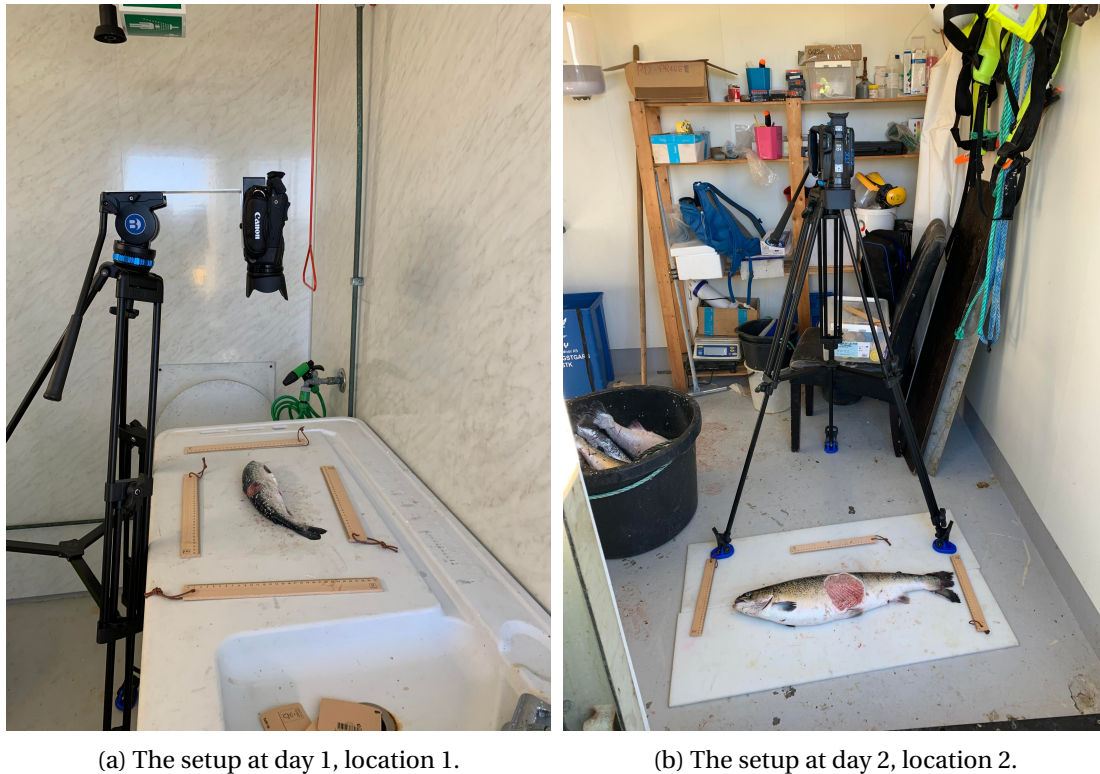


(a) A hose was placed in the bottom of the cage. (b) The dead salmon ended up in a container.

Figure 3.1: Dead salmon from the bottom of the cages were collected.

After collecting all the dead salmon at each cage in a container, a selection of dead fish was picked out of the container and put in separate containers to be photographed. The photos were taken at the employees' deeding barge, inside a veterinary lab. The setup was very simple, as demonstrated in Figure 3.2. The camera used was a CANONXA15 video camera provided by NTNU along with an adjustable tripod. An extension to the camera tripod where made at the NTNU TTK laboratory so that the camera lens could be pointed 90 degrees at the object without including the tripod in the resulting image. The light was limited to a lamp over the veterinary desk and some daylight from keeping the door open. The background was chosen to be white, as the other alternative was a stainless steel desk, with a color similar to the salmon scale.





(a) The setup at day 1, location 1.

(b) The setup at day 2, location 2.

Figure 3.2: Data acquisition setup at the veterinary lab.

The data collection process involved retrieving deceased fish from the containers and placing them on a standardized background for imaging. Some fish had winter ulcers on both sides, thus they were flipped, to photograph both sides. The amount of fish in location 1 was bigger than in location 2. Due to the lower quantity of dead salmon in location 2, the salmon were moved and rotated so that some additional images were captured of the same individual. The first location had fish with a weight of approximately 312-620 g and the other location had fish of approximately 5 kg, which is the fully grown size. This made the fish easier to carry, place and capture on location 1. Due to security protocols, personnel were required to wear multiple protective gear, including helmets, life jackets, and work suits. The use of gloves was preferred while handling the fish, however, to capture clear images, gloves were not worn during photography, which posed an inconvenience during the work. In addition, there was limited space in the veterinary room and the tripod used a significant amount of space. There was no possibility to connect the camera to an external monitor, that is, all the captured images were only available on the small camera screen. This posed challenges when evaluating the camera settings during the data acquisition and the effect of how this challenge affected the results will be discussed in Chapter 5.

### 3.2.2 Camera Settings

The suboptimal environment during data acquisition necessitated the exploration of various camera settings. On the first day, the sharpness of the images was deemed inadequate, prompting corrective measures to enhance the sharpness. Although the initial intention was to manually adjust the focus, the practical difficulties and limitations involved in assessing image sharpness on location necessitated the use of the auto focus setting. Consequently, images from location 1 on day 1 and location

2 on day 2 differed in quality, which will be further explained in the following section. The concept of having a photobox for automatic data collection will improve the possibility to choose the right settings and will be further discussed in Chapter 5.

### 3.2.3 The Raw Data Set from Data Acquisition

Day 1 of data acquisition resulted in 627 images and day 2 resulted in 160 images. The disparity in the number of pictures from the two locations was due to the amount of dead fish at the different locations and their size. Some duplicates were removed in the annotation process, leaving 470 images from day 1 and 38 images from day 2. The big amount of duplicates from day 2 was due to experimenting with different camera settings during the data acquisition. The images were saved as .png images.

The images from day 1, shown in Figure 3.3a, contain fish at an early stage of development, which yields a variety of wounds, with different colors and shapes. The images from day 2, shown in Figure 3.3b, contained fish of size ready for slaughter, with particular red wounds, and with a significant size. Some fish had more wounds, others had a bigger size. The images from day 2 are improved mainly in comparison to day 1 due to better light conditions at location 2 and better resolution settings from the camera. This resulted in the images from day 1 having a resolution of 640x360 and the images from day 2 having a resolution of 1920x1080. Ideally, a higher resolution would be preferred, allowing for a more detailed representation of the image, but this setting was not discovered on the first day of data acquisition.

The data acquisition process was carried out with the primary objective of capturing a large number of images, leading to the inclusion of multiple images of the same individual, with some rotations. An example from each of the days of data acquisition can be seen in Figure 3.3, the development phase of the fish at the different locations is demonstrated.



(a) Data from day 1, location 1.



(b) Data from day 2, location 2.

Figure 3.3: Raw data obtained at the data acquisition field trip.

### 3.3 Data Preparation to Segment Winter Ulcers

Due to the challenges of winter ulcer at the time of the data acquisition, it was decided to develop a model to segment winter ulcers. Other diseases were considered as well, but due to a restricted amount of individuals with other diseases it was decided that to get the best performance, and thus the best system for final software implementation in the photobox, a model for winter ulcers would be suitable to the data set. Image segmentation was the preferred technique to solve the task. It was considered the best solution to identify and measure the characteristics of the ulcer in each individual.

The code folder from this project contains several files depending on each other to prepare the data to be able to train and test the model. U-net require images and binary masks as ground truths on a specific format to be able to train a model. The information needed by each script and the results of running the scripts are shown in Figure 3.4. All the files will be presented and explained in the following section.

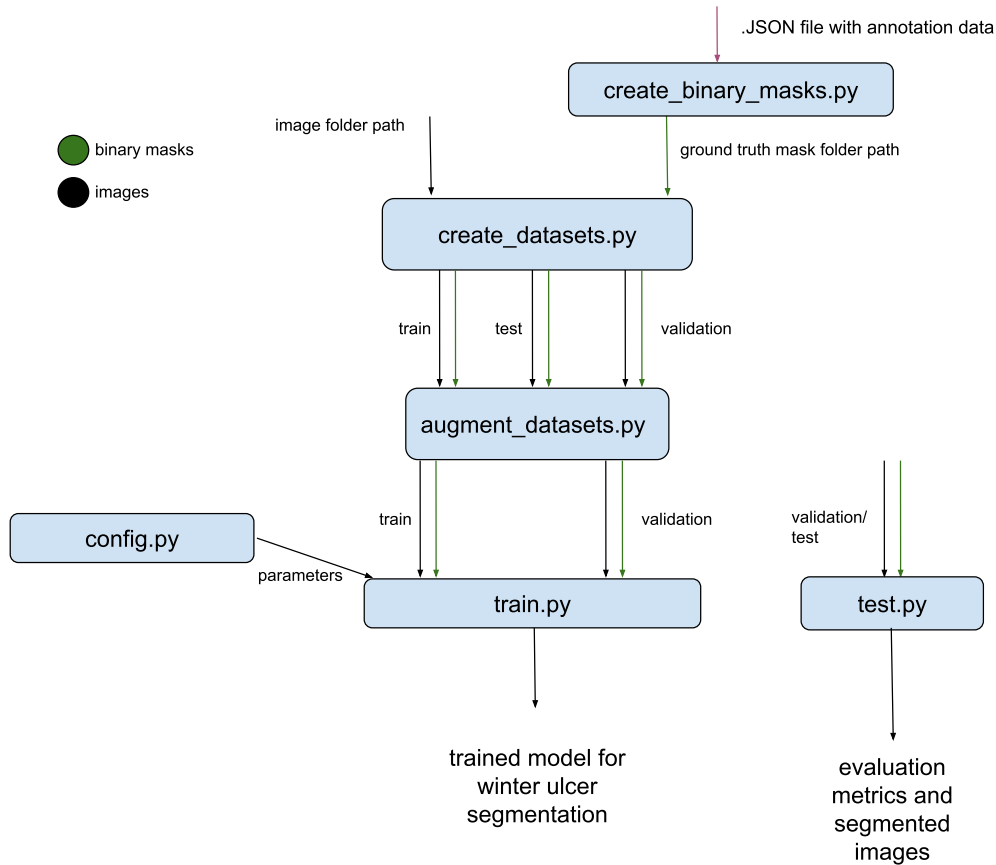


Figure 3.4: An overview of how the files created in the project work together and produce outputs that the latter files depend on.

### 3.3.1 Data Annotation of Winter Ulcers

The first step in preparing the data for the U-net model was annotation. U-net requires binary masks as ground truth, meaning that these labels of the ulcers must be sent into the model together with the images. The annotation was done by the utilization of the *Visual Image Annotation (VIA)* tool, which is a widely utilized software developed by the Visual Geometry Group (VGG) at the University of Oxford. The annotations were carried out on a publicly available online demo of the *VIA* tool, accessible at [https://www.robots.ox.ac.uk/~vgg/software/via/via\\_demo.html](https://www.robots.ox.ac.uk/~vgg/software/via/via_demo.html). The annotations may exhibit variability in terms of accurately delineating the boundaries of the wounds, especially at the wound edges, due to the manual annotation of 508 images. The images were meticulously annotated using polygonal shapes to delineate the precise regions of interest corresponding to the wounds. Some images were duplicates, and thereby removed. The resulting annotation data, which captured the spatial information of the wounds, was then meticulously saved in a .JSON file format, ensuring the preservation and accessibility for subsequent data analysis and processing. The use of the annotation tool is presented in Figure 3.5.



