Master's thesis

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

Herman Stavelin

Supervised Classification Of Unlabeled Acoustic Data Utilizing Cross-Referencing With Labeled Images

Master's thesis in Cybernetics and Robotics
Supervisor: Adil Rasheed
July 2020

NTNU
Norwegian University of Science and Technology

Herman Stavelin

Supervised Classification Of Unlabeled Acoustic Data Utilizing Cross-Referencing With Labeled Images

Master's thesis in Cybernetics and Robotics
Supervisor: Adil Rasheed
July 2020

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

NTNU
Norwegian University of
Science and Technology

ABSTRACT

With the increased focus on man made changes to our planet and wildlife, more and more emphasis is put on sustainable and responsible gathering of resources. In an effort to preserve marine wildlife, the Norwegian government has proclaimed a necessity for creating ecological maps, detailing the presence and amount of wildlife species in Norwegian fjords and oceans.

To this end, a submerged sonar system has been deployed in the Oslo Fjord, gathering vast amounts of marine data. Procuring labeled acoustic data is time consuming and expensive, and analysis is predominantly based on ad hoc mathematical methods that are difficult to verify. It is of interest to determine if a more cost effective labeling procedure can be devised, and if the recent breakthroughs within Machine Learning (ML) enables improvements within classification, compared to classical mathematical methods.

In this thesis the author demonstrates techniques for acquiring and analysing marine data. A procedure for interweaving optic and acoustic data is developed and its validity demonstrated empirically. It is shown that the two data sources can be sufficiently related, spatially and temporally, yielding a rich dataset capable of harnessing the individual strengths of each data source. Deep learning techniques are employed and a Neural Network (NN) is developed and trained on opti-acoustic data. The results show that supervised classification of unlabeled acoustic data can be performed, utilizing cross-referencing with labeled optic data. The methods were able to correctly classify the presence of fish with an accuracy of 64.8 %, demonstrating a proof of concept.

SAMMENDRAG

Med det økende fokuset på menneskeskapte endringer, settes mer og mer trykk på gjennvinbar og ansvarlig innhøsting av ressurser. I et forsøk på å bevare marint liv har den norske regjeringen bestemt at det må lages økologiske kart, som beskriver posisjon og mengde av viltlivsarter i norske farvann.

For å oppnå dette er et sonarsystem blitt utplassert i oslofjorden, for innsamling av store mengder marin data. Å "lable" akustisk data er tidkrevende og dyrt, og analyse er hovedsakelig basert på ad hoc matematiske metoder som er vanskelige å verifisere. Det er av interesse å finne mer kostnadseffektive metoder for å "lable" data, samt om nye gjennombrudd innen Maskinlæring kan forbedre klassifisering, sammenlignet med klassiske matematiske metoder.

I denne oppgaven demonstrerer forfatteren teknikker for innhøsting og analyse av marin data. En prosedyre for sammenkobling av optisk og akustisk data er utviklet og dens gyldighet demonstrert empirisk. Det er vist at de to datakildene kan tilstrekkelig relateres, både spatialt og temporalt. Resultatet er et rikt datasett, som er i stand til å utnytte de individuelle styrkene til hver datakilde. Teknikker innenfor dyp læring er benyttet og et nevralt nettverk (NN) er utviklet og trent på opti-akustiske data. Dette viser at overvåket klassifisering av "unlabeled" akustisk data kan gjennomføres ved hjelp av kryssreferering med "lablet" optisk data. Metodene var i stand til å korrekt klassifisere tilstedeværelsen av fisk med en nøyaktighet på 64.8 % og regnes som et gjennomførbarhetsbevis.

PREFACE

This Master's thesis was conducted during the spring semester of 2020 at the Norwegian University of Science and Technology (NTNU) in Trondheim, Norway. The work continuous the pre-project carried out during the autumn semester of 2019 as a part of the course *TTK4550 - Specialization Project* in Engineering Cybernetics as well as the authors published scientific paper [51]. The authors chosen specialization is within Robot systems. Both the pre-project and the Master's thesis are written in conjunction with Kongsberg Maritime.

The code used in, and written for, this thesis is available on the author's personal github repository; fishynet [50].