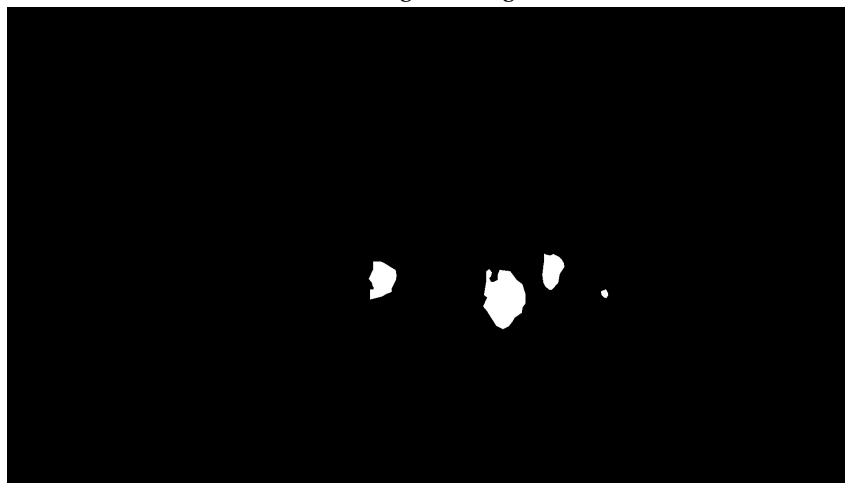
Figure 3.5: An overview of how the VIA tool was used to annotate the data. Image screenshot from [https://www.robots.ox.ac.uk/~vgg/software/via/via\\_demo.html](https://www.robots.ox.ac.uk/~vgg/software/via/via_demo.html).

### 3.3.2 Creating Binary Masks for the Winter Ulcers

The JSON files containing the annotated data were processed using a Python program, denoted as *creating\_binary\_masks.py*. This processing step involved extracting relevant information from the annotations to generate binary masks, which served as ground truth labels for the subsequent implementation of a U-net. These binary masks, accurately representing the spatial extent of the annotations, played a crucial role in training and evaluating the performance of the neural network model in the subsequent stages of the research. Figure 3.6 illustrate how the original images correspond with the binary masks.



(a) Original image.



(b) Binary mask.

Figure 3.6: Original image and the corresponding binary mask.

### 3.3.3 Splitting the Dataset into Training, Testing and Validation Datasets

After the annotations and the creation of a binary mask for all images, they were manually copied to a folder. The dataset folder contained two sub-folders, image and mask. A Python file for the creation of a training, test and validation set was created by running *create\_datasets.py*. The script provides the user with the ability to specify the proportion of the entire dataset to be apportioned into three distinct sets. Before distributing the image-mask pairs into the training, testing, and validation sets, the code employs a randomization process. To prevent the occurrence of duplicates within any of the folders, an auxiliary function was incorporated into the script, which necessitates invocation each time the creation of new datasets is intended. The division of the datasets ended up being 70% for training, 20% for validation and 10% for testing. This resulted in 355 training images, 101 validation images and 52 test images before data augmentation.

### 3.3.4 Data Augmentation for the Winter Ulcer Dataset

To increase the number of images, a data augmentation process was added to the pipeline. Data augmentation is the process of increasing the amount of data by applying adjustments to the existing

data [49]. This technique has demonstrated remarkable effectiveness when the availability of data is limited and was used in the originally released paper of U-net [49] [39]. Therefore, data augmentation was chosen as a valuable approach in this study. The objective was that the data should be realistic, like their original, but with a slight variation. The data augmentation program is to be found in *data\_aug.py*. The data augmentation techniques applied include horizontal flip, vertical flip, and rotation using the *imgaug* library. Every image in the training dataset was augmented to a horizontally flipped version, a vertically flipped version and a randomly rotated version within the limit of  $[-45^\circ, 45^\circ]$ . The augmentations are visualized in Figure 3.7. The size of the images was determined in the augmentation process. The annotations were done equally for the masks and the images so that the correspondence was kept. The test and validation images were sent through the *augment* function to get the right dimensions and format for the U-net, but they were not augmented. The data annotation process resulted in 1420 training images, 101 validation images and 52 test images. An example of the augmentations for an image and their corresponding masks can be seen in Figure 3.7 and Figure 3.8.

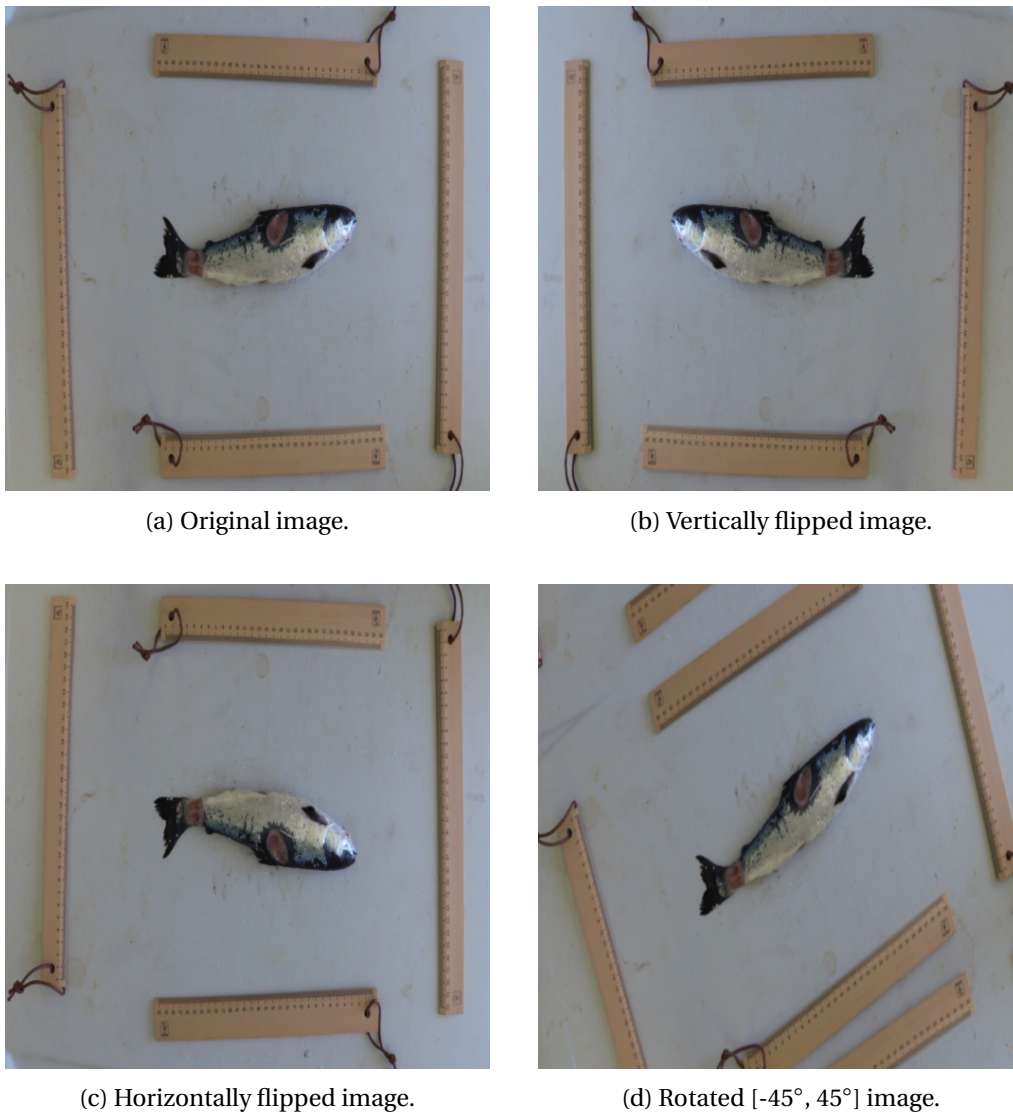


Figure 3.7: The data augmentations performed on the training images.

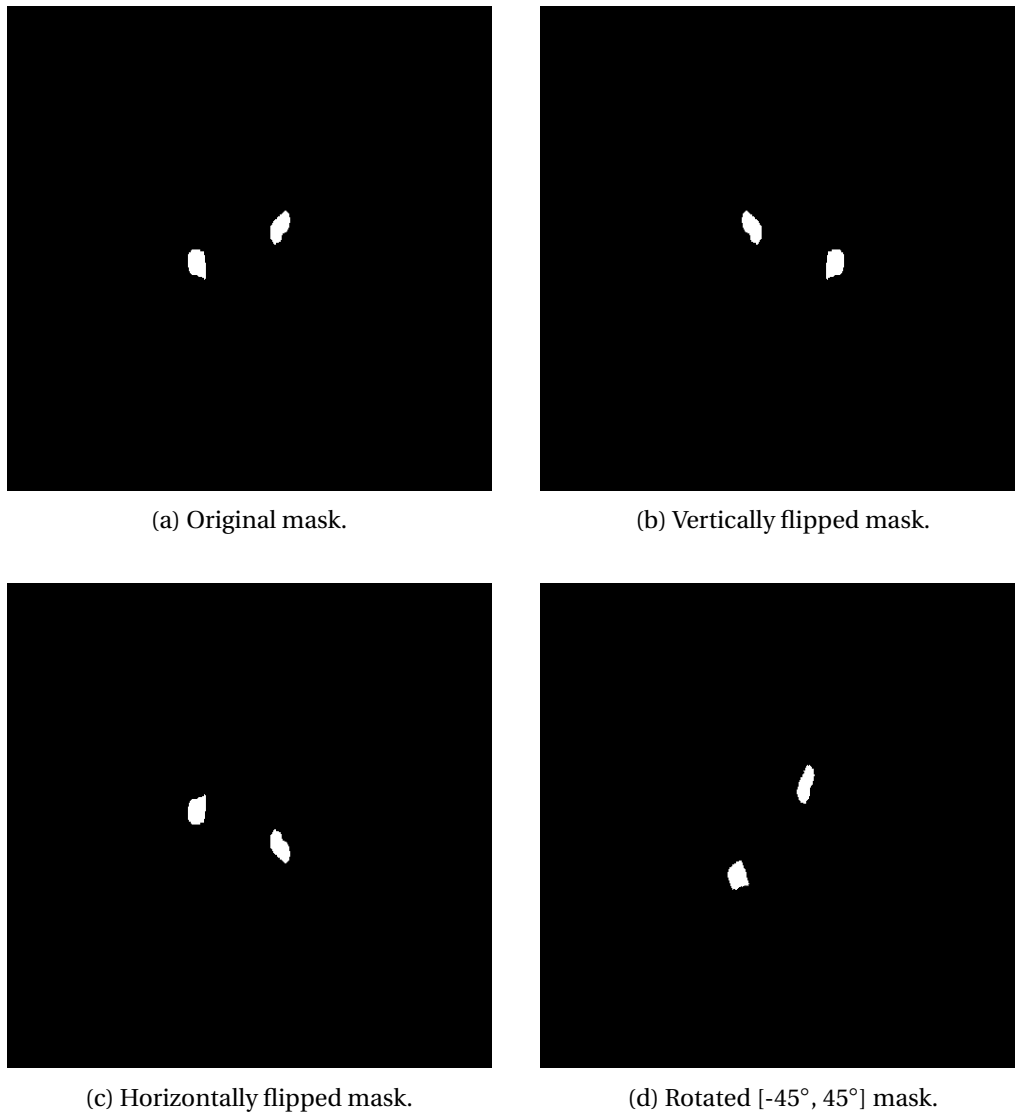


Figure 3.8: The data augmentations performed on the training masks.

### 3.4 Preparing Data to Segment Fish Bodies

To improve the likelihood of successfully segmenting winter ulcers, the background was removed to reduce the occurrence of false positive predictions. Some images did have a noisy background containing blood or other disturbances. In a few cases on the test set, these spots were segmented and predicted as ulcers. This led to the need for background removal, in addition to the interest in measuring the area of the fish body.