For training and testing a remote computer supplied by Kongsberg Maritime was utilized. The computer is equipped with an Intel i7-7700K max clocked at 4.5GHz, and an NVIDIA GeForce GTX 1080 Ti max clocked at 1911MHz, running Ubuntu 18.04.

The author wishes to thank Kongsberg Maritime for their assistance in providing him with the necessary tools and data, making this thesis a possibility. Especially, he extends his thanks to Arne Johan Hestnes, Per Ove Husøy and Frank Reier Knudsen for their helpfulness and support. Finally, he wishes to thank his friends and family for their relentless support. Due to the ongoing pandemic caused by COVID-19, the author has spent record time inside, and consequently his roommates, and dear friends, has been more important than ever.

- Herman Stavelin

TABLE OF CONTENTS

LIST OF FIGURES

ACRONYMS

AI Artificial Intelligence. 1

FCN Fully Convolutional Network. 11

FN False Negative. 18

FOV Field Of View. 6, 8, 11, 23, 25, 31, 34, 43

FP False Positive. 18

mAP mean Average Precision. 11, 33

ML Machine Learning. i, 2–4, 18, 41

NN Neural Network. i, 3–5, 15, 16, 20, 30–32, 35, 38, 40–43

PCA Principal Component Analysis. 5, 19, 30, 31, 37, 38

SNR Signal to Noise Ratio. 8

TN True Negative. 18, 30, 34, 38

TP True Positive. 18, 30, 34, 42

TS Target Strength. 8

TVG Time Variable Gain. 10, 11

WSL Windows Subsystem for Linux. 22

YOLO You Only Look Once. 11, 12, 33

CHAPTER 1

INTRODUCTION

Since the rise of mankind the biodiversity on Earth has gradually diminished [45]. As the human species has continued to grow, so has our needs for land and resources. The growth is exponential and it has been confirmed numerous times that the extinction rate is at an all time high in modern times. Some researches even proclaim that the earth currently is in a mass extinction spasm [9]. While scientists dispute whether the earth actually is on its sixth major ext inction event, the consensus is clear on that species are disappearing at an unprecedented rate [6, 30]. If future generations shall be able to survive on this planet, then our expansion must be conducted in a sustainable manner.

The more humans populate the Earth, the more food must be procured. Fish is an absolute necessity for the survival of humanity. As more and more fish are extracted from the oceans, it is uncertain if the fish populations are able to endure and persist. By today's standards it is unclear if there will be enough fish to feed the world by 2050, according to the World Wildlife Fund (WWF). They claim that it is necessary to enforce a global management system in order to ensure sustainable fishing. For this to be a feasible task, much more must be known about the ecological status and inner workings of the various ecosystems that surround us.

1.1 Background and Motivation

For each passing year growing quantities of marine life disappear from the Norwegian fjords and oceans. In order to combat this development, the Norwegian government has launched a project called Frisk Oslofjord - Healthy Oslo Fjord [17]. The project has in its statuettes to enable green businesses to thrive and to maintain healthy coastal cultures. To this end, it is vital to collect knowledge that will fortify the foundation of sustainable management, and improve the health of marine resources and environment.

One of the main goals of the Healthy Oslo Fjord project is to prepare detailed ecological maps of the Oslo fjord in particular, but all Norwegian fjords and oceans are of interest. These maps are expected to show the class of present marine species and their locations at any particular time. By fulfilling this goal, significant insight will be acquired, explaining the behaviour of marine organisms in the Norwegian fjords and oceans. This knowledge can be used to improve the sustainability of marine harvesting.

Presently, the mapping procedure is conducted manually by inspecting images, and then recording the findings. However, with the recent success of Artificial Intelligence (AI)

and ML in image classification, text interpretation and big data analysis, new possibilities are opening up to address relevant questions. For example, in [37], [3], and [60], scientists have already shown the power of computer vision and ML, not only in identifying, but also classifying various marine species. The approach, owing to the ease of automation, will allow mapping of the fjords and oceans in general, with much higher spatio-temporal resolutions as well as reducing the manual labor required and enabling sustainable marine harvesting.