A wide range of techniques were tested to remove the background. The most desired case was to use an implemented library in Python or a pre-trained model from the internet to avoid the need for annotation of the bodies on the data, as annotation is time demanding. Some techniques and models were tested and explored, but none of them gave satisfying results. It was therefore necessary to create an additional model to segment the fish bodies and then remove the background. The same dataset with the 508 images from the field trip was used to develop and train this model. An overview of how the scripts in the development of the body model work together is presented in Figure 3.9.



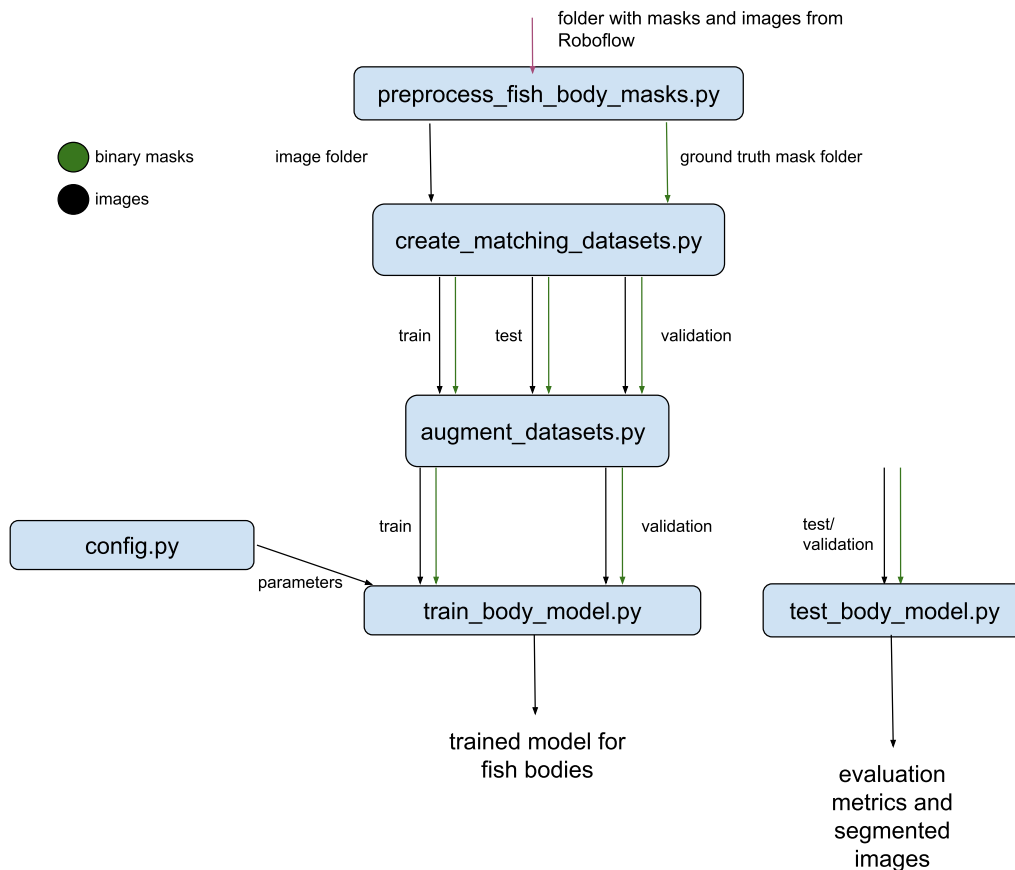


Figure 3.9: An overview of how the files created in the development of the body segmentation model work together.

This section will describe the work of the data preparation for creating a model to segment the fish bodies and thus separate the region of interest from the background. Since the focus of this project was to segment winter ulcers, the same model architecture will be used to train both models. The model design and implementation are to be found in Section 3.5.

### 3.4.1 Creating the Binary Masks for the Fish Bodies

To generate binary masks suitable for use in a computer vision model, an open-source platform for computer vision known as *Roboflow* was employed. The platform facilitated the use of machine learning models to assist in the annotation process, resulting in rapid annotation. The annotation process involved a single-click selection of the fish after just a few sample annotations.

*Roboflow* provided the possibility to download the dataset in one folder containing the masks and the images. Consequently, there was no need to use the previous method and read .JSON files. The masks were in a different format compared to the ones generated in the *creating\_binary\_masks.py* and needed preprocessing to work as label input to U-net. This happens in *preprocess\_fish\_body\_masks.py*, which split the folder with masks and images into two folders, one with masks and one with images and transforms the masks into the right format.

### 3.4.2 Creating Matching Datasets for the Two Models

The process from the implementation of the model for the winter ulcers was repeated with the fish body segmentation model. The difference was the creation of the datasets. Having two models working together required the test set to be the exact same. The idea was to send the test images through both models, where one model's output is the input of the other. The test sets was not seen by either model before testing, ensuring valid testing. The *create\_matching\_datasets.py* script takes in the two folders of masks and images for the fish bodies and sorts the images and masks belonging to the test, validation and training set for the ulcer model. This script takes advantage of the fact that the image IDs are kept in every process it goes through, resulting in easy data handling to create the correct datasets.

### 3.4.3 Data Augmentation for the Fish Body Dataset

The augmentation process was repeated for the fish body dataset. Due to the test dataset not being augmented, the datasets for the two models contained identical test images.

## 3.5 Neural Network Design

Once the data material was prepared, the neural network for automatic segmentation was designed. The desired objective to first segment the fish and then segment the ulcer posed the need for two different datasets, as described in the previous section. This resulted in one fish body dataset and one ulcer dataset. The focus of this thesis was to segment winter ulcers, the background removal was added as an extra functionality. For this reason, the same neural network, U-net, was used to perform the segmentation. The code implementation of U-net along with the arguments for the selection of U-net is presented in the following section.

### 3.5.1 U-net

U-net was chosen due to its special design for image segmentation tasks, where the input and output are images. In the case of segmentation of wounds on fish, the goal was decided to be segmentation of the area of the wounds, involving pixel-level classification. Other models, such as YOLO, can provide a four point detection of objects, creating a rectangle around the desired object, but U-net can provide pixel-wise classification, making it more suitable for segmenting wounds of different, non-geometrical shapes. U-net's architecture has been proven to be well-suited for this type of classification, allowing the capturing of both local and global context information and preserving spatial details. Previous literature has demonstrated the successful utilization of Mask-RCNN for image segmentation [50] [20]. However, U-net was considered advantageous due to the limited available data in this project and was consequently preferred. During the research of this project, it was found that a significant amount of successful segmentation tasks have been solved by U-net segmenting similar shapes and objects, as described in Subsection 2.1.3. U-net additionally have the advantage of being simple and efficient, requiring less computational resources than Mask-RCNN. Due to these arguments, the U-net architecture was deemed an optimal solution for addressing the problem at hand.

The U-net was implemented with inspiration from the code developed by Tomar, N. (2021) [47] which created a basic that was further developed and modified. The objective of the code base from [47] was to conduct semantic segmentation of retinal blood vessels in a limited dataset comprising a small number of images. This was deemed a favorable starting point, as this segmentation task exhibits a significant degree of similarity to the targeted problem being addressed in the current project. The net was implemented from scratch with the use of Python classes, making it intuitive and organized so that changes in the architecture could be implemented easily.

The implementation of U-net used in this project differs from the original U-net from [39] due to batch normalization inserted between the convolutional layer and the ReLU. This aims to reduce the internal covariance shift and make the network more stable during training. The following section will present the U-net used in this project and Figure 3.10 presents an overview of the architecture.

The implementation of U-net in both models consists of four encoder blocks and decoder blocks as seen in Figure 3.10. The input to the U-net was set to be a 512x512 image with 3 channels (RGB image). Each encode block consists of a convolutional layer, a batch normalization layer, a ReLU activation function and a max pooling layer. The convolutional layers consist of two 3x3 convolutions, the max-pooling layers have dimensions of 2x2. The encoder block generates two outputs. It sends the output from the convolutional layer as input to the corresponding decoder block and the output from the max pooling layer is passed into the next encoder block. In every encoder layer, the spatial dimensions of the image are halved and the number of feature channels are doubled.

The decoder blocks consist of an up-sampling layer, a concatenation with the input it gets from the skip-connection, hence the output of the corresponding encoder block, and a convolution block. The up-sampling is of dimension 2x2 and the convolution layer consists of two 3x3 convolutions. A bottleneck layer has been implemented as a bridge between the encoder and decoder block. This bottleneck compresses and fuse the information for capturing both low-level and high-level features. The final layer of the U-net can be seen as a classifier and is a convolutional layer with 1 kernel to output the probability map with the same 512x512 size as the input, but with 1 channel. To create a binary segmentation image, a threshold of 0.5 is applied to the output, such that all probabilities over 0.5 are assigned a value of 1, meaning they belong to the object. This results in a binary image which is seen as the prediction mask.

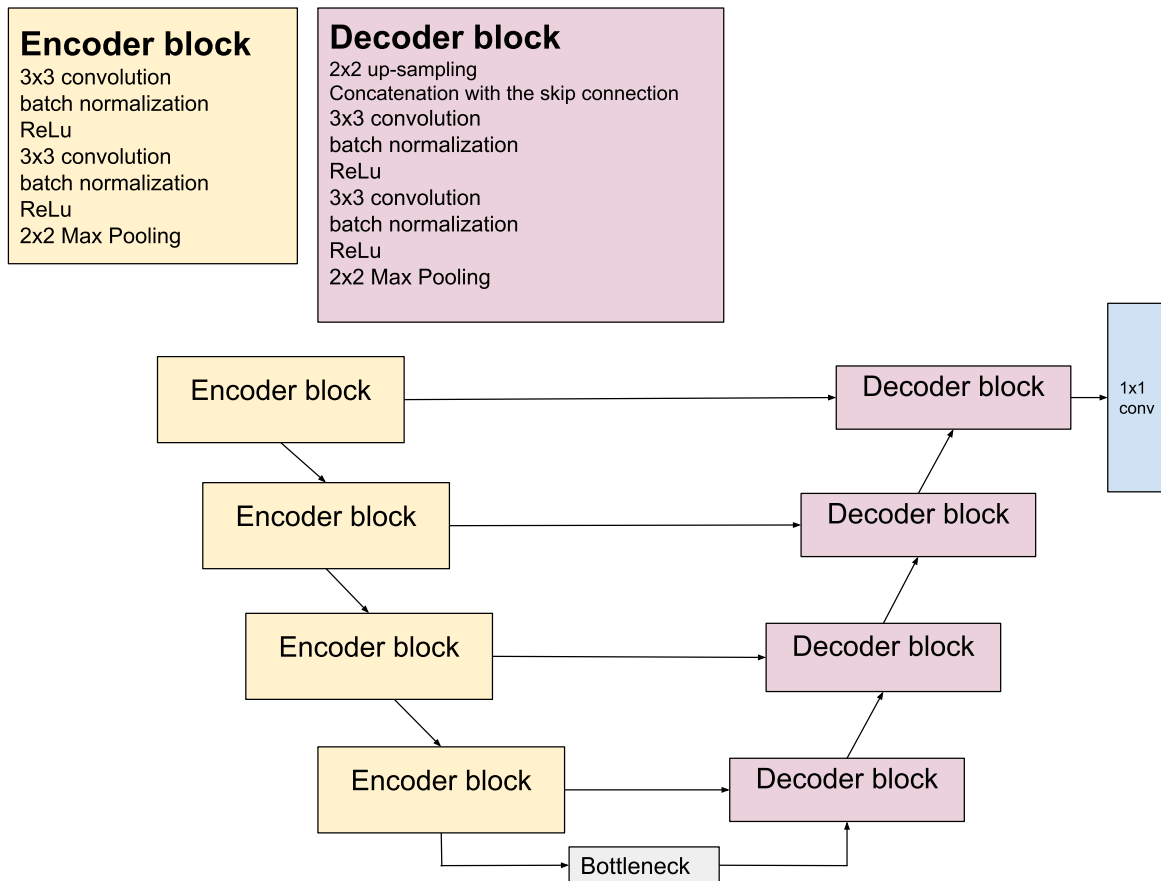


Figure 3.10: The U-net architecture.

The training and tuning of U-net is presented in chapter 4.

### 3.6 Combining the Two Models in Series

After the experiments to get the two models to work thoroughly, which is described in Chapter 4, there was some work left to put them together. The output of the fish body segmentation model, hereby called model 1, was fed into the ulcer segmentation model, hereby called model 2. The output of model 1 was a binary predicted mask for the fish body, which needed to be combined with the original test image with a binary AND operation. This operation is demonstrated in Figure 3.11. The selection of black as the background color was based on the divergence between the ulcers and the color. Although green could have also been a suitable alternative due to its dissimilarity from red, black was deemed preferable to facilitate data manipulation.

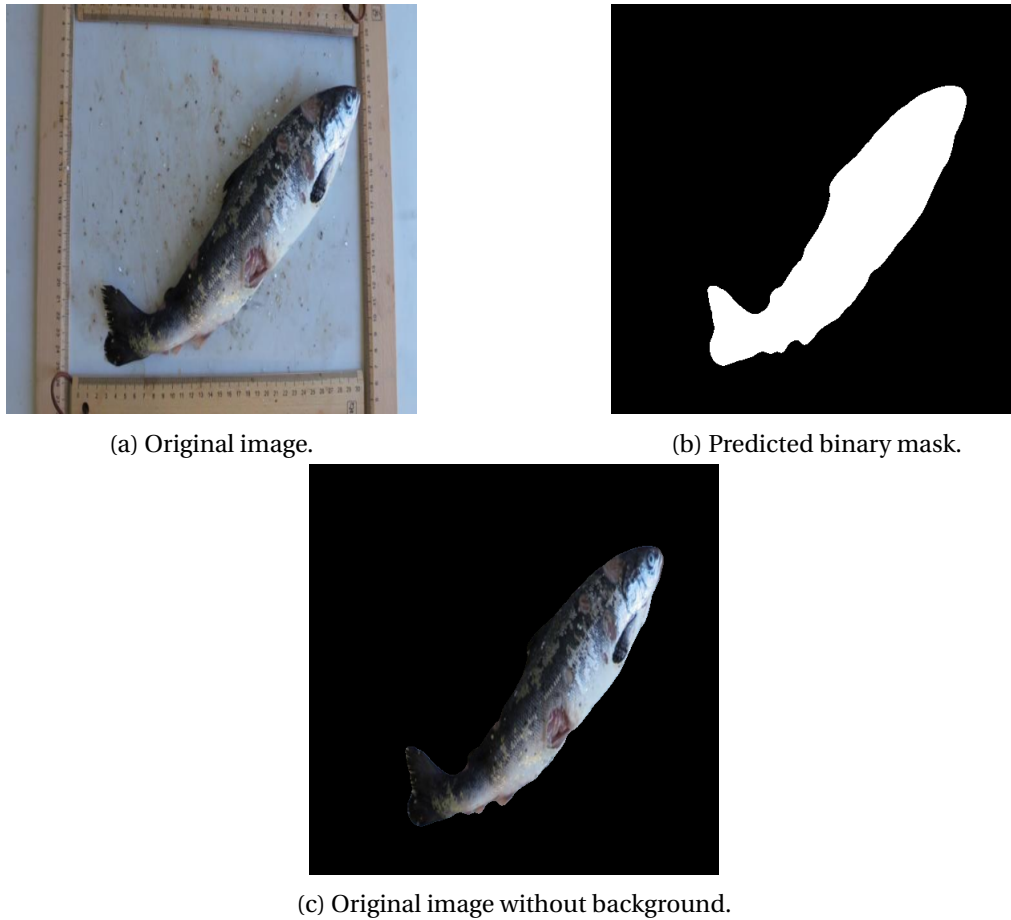


Figure 3.11: Result of original image AND predicted binary mask.

The workflow of how the two models are connected and work together to segment the images is shown below in Figure 3.12. The models work together in series. The input to the first model is images of fish in the right format. The first model segmented the fish body and thereby removed the background. The second model takes a picture with the fish body as an input and outputs an image with a segmentation of the winter ulcer. This system can be integrated into the photobox where the first input image will come from the photobox camera and the output image can be shown on an HMI screen for the employees handling the system.

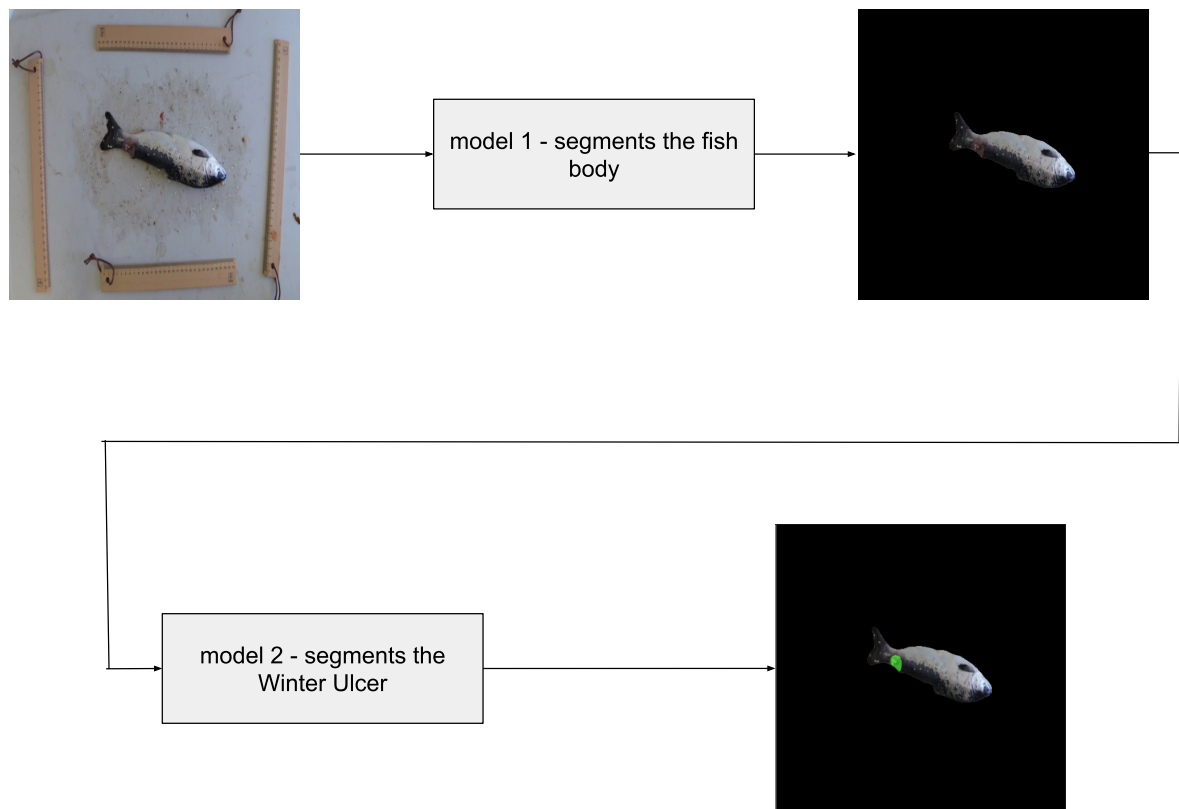


Figure 3.12: The flow of images through the developed models.

To extract additional information on the system performance, an additional indicator parameter called ulcer factor was derived. The feature is the result of dividing the number of pixels denoting the winter ulcer by the number of pixels denoting the fish body. The white pixels of respectively winter ulcers and fish bodies were saved to two different .JSON files and the data was processed further so that an ulcer factor could be found. This was done due to the lack of a known correlation between the pixels and the real-world measures. The ulcer factor can be seen as the percentage of the surface of the fish that was infected by winter ulcer. This was added to demonstrate the possible features that could be easily integrated into the photobox system if desired.

## Chapter 4

# Training and Results

Following the design and implementation of U-net, as explained in the previous chapter, some fine-tuning was needed to achieve the best performance of the model. The best model was found through experiments with different combinations of hyperparameters. Various combinations of different hyperparameters and model architectures were trained and evaluated before a final model was chosen. The training of the models was a time-consuming process posing a limitation on how many models that could be trained. This chapter presents the experiments and methods used to find the resulting model which will be presented in the chapter additionally.

### 4.1 Training U-net

To get the best performance for image segmentation of winter ulcers, a lot of experimentation was performed, experimenting with different combinations of hyperparameters, data augmentations and model depths.

During the experiments, an issue related to limited memory access on the GPU was encountered. As a consequence, the combinations of batch sizes, model depths, and image sizes were restricted. The implications of this error on the obtained results will be examined and elaborated upon in Chapter 5.

The models for fish body segmentation and winter ulcer segmentation were trained and tested separately. The experimental methods of evaluating different hyperparameters were similar for both models.

The tuning procedure involved training the model for a specified number of epochs by executing the training files. Subsequently, the models were evaluated by executing the test files using the validation data as input. The validation loss and the training loss were plotted during training, so that the performance could be monitored. This was considered to be an important metric to avoid the network to overfit the training data. Early stopping was implemented to avoid overfitting and stop the training when the validation loss did not improve enough. The requirement for early stopping was set to be an improvement in the validation loss of less than 0.001 for 5 epochs. After running each model, the model was saved as a checkpoint. The test file ran the 101 validation images through the trained model and calculated the IoU, F1-score, Recall, Precision and Balanced Accuracy. These metrics were

taken into account when deciding what parameters to adjust and which model to prefer. The metrics were saved to a .JSON file, together with the hyperparameters used to train the model. This enabled a clear overview of the different models that were trained and their performance on the validation set.

The loss function chosen for the training of U-net was the region-based loss function, Dice loss, as this has been shown to perform well on class imbalance problems [51]. Dice loss is a popular choice of loss function in medical image segmentation [52]. The goal is to minimize the error of the overlap, and dice loss is related to the overlap of the ground truth and the prediction. These are the main reasons for choosing dice loss in the advantage of for example cross entropy loss.

Due to the mentioned memory issues with the GPU used to train this model, there were done some experiments with different image sizes. Having smaller images allowed for increasing batch sizes, but the performance was decreased when the image size was decreased. This is due to a loss of information when the original image is scaled down. The smaller images may have fewer details which result in less accurate predictions. Choosing a large image size, however, requires more computational resources to process, and there is a trade-off between the computational resources and image size which needs to be balanced. From these experiments, it was decided that the recommended image size should be 512x512.

The optimizer used was the Adam optimizer, which is popular in deep learning and in U-net and has proven robust performance in CNNs [53]. Due to efficient memory usage and fast convergence, this optimizer was favored. The Stochastic Gradient Descent (SGD) optimizer was tested but exhibited slower convergence in the losses than the Adam optimizer. The slow convergence might be due to the small batch sizes as SGD updates the network parameters at each mini batch [30]. Due to limited computational resources requiring small batch sizes, the SGD optimizer was ultimately discarded.

As described in the theory in Subsection 2.1.3, the output of U-net will contain probabilities for each pixel to belong to the positive class. A threshold is then needed to determine if the pixel should be set to 1 for positive pixels or 0 for negative samples. The threshold was set to 0.5 after experimentation, indicating that all probabilities over 0.5 were considered to belong to the ulcers.