Object detection on visual data is currently a very active field [31]. Algorithms regarding object detection under water has also been successfully implemented. However, despite the huge potential of exploitation of the ML based approach, the technology is not perfect [4, 55]. We refer to [34] for a survey of deep learning methods on underwater marine object detection and automated approaches for monitoring of underwater ecosystem including seagrass meadows. The algorithms which can give super-human performance in image classification in good daylight might suffer to make correct classifications in underwater scenarios where the visibility is highly diminished due to poor light conditions. Furthermore, the camera has severely limited range underwater. These limitations can be countered through the use of acoustic transducers.

Classification utilizing acoustic transducers has been done for a long time [62], but more often than not, they rely on ad hoc mathematical methods that are difficult to verify. Echograms are traditionally analyzed using statistical characteristics of the aggregations of organisms. Feature-based classification methods usually favor a classical machine learning paradigm and utilize hand-crafted features. Deep learning, which has been shown to be very effective at various tasks in computer vision such as object detection and recognition, has yet to permeate echogram analysis [44].

With the rise of the ML-paradigm within computer vision, as well as in general, this thesis will explore solutions to underwater object detection within this paradigm. Employing ML-techniques with acoustic data in order to detect underwater objects has been done several times on sonar images [8, 18, 23, 38, 39, 46, 47]. This paradigm has shown promise in classification of schools, individual fish and seabed [18], discerning between rocks and mines [47], identification of herring [44], etc. However, such sonar systems are quite expensive and thus not always eligible. In this thesis a more tractable sensor system with a focused split-beam sonar system with chirp capabilities, recording echos in a constant limited volume, is investigated and utilized. *This type of acoustic data has not been significantly researched, within the ML-milieu*. However, it has been shown that discerning between sticklebacks and whitefish can be achieved with such a setup, using random decision forests on frequencies in the 90-170kHz range [59].

A significant challenge, when classifying acoustic data, is obtaining labels. Both for training supervised algorithms and for verifying standard mathematical methods. Obtaining labels for acoustic data are done by either, manually labeling data as it is procured, in a controlled environment [59] (for example by using fishing nets), or, by extracting portions of echograms that, based on ad hoc methods and empiricism, are believed to be fish [8]. Since acoustic data can not always be gathered in such controlled environments, and labeling of portions of echograms is not necessarily completely sound, an alternative approach is desired. Therefore, focus will be on procuring labels for acoustic data, by cross-referencing with labeled optical data. Thus, creating a multi-sensor dataset, con-

taining opti-acoustic data, that enables the possibility for supervised algorithms to train on unlabeled acoustic data. This way, labeling can be done after measurements are completed and perform in real-time. *Utilizing opti-acoustic data in a supervised ML context has not been extensively researched.*

1.2 Research Questions and Tasks

The **primary research question** this thesis seeks to answer:

> *Is it feasible to employ deep learning to identify fish, utilizing unlabeled acoustic data in conjunction with labeled optical data?*

In order to satisfactorily answer this, the following **partial research questions** must be answered:

RQ1: *Does the acoustic data accommodate sufficient patterns for classification to be a possibility?*

RQ2: *How can ML be used to extract patterns from acoustic data in real-time?*

RQ3: *How can optical and acoustic data be utilized in conjunction, to aid in automatic detection and classification?*

The most important *contributions* embodied in this thesis, thereby realizing the aforementioned research questions, are the following **research tasks**:

RT1: *Creating tools for extracting acoustic data.*

RT2: *Devising a geometric relationship between optical and acoustic data.*

RT3: *Generating a labeled dataset with opti-acoustic data.*

RT4: *Demonstrating empirically, that optical and acoustic data can be combined to aid classification.*

RT5: *Designing and implementing a NN capable of discerning between the presence and absence of fish, utilizing opti-acoustic data.*

1.3 Thesis Outline

The thesis is divided into five main chapters:

The first chapter contains a general introduction to the topics that will be discussed in the thesis, including research questions and tasks.

In chapter 2 all the necessary background theory, required for the later chapters of the thesis, is presented. There are four main sections. In the first section it is shown how sonar systems work, after which, the procurement of optical data is undergone. Then a brief

overview of NNs is presented. In the final section ubiquitous metrics for ML and relevant statistical methods are explained.

In chapter 3 all methods and techniques that were developed and used in this thesis are presented. The procedures for acquisition and extraction of acoustic data are presented. The geometric relationship between the optical and acoustic sensors is developed and a scheme for empirically verifying the opti-acoustic relationship is designed. An algorithm for utilizing opti-acoustic data is developed.

In chapter 4 all the results from the previous chapter are presented and discussed. The results obtained from the data acquisition and extraction, opti-acoustic derivation and validation, and the performance of the NN utilizing opti-acoustic data, are presented and discussed.

Finally, in chapter 5 the entire thesis is briefly summarised and the most important findings are ascertained. All research questions and tasks are evaluated. At the very end potential future work is presented and discussed, as well as the potential ramifications of the conducted work contained within the thesis.

CHAPTER 2

THEORY

The topics presented in this chapter:

- Basic background theory on sonars.
- The acquisition of optical data.
- Background theory on NNs.
- Ubiquitous metrics and theory behind Principal Component Analysis (PCA).

2.1 Sonar

An active [1] SONAR (SOund Navigation And Ranging) is a device that is capable of emitting and recording acoustic waves that can be used to detect and locate objects. A sonar consists of a transducer, a transmitter and associated electronics such as amplifiers and data acquisition systems [54]. A transducer is the combination of a microphone and a loudspeaker all in one [21]. The transducer is the element responsible for converting electrical signals to sound waves and vice versa. The transmitter is the element that generates the waveforms that the transducer emits.

The transducers consists of one or more elements that vibrate when applied an electrical signal. These vibrations generate an acoustic wave, usually referred to as a pulse. The acoustic wave expands as a spherical wave in a homogeneous medium [22]. The wave propagates through the water column and is partially reflected when observing an impedance difference, as illustrated in Figure 2.1. The reflections, which are referred to as echos, are continuously recorded by the transducer. This process repeats over and over.

There exists numerous varieties of transducers. For the remainder of the text, split-beam transducers will be the only type discussed. This type of transducer normally has three or four elements that are capable of recording echos. These elements are partitioned in distinct, geometrically symmetric sectors, such that the angle of the incoming echo can be determined by utilizing the geometrical spacing of the listening elements.

[1] Passive sonars are not within our interest and thus neglected

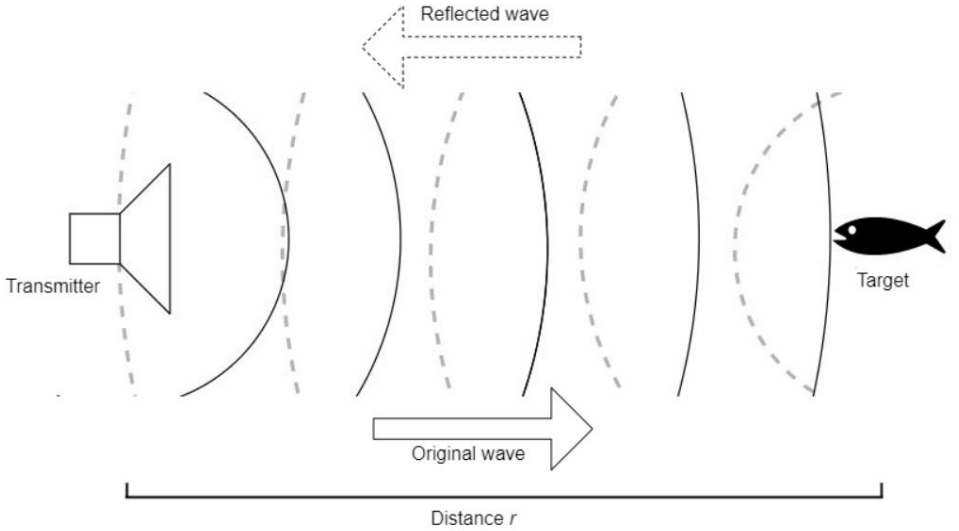

Figure 2.1: The basic principle behind any sonar system.

From the measured time it takes for a signal to be propagated back to the transducer, and the speed of sound of the medium the signal is traversing, the distance to the reflecting object can be calculated:

$$R = \frac{c\tau}{2} \tag{2.1}$$

Even though the main goal of the transducer is to record information about objects of interest, there are necessarily unwanted signals present. The echo, which the sonar receives, mainly contains three different types of information:

1. The reflected signal from a target.
2. Reverberation, which is unwanted echo typically caused by echos from the surface, bottom and volume scattering.
3. Additive noise, which are acoustic signals emitted by something else than the sonar.