Several experiments were done with different depths of the model. Using the model design seen in Figure 3.10 as a base, two additional nets were made. One was made as a smaller net, removing the two last layers before the bottleneck, giving 3 encoder blocks and 3 decoder blocks. The other net was made by adding an encoder and a decoder, thus it had 5 encoders and 5 decoders. In Table 4.1, these architectures are denoted by their depths, respectively 3, 4 and 5. The different depths are demonstrated in Figure 4.1. The content within the blocks remains identical to that depicted in Figure 3.10.



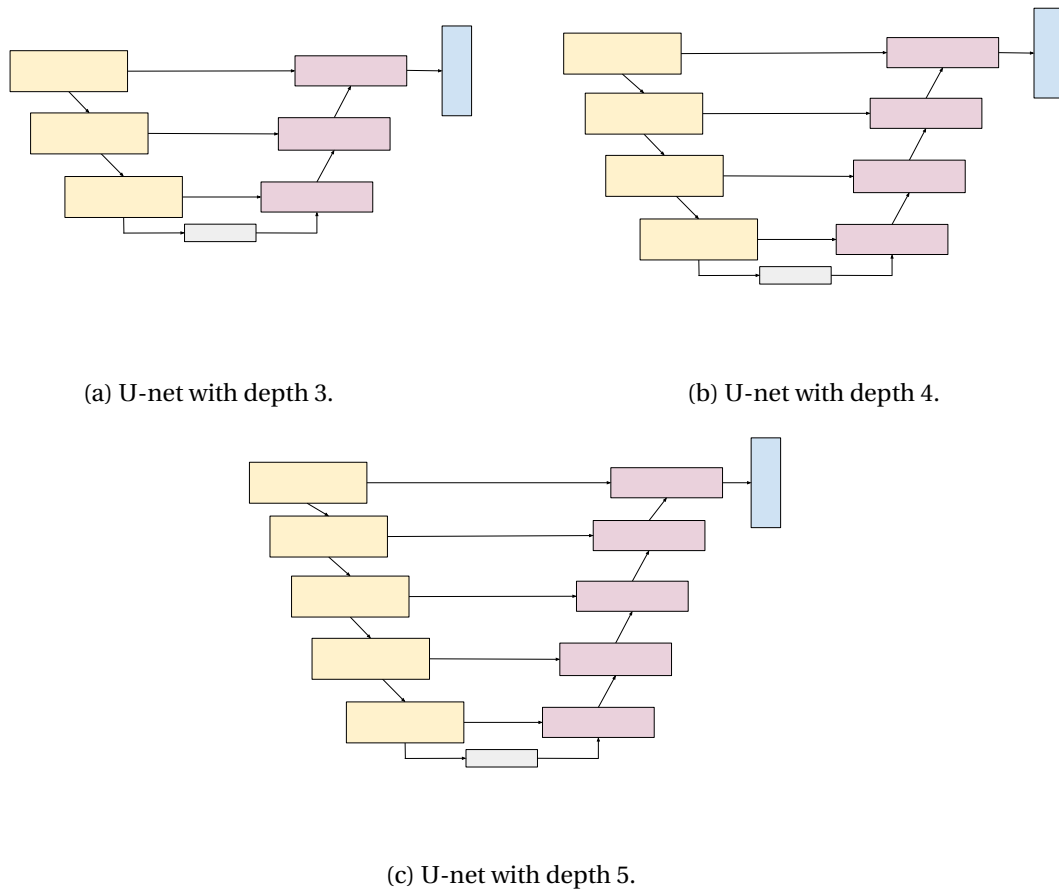


Figure 4.1: A visual presentation of U-net with different encoder/decoder depths.

## 4.2 Experiments

A selected combination of hyperparameters and image sizes was tested during tuning. Some combinations will be presented here, but additional models were trained in the search for the optimal model. The different combinations to be presented can be found in the table below. Early stopping is denoted as ES in the table.

Model	Image Size	Batch Size	Number of epochs	Learning Rate	U-net depth
1	255x255	16	100 (ES 68)	0.0001	4
2	512x512	16	100 (ES at 67)	0.0001	3
3	512x512	16	300	0.00001	4
4	512x512	12	50 (ES at 42)	0.0001	5

Table 4.1: Table overview of the model combinations.

Model	F1	IoU	Recall	Precision	Balanced accuracy
1	0.8484	0.7693	0.8343	0.8925	0.9169
2	0.8525	0.7763	0.8397	0.8989	0.9197
3	0.8437	0.7673	0.8354	0.8840	0.9175
4	0.8748	0.7995	0.8633	0.9094	0.9315

Table 4.2: Metrics overview of the model combinations.

What can be seen from the tables is that the depth of the network does not have too much signification for the metric results. This can be because the model already has reached a good level of complexity and increasing the layers too much may lead to overfitting. Additionally, the data is limited and may not have enough features to warrant a more complex model. Nevertheless, the models with a depth of 5 exhibits faster convergence and the early stopping criterion were triggered.

What was experienced additionally was that the batch size does not affect the performance significantly, but as explained in the previous section, there were limited choices of batch sizes due to limited GPU memory. In the experiments, the batch size where thus set to be as large as possible, or slightly less. There is a common practice to set the batch size to  $2^k$ . This is not a requirement, and as the GPU had memory issues, other batch sizes were explored. The models of depth 5 triggered early stopping and achieved a good performance although they were trained for fewer epochs.

Two interdependent hyperparameters that required tuning were the learning rate and the number of epochs. The learning rate influences the rate at which the optimizer converges to an optimal solution, and therefore, a lower learning rate requires more epochs to reach convergence. Given that early stopping was implemented to prevent overfitting and reduce the computational burden of training, there was no need to limit the number of epochs. However, the time spent for training was dependent of the epochs, with higher epochs being time demanding.

The results displayed in Table 4.3 indicate that the balanced accuracy exhibits a consistently high level of performance across all applications. As a result, during the parameter tuning and selection of the optimal performance, the balanced accuracy metric was not given a significant emphasis or weight. The evaluation process placed significant emphasis on the F1 score and Intersection over Union (IoU) score as the primary metrics. These metrics were considered highly important due to their ability to measure the overlap between the predicted area and the ground truth in the context of segmentation. The performance of the model was primarily determined by these metrics, given their direct relevance to the segmentation task. Although the other metrics were not disregarded, they served as additional control parameters during the selection of the model with the best performance. Their inclusion ensured a comprehensive assessment of the model's effectiveness beyond the F1 score and IoU score, providing a more holistic evaluation of its segmentation capabilities.

The final results for the best model chosen based on the validation data from Table 4.3 will be presented in the following section.

### 4.3 Metric Results From the Resulting Ulcer Model

Following the tuning of the model through the utilization of validation data performance metrics, the final evaluation of the optimal model was performed on the test data. The 52 test images were run through model 5 from Table 4.1. The performance of the best model is presented in the table below:

Metric	Score
F1	0.8160
IoU	0.7424
Recall	0.8145
Precision	0.8301
Balanced Accuracy	0.9168

Table 4.3: Metrics for the resulting model for segmentation of winter ulcers.

The best model performance was obtained by combination 5 in Table 4.1. The F1 score and Intersection over Union (IoU) metrics provide insights into the degree of overlap achieved by the model in segmenting the ulcers. The results indicate that the model demonstrates satisfactory performance in accurately delineating the ulcer regions. The model shows a well balanced accuracy which takes into account the class imbalance. The FPS was 25.05 images per second, meaning that the model was able to process 25.05 images each second during testing.

The model has an overall decrease in the evaluation metrics for the test data. The primary reason is the difference between the test data and the validation data, which was selected randomly. As the model was tuned with the performance of the validation data, the model was optimized to this particular data. Consequently, the model could not obtain the same performance for the test data. The model's performance on the test data reflects its capability to handle real-world data that never have been encountered.

The plot of the validation and training loss for the chosen model can be seen in Figure 4.2. The plotted results indicate that both the validation set and the training set are relatively representative of the population, as the disparity between them is minimal. Additionally, the plot provides evidence supporting the adequacy of the dataset sizes in with each other. The observed low losses signify that the model is effectively fitting the data. The plot also shows an early stopping at epoch 42, which prevents the model from overfitting. The results obtained will be further discussed in chapter 5. The next section will present the output images from the model.

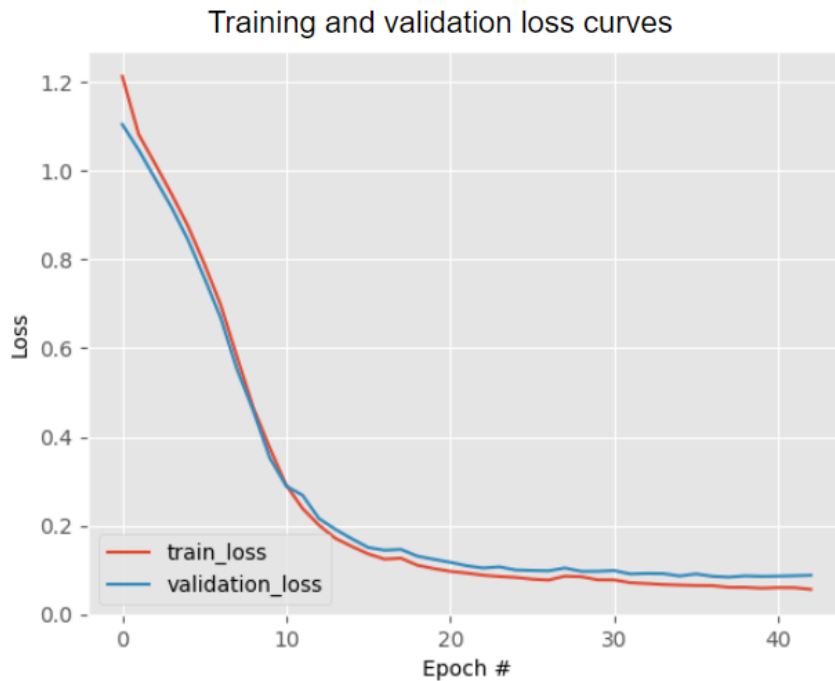


Figure 4.2: The curves for validation loss and training loss for the resulting model.

#### 4.4 Results Visualization

The test results from the resulting model are shown in the figures below. This type of visualization of the results provides an intuitive understanding of the predicted segments, but is harder to evaluate thoroughly. However, the visualized results provide information that could be of great interest to the aquaculture industry.

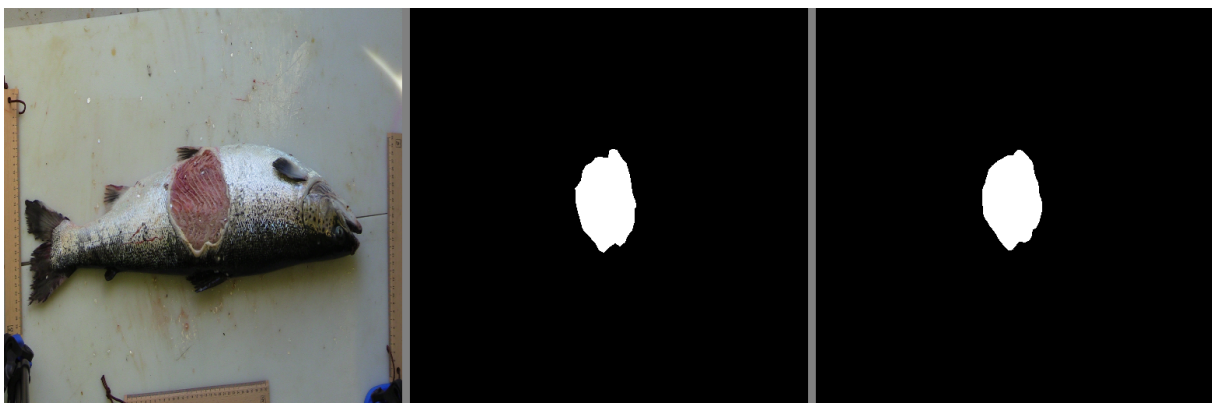


Figure 4.3: Example prediction: Original image, ground truth mask, predicted mask.