An illustration of the Field Of View (FOV) of the transducers as well as how it interprets the presence of objects within its FOV is presented in Figure 2.2. Objects, for example fish, at different depths will reflect the transducers waves, and the back-scattered echos are recorded. Fish outside the beam will naturally not be observed at all. In the case of two or more fish at the same distance from the transducer, the same result is observed in the echogram, but due to angular information extracted by the transducer, the targets can be discerned. The amplitude of the pings are what is seen on the right hand side of Figure 2.2.

The acoustic frequencies used in sonar systems vary from very low (infrasonic) to extremely high (ultrasonic) [36]. High frequency sonar systems naturally produce better

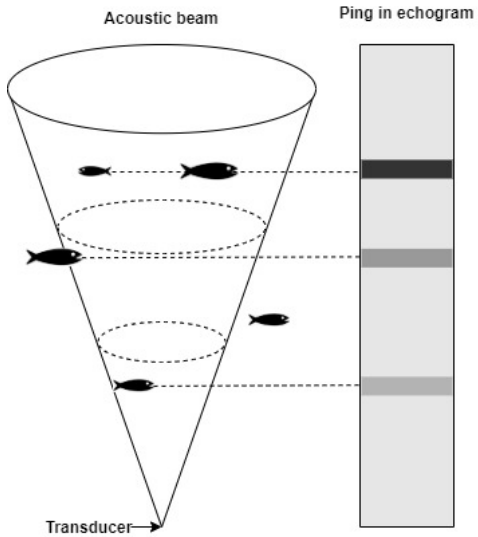

Figure 2.2: The acoustic beam and the obtained echogram for one ping.

range resolution, but the waves carry less energy which leads to shorter propagation range [22]. A remedy to alleviate this trade off is sweep transmissions also known as CHIRP (Compressed High Intensity Radar Pulse). Instead of sending a single beam at a single frequency, a system using chirp send pulses at many frequencies simultaneously. Chirp is superior at target differentiation since different frequencies carry distinct information due to the difference in reflection at various frequencies. The response from a chirp transmission is displayed in Figure 2.3.

2.1.1 The Sonar Equation

The active sonar system can be summarized by the active sonar equation. This equation ties together all the various aspects of the sonar system, including the effects of the medium, the target, and the equipment [41]. There are three underlying assumptions behind the active sonar equation [5]:

1. Single targets are point sources.
2. Waves hitting the target are plane waves.
3. Sound spread out in a spherical manner.

The equation is given by

$$SNR = SL - 2TL + TS - NL \tag{2.2}$$

where SNR is the Signal to Noise Ratio, SL is the Source/Sound Level, TL the transmission loss, TS the target strength, and NL is the noise level. The transducer produces

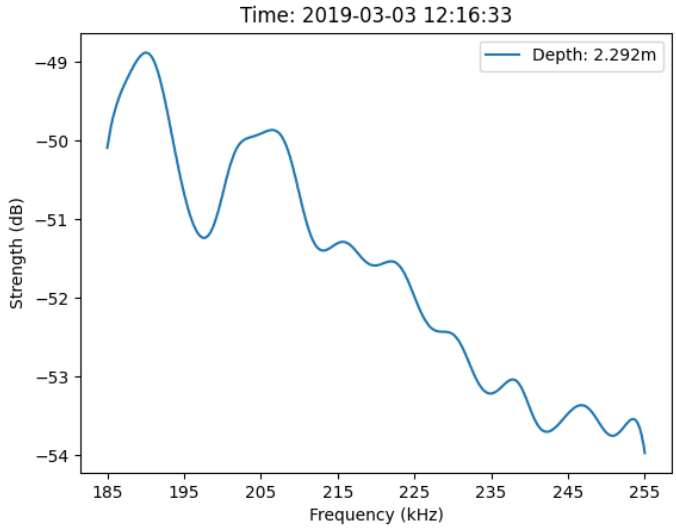

Figure 2.3: An example of a chirp response.

the SL. The sound intensity is reduced due to transmission loss TL before it hits a target, yielding a Target Strength (TS), or the volume of water V reflecting S_v. Then, the sound is reflected back to the transducer, losing as much energy as on the way from the transducer [5]. Note that the Signal to Noise Ratio (SNR) will increase with increasing TS.