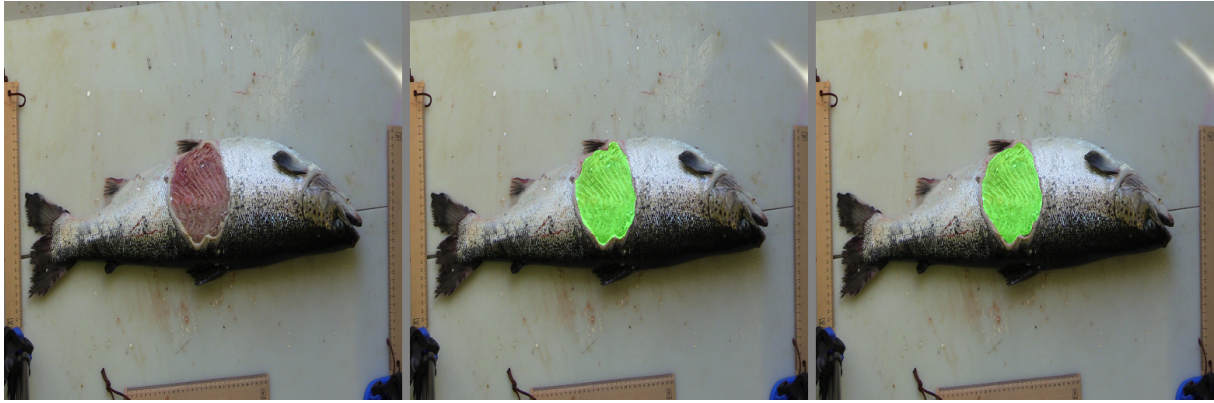


Figure 4.4: Example prediction: Original image, ground truth mask, predicted mask.

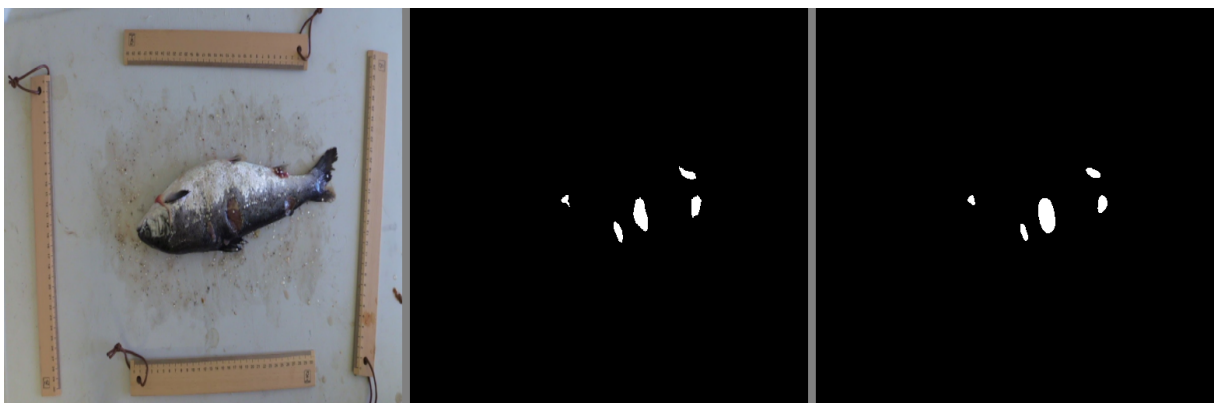


Figure 4.5: Example prediction: Original image, ground truth mask, predicted mask.



Figure 4.6: Example prediction: Original image, ground truth mask, predicted mask.

The model predictions have been presented in two different manners. The visualization in Figure 4.3 and Figure 4.5 gives the possibility of an intuitive comparison of the predicted area and the ground truth. Figure 4.4 and Figure 4.6 presents the segmentation on the original masks.

In the first example image, Figure 4.3 and Figure 4.4, the model exhibits less precise edge detection compared to the ground truth. The second example, Figure 4.5 and Figure 4.6 demonstrate the model's capability to successfully segment all ulcers present on the fish. However, there is a notice-

able decrease in edge accuracy. The predicted results when marked on the original images show an overall satisfying performance. The results will be discussed in Chapter 5.

## 4.5 Results After Moving the Background

The process to find the best model to remove the background was identical to the process of finding the best winter ulcer model. A selected amount of models were trained, tested on the validation set and evaluated. The best model to remove the background was obtained with the combination of hyperparameters as shown in Table 4.4. The results obtained when testing on the 52 test images for the fish body model are presented in Table 4.5. The results are slightly better than for the ulcer model, which indicated that the segmentation of fish bodies is very well suited for the U-net with this dataset. This might be due to the fish bodies having a more consistent visual shape and color. The fish also exhibits more distinct borders than the ulcers. Additionally, there was one fish in each image, in comparison to multiple winter ulcers. The results obtained for the fish body model was obtained by only having 20 epochs, which when compared to the ulcer model demonstrate how the fish bodies are easier to segment than the ulcers.

Model	Image Size	Batch Size	Number of epochs	Learning Rate	U-net depth
1	512x512	4	20	0.0001	4

Table 4.4: Hyperparameters combinations for the resulting body model.

Model	F1	IoU	Recall	Precision	Balanced accuracy
Fish body model	0.9608	0.9304	0.9767	0.9522	0.9817

Table 4.5: Metric results for the fish body model.

As expected, the background removal after adding a fish body segmentation model to the system improved the predictions in certain images. Below follows a sample containing noise in the background, leading to false positive predictions on the background noise in Figure 4.7. Figure 4.8 demonstrates a prediction of the same sample after having passed through both the models, which avoids the false positives seen in Figure 4.7.



Figure 4.7: Output after only winter ulcer model: Original Image, segment ground truth, predicted segmentation.



Figure 4.8: Output after going through the system with two segmentation models: Original Image, segment ground truth, predicted segmentation.

Nevertheless, the overall performance of the combined system, resulted in a poorer performance compared to when the model was provided with images containing the background. The performance of the best model, respectively model 5 in Table 4.1 when the background was removed is presented in Table 4.6. The lower performance might be due to a data distribution mismatch. By removing the background during testing, the introduction of a different data distribution is made compared to what the model was trained on. The model learned the segmentations of the wounds based on the presence of the background during training and struggled more when presented for images without background. The removal of the background also provides a loss of contextual information which may have helped the model to make accurate segmentation predictions. The recommendations for how this can be avoided or if the background should be removed or not can be found in Chapter 5.

Model	F1	IoU	Recall	Precision	Balanced accuracy
Total system	0.6657	0.7396	0.7265	0.7842	0.8822

Table 4.6: Metric results for the combined system with the body model and the ulcer model.

## 4.6 The Ulcer Factor

As mentioned in Chapter 3, an additional feature, the ulcer factor was added. An example of a result can be seen in Figure 4.9. According to the calculated ulcer factor of 0.2199, it can be inferred that approximately 21.99% of the visible fish surface is affected by ulcers. This value is difficult to verify, but can seem reasonable by inspection of the image.



Figure 4.9: IMG\_0353\_0.png resulted in an ulcer factor of 21,99%.



## Chapter 5

# Discussion and further work

This chapter will present the findings of this thesis and their implications. It will explore potential areas for further research and discuss the strengths of the implemented methodology. Additionally, the concept of the photobox will be examined in the context of the findings from this thesis.

### 5.1 Discussion and Evaluation of the Model Performance

The implementation of the developed model for segmenting winter ulcers yielded an F1 score of 81.60% and an Intersection over Union (IoU) score of 74.24%. These results validate the effectiveness of the employed methods in achieving the objective of segmenting winter ulcers, demonstrating satisfactory performance. This emphasizes that this method can be applied to the photobox to segment winter ulcers or other similar-looking damages to deceased fish. Despite the reported Intersection over Union (IoU) scores of 92.03% and 77.56% achieved by Ronneberger, Fischer, and Brox in their publication on the U-Net model [39], the performance of the neural network employed in this project is considered satisfactory for a manually created dataset. The performance can be seen as a good first model in the development of robust software for the photobox. The previous literature in this field demonstrates the interest in classifying skin diseases as this is a considerable global problem in aquaculture. In this particular case, it is noteworthy that the successful segmentation of winter ulcers on deceased salmon using neural networks has been achieved, which, has not been previously explored according to the literature review. This accomplishment serves as a proof of concept, demonstrating the viability of this approach.

The visualized results in section 4.4 can be interpreted as good results as the classifier segments the winter ulcers. When the predictions are put on top of the original image, the resultant visual coherence is notable, as human visual perception finds the predictions to be aesthetically pleasing without any discernible shortcomings. While the precision of the edges may not match that of the ground truth, it is worth considering that the significance of precise edge detection may not be of paramount importance in this context. Section 4.4 shows that all ulcers are segmented, which can be used for accurately counting the ulcers. An essential factor in the evaluation of the model and the desired accuracy is the fact that the photobox is analyzing dead animals, which makes it less critical for the animal and its welfare.

The choice of segmentation of winter ulcers provides the possibility to extract multiple features that can provide various information. If the objective is to classify whether a winter ulcer is present or not, the classifier could satisfy the objective by searching for any positive pixels in the outputs. The classifier could achieve high accuracy as the model segmented all winter ulcers in the test set. The complex information about shape and localization can provide additional information and allows for the classification of the ulcers in regards to size, shape, color or localization.

## 5.2 The use of Neural Networks to Segment Winter Ulcers and the Choices for the Final Solution

During the code development process, the issue of utilizing the oldest segmentation techniques, specifically pixel thresholding or blob-detection, arose due to the observation that the selected fish sores particularly exhibited a distinct deep red color. Observing the dataset, some sores were appearing to be white and of a different color, demonstrating the need for a more complex algorithm with deep learning. This posed the need for a more complicated algorithm than the simplest segmentation techniques. The relevant literature demonstrated multiple applications of Support Vector Machines (SVMs). This could have been a great approach if only the presence of the ulcers were desired classified, the implemented model can provide more information and denote the shape and localization of the ulcers additionally.

U-net was decided to be the best alternative of image segmentation nets to solve the task in this project. Some other popular alternatives for image segmentation were Fully Convolutional Networks, Mask R-CNN and DeepLab [18]. U-net was chosen due to its ability to handle small training datasets, as the data available in the project was limited. A viable approach for this project with limited data could be to use transfer learning with models like Mask R-CNN and fine tune the weights for the specific problem in the project. However, the findings obtained in the thesis demonstrated the suitability of employing the U-net model for accomplishing the objectives of the undertaken task. A reasonable inference can be made that an increased volume of data has the potential to positively increase the performance of U-net, as more data will help the model to generalize better. Nevertheless, it is essential to take into account certain limitations of U-net in future research. Due to its significant computational demands, the deployment of U-net in a real-time photobox scenario can pose challenges, necessitating substantial computing resources. Additionally, U-net offers limited insight into the uncertainty associated with its predictions. After the introduction of U-net, numerous variations have been published, which may be favored in future research endeavors [54].

In the search for an optimal solution, two models in series, each segmenting different objects in the image, were implemented. This approach proved to be effective in achieving accurate segmentation of winter ulcers when the background was noisy. However, the results of the tests did not improve, which was likely to the loss of contextual information and different data distribution between the training set and the test set. The metric results for the total system when tested on the test image set were not expected to improve as the ulcer model was trained on images with background. However, the model was put in series to demonstrate how the flow of the photobox could be. A possible approach can be to train the winter ulcer model on images without background to solve these issues.

That could have provided a system more robust to noisy backgrounds. The amount of data was considered insufficient for this approach, since the desired way to test the system was to send images through both models that neither model had encountered before. To be able to train the ulcer model on data without background, one must first have trained the fish body model to remove the background, then train the ulcer model on the images from the test set after the first model. This was not done due to small datasets and the results simulate a real scenario where both models are presented for images that have never been seen. In the latter implementation, one should aim to train the ulcer model on images without background if two models in series are preferred. An approach in this project could also have been to remove the background manually and then train the model, but it was chosen to add a background removal after the ulcer model was trained and it was desired to automate the whole process.

One advantage of choosing the method with two separate models is the flexibility this provides. Separating the models provides easier possibilities for debugging and troubleshooting the models. The output of the first body segmentation model can easily be fed into other models for different disease analyses with computer vision. Another advantage of removing the background is that one can find a Region of Interest (RoI) and thereby decrease the computational resources required by the model to perform further predictions. The main drawback of this solution is the need to train two models and annotate two different types of objects, creating the binary masks needed. An alternative strategy could be to use a different algorithm, like for example YOLO, to detect the fish and thereby the RoI. Feeding the Ulcer Segmentation model with the output from the Fish Body segmentation model may lead to mistakes being transferred from one model to the other. This may affect the performance of the whole system and its correspondence to the real ground truth as the first model may remove information that the second model would have marked. The additional ulcer factor is vulnerable to errors in the two models and depends on their performance.

An alternative approach to employing two sequential models involves the utilization of a Multi-Task Learning (MTL) model [55], which aims to simultaneously predict multiple labels for the objects of interest. The model's effectiveness relies on the presence of task similarity, which can potentially result in reduced generalization performance. Additionally, it should be noted that such models demand a substantial volume of data and exhibit inherent complexity. To achieve the task of distinguishing between a fish body and winter ulcers, which are dissimilar objects, it was chosen to utilize two sequential models.

### **5.3 The Data Acquisition and the Dataset Used**

The data acquisition process played a crucial role in determining the outcome of this project, as the performance of neural networks is heavily influenced by the quality of input data. In other words, the network can never be better than the data it is trained on. The data collection endeavor can be deemed successful, given its acquisition of usable data, which, in turn, facilitated the development of a proficient segmentation model for winter ulcers. The data annotation, being a time-demanding process, allowed for satisfying results which can be evaluated to cover the winter ulcers satisfyingly.

Moreover, the data holds potential for future utilization by other researchers, who can leverage it to implement different algorithms. The creation of the dataset for this study fills a research gap, as no previous instance of this dataset had been established, signifying a novel contribution to the field.

As detailed in subsection 3.2.1, the data collection process during the field trip posed several challenges. The manual creation of standardized images was particularly difficult due to limited light conditions and restricted time at the fish farms. This may have introduced a bias that can affect the transferability to the resulting photobox. The data quality could have been significantly improved if optimal light settings were available, and if a team of more people was available to divide the tasks into one handling the fish and another assisting with photography, the photos could have had a better uniform format. During image collection, the screen of the camera was not very visible, posing poor calibration and correction of settings during fieldwork. This left unused potential in image quality when it comes to accuracy, clarity and reliability of the images. Further investigation is warranted to explore the trade-off between image resolution and the computational resources required when selecting camera settings for the photobox.

Another challenge to consider is the similarity between the background and the lower fish body. Having a uniform background is an advantage, but preferably, the color should be one that was not represented in the object. This would give a big advantage for the algorithm in separating the object of interest from the background and might have made the process of removing the background easier, making the images more suitable for pre-trained models and libraries. However, the implemented model to segment the fish bodies did get notable performance with an IoU of 93,03% and an F1 score of 96.0%, and the selection of a white background over a stainless steel desk can be deemed as the appropriate choice. The photobox should have a background that can easily be separated from the fish, meaning that the color should not appear in the fish.

The quantity of salmon to be captured was limited when doing data acquisition manually, hence the amount was also affecting the performance results of the model. The limited time period of the data acquisition may have imposed a bias, making the dataset not representative of the underlying population and their ulcers. A short time of data acquisition may in general have made an impact on the diversity of the dataset, and preferably, the dataset could have been collected at more locations with varying development phases, temperatures and sea cage sizes to better get a diverse dataset. Another possible source to bias is the selection of the dead fish from the boat. The fish were selected from their sores and the fish with the biggest and most notable sores might have been favored in comparison with the ones with less visible sores. This may have decreased the dataset's ability to represent the real population and the diversity of winter ulcer cases. The augmentation techniques were meticulously chosen to ensure the absence of any patterns or biases that were not inherent in the authentic dataset. However, the test set remained relatively small. This could potentially affect the conducted evaluations since the test set may lack representativeness for the overall performance when faced with unseen data [56]. Striking a balance between excessive data augmentation and compromised generalization is essential, and the approach in this thesis aimed to achieve this trade-off. In summary, it can be postulated that the dataset might exhibit limited representativeness concerning the broader population of deceased salmon derived from sea cages in aquaculture. The mention of the

proof of concept remains noteworthy, highlighting the applicability of the employed methods.

The dataset might affect the transferability to the photobox, but can be avoided if the data acquisition process is improved. To effectively collect better data, a substantial quantity of fish exhibiting diverse symptoms would need to be collected over a potentially prolonged period. Ideally, it could be recommended to transport the dead fish to a controlled environment, such as a factory, where standardized image acquisition methods can be employed under controlled lighting conditions. Ideally, the hardware of the photobox could be developed and the data acquisition could happen inside the photobox, creating standardized data. If high-quality data containing a wide range of diseases is available, the approach utilized in this thesis could be extended to facilitate the implementation of the photobox being developed by ScaleAQ. Certain adjustments may be required to the existing methods, given that the characteristics of diseases can differ.

The inclusion of rulers within the images was undertaken to facilitate the comprehension of the pixel-to-real-world measurement correspondence in centimeters. Regrettably, this intended strategy did not yield the desired outcome owing to the inadequate resolution of the images. Nevertheless, it is important to note that despite this setback, the objectives of the thesis were successfully attained, with the rulers serving as supplementary elements rather than essential components.

The data annotation process provided masks that were successfully matched with the images and provided a crucial component in the development of the model. The tool *Roboflow* was not encountered prior to the completion of fish body annotation, but its utilization would have been preferable during the initial annotation process, given its capability to assist with annotations, thereby enabling faster annotation speed. However, the *VIA* tool provided satisfying annotations but was more time demanding.

## 5.4 The Photobox as a Concept

As mentioned in section 5.3, the lack of a standardized methodology for data collection, characterized by consistent lighting conditions and uniform background, presented inherent challenges during the process of acquiring the dataset. This could, as mentioned as a recommendation in section 5.3, be solved by doing the data acquisition with the photobox, providing a consistent environment. An automated imaging system for the analysis of deceased fish, designed for use in aquaculture, can be deemed as a viable platform for expedited development. The hardware of this photobox can advantageously be made very easily, including a camera as the most advanced feature. This photobox can be placed both on the boat and the deeding barge, preferably on the deeding barge to avoid waves affecting the quality of the images. One reason for the necessity of such a system is the big amount of dead fish in aquaculture every year. There are proven to be no requirements to report the mortality reasons for these fish and manually detecting the mortality reasons is an impossible process due to the significant amount. Gathering information about mortality reasons can give the fish farmers possibilities to start preventive measures, reduce costs and prevent fish suffering. Having numbers on the mortality reasons in aquaculture may also encourage increased research on the diseases and the preventive measures that can be done to avoid them.

It is noteworthy to consider that the system relies on the suffering of certain individuals to save others, notwithstanding the fact that the fish in question has already perished. The literature study provided information about the huge amount of dead individuals in aquaculture, therefore the system can be considered very useful due to many samples of dead fish, providing a lot of information if the photobox is well developed. A notable limitation for fish on a conveyor belt is that one is only able to capture the condition of half the fish, considering the 2D camera. One solution to this is to have two similar photoboxes after each other on a conveyor belt and flip the fish between them, to analyze both sides. There might also be considered to take pictures from above or below and it should be considered if four photoboxes will be the best solution to create the desired outcome. Another aspect to consider regarding welfare is that this system only detects physical problems for the salmon, to best monitor welfare in sea cages, additional monitoring instruments should be applied.

The limitations of the photobox will be determined by the visibility of the diseases. The field trip provided knowledge about how the veterinary does the manual work for disease classification today, stating the fact that some diseases require an autopsy to be detected. For this reason, there will be only a selective amount of diseases that can be analyzed and detected by computer vision. In general, diseases that are easily seen with the human eye can be detected by the photobox. Other diseases will be harder to detect and might potentially require a huge amount of data for a deep learning model to see a pattern that the human eye cannot see. Moreover, the data can be subject to analysis by domain experts to discern the presence of multiple diseases in particular regions and examine the data in conjunction with other factors.

To increase the potential diseases that could be detected with machine learning, one could with great advantage add more automated tools to a dead fish system. It should be fully possible to add some automatic tests that can be added to computer vision analysis such as blood tests. If the photobox gets the dead fish through a conveyor belt, there could be added blood test after the camera has seen the fish to not affect what the camera sees. In vaccination, conveyor belts are used and the process is fully automatic, so the potential to take additional biological tests on dead salmon on a conveyor belt exists.

## **5.5 Software Transferability to the Photobox**

This program has the potential to be integrated into a photobox together with other disease detection programs. The evaluation of the model's performance could be entrusted to individuals possessing the appropriate expertise to assess its suitability for implementation without necessitating additional modifications, ensuring that it adequately provides the requisite information. Nevertheless, the employed methodology demonstrated decent performance and offers good scalability to train the model on even better data and probably improve the performance.

What should be considered if the resulting model should be used in the final photobox is the disparity between the images that this model is trained on and the images captured by the photobox. Factors that can impact the performance of the model are background, lightning conditions and image qual-

ity. It might be recommended to do the data acquisition with the photobox and pursue the procedure provided in chapter 3. The code was structured to support seamless modification and training on new image datasets with minimal changes to the underlying architecture, this enable the program to be used for other images. The procedure presented in chapter 3 can provide a base for other diseases where segmentation is desired.

There is a great possibility of developing the program further such that it can segment and analyze more than one fish in a single image. The requirement for this extension is to train the model on a dataset containing images of more than one fish. The developers should then have in mind that the body segmentations must be able to distinguish the different individuals and place the right ulcer at the corresponding fish. This extension is dependent on how the hardware design is developed, depending on if the conveyor belt passes more than one fish through the camera box or not. Be aware that the FPS rate of the two models must be taken into consideration when deciding the speed of the conveyor belt and thereby how fast the images need to be analyzed. The fish should pass at a speed corresponding to the slowest processing model. However, the FPS will be determined by the computational resources available.

The body segment model would be a great base to extend the diseases the photobox can analyze. A background removal is a natural choice of the first step in all applications.

## Chapter 6

# Conclusion

This thesis has provided a solution for the segmentation of winter ulcers in dead salmon from sea cages in aquaculture. The motivation behind the research was to contribute to the software solution in a photobox which is to be developed by ScaleAQ to perform an automatic assessment of dead fish from sea cages in aquaculture. The current method consists of manually counting and analyzing the fish, but it is desired to automate this process through the photobox using computer vision techniques.

Due to the lack of available data, a dataset was created by data acquisition at a field trip and manual data annotation of the winter ulcers. Methods used for automatic winter ulcer segmentation was identified and U-net were selected to solve the segmentation problem. The U-net model was trained on the dataset to segment the ulcers. The winter ulcer segmentation model yielded a satisfying performance with an IoU of 74,24% and an F1 score of 81,60%, which can equip the aquaculture industry with data features of great interest. Additional functionalities such as background removal were added to the model, which demonstrated the possibilities that can be exploited in future applications. An ulcer factor, the ratio between the infected surface and the body surface, was added as an extra feature to exhibit how additional information can be extracted from the implemented solution.

The methods and results obtained demonstrate that computer vision can be considered a great approach for automatic assessment of fish mortality causes. Semantic segmentation, which was used in this thesis, can be combined with other computer vision techniques, such as object recognition and classification to provide a more complete analysis of the dead fish condition. In conclusion, this work has contributed to the development of the software of the ScaleAQ photobox, addressing a previously unaddressed gap in the existing literature and offering notable benefits to the aquaculture industry.



# References

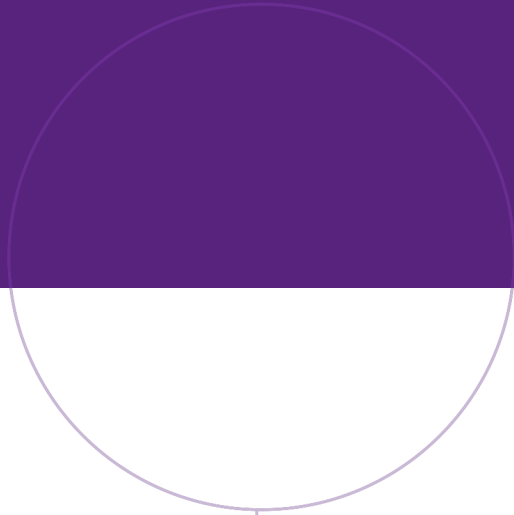
- [1] Martin Føre, Kevin Frank, Tomas Norton, Eirik Svendsen, Jo Arve Alfredsen, Tim Dempster, Harkaitz Eguiraun, Win Watson, Annette Stahl, Leif Magne Sunde, Christian Schellewald, Kristoffer R. Skøien, Morten O. Alver, and Daniel Berckmans. Precision fish farming: A new framework to improve production in aquaculture. *Biosystems Engineering*, 173:176–193, 2018. Advances in the Engineering of Sensor-based Monitoring and Management Systems for Precision Livestock Farming.
- [2] Gorka Merino, Manuel Barange, Julia L. Blanchard, James Harle, Robert Holmes, Icarus Allen, Edward H. Allison, Marie Caroline Badjeck, Nicholas K. Dulvy, Jason Holt, Simon Jennings, Christian Mullon, and Lynda D. Rodwell. Can marine fisheries and aquaculture meet fish demand from a growing human population in a changing climate? *Global Environmental Change*, 22(4):795–806, 2012.
- [3] Rebecca Goldberg and Rosamond Naylor. Future seascapes, fishing, and fish farming. *Frontiers in Ecology and the Environment*, 3(1):21–28, 2005.
- [4] MOWI ASA. *Salmon Farming Industry Handbook*. MOWI, 2021.
- [5] Ellen Sofie Grefsrud, Lasse Berg Andersen, Bjørn Einar Grøsvik, Ørjan Karlsen, Bjørn Olav Kvamme, Pia Kupka Hansen, Vivian Husa, Nina Sandlund, Lars Helge Stien, and Monica E. Solberg. Rapport fra havforskningen 2023/6. *Havforskningsinstituttet*, 2023. Accessed: 2023-05-08.
- [6] Kristoffer Vale Nielsen Martin Haugsmo Iversen Chris Noble Jelena Kolarevic Hilde Frotjold Kathrine Nilsen Eirik Wilkinson Barbo Klakegg Hege Sørvåg Hauge Per Anton Sæther Tore Kristiansen Lars Helge Stien Jonatan Nilsson, Kristine Gismervik. Laksvel - standardisert operasjonell velferdsovervåking for laks i matfiskanlegg. *Havforskningsinstituttet*, 2022.
- [7] Fiskeridirektoratet. Laks, regnbueørret og ørret - matfiskproduksjon. *Fiskeridirektoratet*, 2022.
- [8] Duncan J. Colquhoun and Anne Berit Olsen. Fiskehelse rapporten 2022. *Veterinærinstituttet*, pages 110–112, 2022.
- [9] Yu Huihui Fang Xiaomin Song Lihua Li Daoliang Chen Yingyi Yang Ling, Liu Yeqi. Computer vision models in intelligent aquaculture with emphasis on fish detection and behavior analysis: A review. *Springer Link*, 2021.

- [10] Md Shoaib Ahmed, Tanjim Taharat Aurpa, and Md. Abul Kalam Azad. Fish disease detection using image based machine learning technique in aquaculture. *Journal of King Saud University - Computer and Information Sciences*, 34(8, Part A):5170–5182, 2022.
- [11] William S Noble. What is a support vector machine? *Nature biotechnology*, 24(12):1565–1567, 2006.
- [12] Shaveta Malik, Tapas Kumar, and A. K. Sahoo. Image processing techniques for identification of fish disease. In *2017 IEEE 2nd International Conference on Signal and Image Processing (ICSIP)*, pages 55–59, 2017.
- [13] Roger Sie-Maen Chong. Chapter 50 - epizootic ulcerative syndrome. In Frederick S.B. Kibenge, Bernardo Baldisserotto, and Roger Sie-Maen Chong, editors, *Aquaculture Pathophysiology*, pages 621–627. Academic Press, 2022.
- [14] Ahmed Waleed, Hadeer Medhat, Mariam Esmail, Kareem Osama, Radwa Samy, and Taraggy M Ghanim. Automatic recognition of fish diseases in fish farms. In *2019 14th International Conference on Computer Engineering and Systems (ICCES)*, pages 201–206, 2019.
- [15] Gangyi Yu, Junbo Zhang, Ao Chen, and Rong Wan. Detection and identification of fish skin health status referring to four common diseases based on improved yolov4 model. *Fishes*, 8(4), 2023.
- [16] Hitesh Chakravorty, Rituraj Paul, and Prodipto Das. Image processing technique to detect fish disease. *International Journal of Computer Science and Security (IJCSS)*, 9(2):121–131, 2015.
- [17] Fishwell Techonology AS. Digital telling og rapportering av død fisk basert på videoanalyse gjennom avansert ai/ml. Accessed: 2023-18-05.
- [18] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3523–3542, 2022.
- [19] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.
- [20] Bjarne Kvæstad, Bjørn Henrik Hansen, and Emlyn Davies. Automated morphometrics on microscopy images of atlantic cod larvae using mask r-cnn and classical machine vision techniques. *MethodsX*, 9:101598, 2022.
- [21] Scale AQ. We scale aquaculture. Accessed: 2023-29-05.
- [22] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3523–3542, 2022.
- [23] Jintao Liu Qinfeng Gao Shuanglin Dong Chao Zhou Xinting Yang, Song Zhang. *Reviews in Aquaculture: Deep learning for smart fish farming: applications, opportunities and challenges*. John Wiley Sons Australia, Ltd, 2020.

- [24] Anders Krogh. What are artificial neural networks? *Nature biotechnology*, 26(2):195–197, 2008.
- [25] Chris V. Nocholson. Neural networks definition. <http://wiki.pathmind.com/neural-network#define>. Accessed: March 16, 2023.
- [26] Berndt Müller and Joachim Reinhardt. *Neural networks: an introduction*. Springer-Verlag Berlin Heidelberg GmbH, 1990.
- [27] Dasaradh S K. A gentle introduction to math behind neural networks. *Towards Data Science*, 2020. Accessed: May 13 2023.
- [28] Zack Brodtman. The importance and reasoning behind activation functions. *Towards Data Science*, 2021. Accessed: May 29, 2023.
- [29] C-C Jay Kuo. Understanding convolutional neural networks with a mathematical model. *Journal of Visual Communication and Image Representation*, 41:406–413, 2016.
- [30] Shun ichi Amari. Backpropagation and stochastic gradient descent method. *Neurocomputing*, 5(4):185–196, 1993.
- [31] Mohammad Mahdi Bejani and Mehdi Ghatee. A systematic review on overfitting control in shallow and deep neural networks. *Artificial Intelligence Review*, pages 1–48, 2021.
- [32] Xue Ying. An overview of overfitting and its solutions. In *Journal of physics: Conference series*, volume 1168, page 022022. IOP Publishing, 2019.
- [33] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.
- [34] IBM. Convolutional neural networks. <https://www.ibm.com/topics/convolutional-neural-networks>, Accessed: May 13 2023.
- [35] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, Eftychios Protopapadakis, et al. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018, 2018.
- [36] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*, pages 1–6, 2017.
- [37] Baris Kayalibay, Grady Jensen, and Patrick van der Smagt. Cnn-based segmentation of medical imaging data. *ArXiv*, abs/1701.03056, 2017.
- [38] Yanming Guo, Yu Liu, Theodoros Georgiou, and Michael S Lew. A review of semantic segmentation using deep neural networks. *International journal of multimedia information retrieval*, 7:87–93, 2018.
- [39] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

- [40] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar R. Zaiane, and Martin Jägersand. U<sup>2</sup>-net: Going deeper with nested u-structure for salient object detection. *CoRR*, abs/2005.09007, 2020.
- [41] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.
- [42] Pavitrakumar Kamble. U-net explained: Understanding its image segmentation architecture. *Towards Data Science*, 2018.
- [43] Marianne Stehling. Metrics to evaluate your semantic segmentation model. *Towards Data Science*, 2019.
- [44] Montunrayo Olugbenga. What is balanced accuracy and why you need it. <https://neptune.ai/blog/balanced-accuracy>, 2021. Accessed: May 4, 2023.
- [45] Shruti Jadon. A survey of loss functions for semantic segmentation. In *2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB)*, pages 1–7. IEEE, 2020.
- [46] Pranjal Sexena. Increase frame per second (fpr) rate in the custom object detection step by step. *Towards Data Science*, 2020. Accessed: May 23, 2023.
- [47] Nikhil Tomar. Retina blood vessel segmentation in pytorch. GitHub repository, 2021. Accessed: May 3, 2023.
- [48] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. *ArXiv*, abs/1912.01703, 2019.
- [49] Reza Azad, Ehsan Khodapanah Aghdam, Amelie Rauland, Yiwei Jia, Atlas Haddadi Avval, Afshin Bozorgpour, Sanaz Karimijafarbigloo, Joseph Paul Cohen, Ehsan Adeli, and Dorit Merhof. Medical image segmentation review: The success of u-net, 2022.
- [50] Jeremiah W. Johnson. Adapting mask-rcnn for automatic nucleus segmentation. *ArXiv*, abs/1805.00500, 2018.
- [51] Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. Dice loss for data-imbalanced nlp tasks. *ArXiv*, abs/1911.02855, 2019.
- [52] Yue Zhang, Shijie Liu, Chunlai Li, and Jianyu Wang. Rethinking the dice loss for deep learning lesion segmentation in medical images. *Journal of Shanghai Jiaotong University (Science)*, 26:93–102, 2021.
- [53] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

- [54] Nahian Siddique, Sidike Paheding, Colin P. Elkin, and Vijay Devabhaktuni. U-net and its variants for medical image segmentation: A review of theory and applications. *IEEE Access*, 9:82031–82057, 2021.
- [55] Simon Graham, Quoc Dang Vu, Mostafa Jahanifar, Shan E Ahmed Raza, Fayyaz Minhas, David Snead, and Nasir Rajpoot. One model is all you need: Multi-task learning enables simultaneous histology image segmentation and classification, 2022.
- [56] Zhuoxun He, Lingxi Xie, Xin Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Data augmentation revisited: Rethinking the distribution gap between clean and augmented data. *ArXiv*, abs/1909.09148, 2019.



Norwegian University of  
Science and Technology