The TS is a measure of the acoustic scattering of a target. This is often called the acoustic area or reflection area. Formally, TS is defined as:

$$TS = 10\log_{10}\left(\frac{I_r}{I_i}\right) = 10\log_{10}\frac{\sigma_s}{4\pi} = EL - 2TL - SL \tag{2.3}$$

where I_r is the acoustic intensity of the scattered wave from the target, I_i is the acoustic intensity of the incident plane wave measured at a unit distance, and EL is the Echo Level.

A variable of great importance is S_a which is a measure of the areal backscattering. Ensuing is a description of the derivation of S_a. The formulations presented relies on [12, 13, 28, 32] which in turn largely relies on the bedrock *Acoustical Oceanography* by Medwin and Clay [11]. Initially, three further assumptions are made:

1. The scattered echos from different object in the sonars FOV have random phases.

2. Multiple scattering effects and interaction between object can be neglected.

3. Excess attenuation from power extinction caused by volume scattering in the sonars FOV can be neglected.

Assumption 1 corresponds to random spacing of objects in one *ping*, and movement of the objects to the next *ping*. Assumption 2 means that only echos backscattered directly from the objects are significant, so that those backscattered via other objects (second-order effects) can be ignored. Assumption 3 may be a reasonable approximation, except for strong scatterers at high densities, distributed over an extended volume.

For a multitude of small objects in a sampled volume, the echos from individual objects cannot be resolved, but combine to form a received signal with varying amplitude. Under the above assumptions the total echo intensity is the incoherent sum of the individual echo intensities. The volume backscattering coefficient S_v is the backscattering cross section per unit volume. Consequently, the volume backscattering coefficient can be calculated as a sum over backscattering cross sections per unit volume.

$$S_v = \lim_{\Delta V \to 0} \left(\sum_{j=1}^{N} N_j \sigma_{bs,j} \right) = \lim_{\Delta V \to 0} \left(\frac{1}{\Delta V} \sum_{j=1}^{N} m_j \sigma_{bs,j} \right) \tag{2.4}$$

where N is the number of scattering object types, $N_j = \frac{m_j}{\Delta V}$ is the number of scattering objects of type j per volume ΔV, m_j is the number of scattering objects of type j in the volume ΔV, and $\sigma_{bs,j}$ is the backscattering cross section for an object of type $j, j = 1, ..., N$. From Equation 2.4, $m_j \sigma_{bs,j}$ represents the total backscattering cross section for scatterers of type j, in the volume ΔV. Consequently,

$$\Delta \sigma_{bs} = \sum_{j=1}^{N} m_j \sigma_{bs,j} \tag{2.5}$$

represents the total backscattering cross section over all scatterer types, in the volume ΔV. From the two preceding equations it follows that $S_v = \lim_{\Delta V \to 0} \frac{\Delta \sigma_{bs}}{\Delta V} = \frac{d\sigma_{bs}}{dV}$, such that

$$d\sigma_{bs} = S_v dV \tag{2.6}$$

From Equation 2.5 it is seen that $d\sigma_{bs}$ represents backscattering from a multitude of objects in the unit volume dV, including objects of different types, and objects of the same type with different sizes.

For brevity it is assumed known that the transmit-receive electrical power transfer function is given by:

$$\frac{\Pi_R}{\Pi_T} = F_\Pi G^2(\theta, \varphi) \frac{\lambda^2 e^{-4\alpha r}}{(4\pi)^2 r^4} \sigma_{bs} \tag{2.7}$$

where F_Π is the electrical impedance factor, G the axial transducer gain and α is the acoustic absorption coefficient of the medium. For a complete derivation see [32].

Equation 2.7 applies both to single scattering objects in the far field, as well as to a multitude of far-field objects of different types, materials and sizes confined to a sufficiently small volume in space, so that the backscatter at the transducer appears as coming from a single point in the far field. For backscattering from the small unit volume dV in V_{obs} we get from Equation 2.7 that

$$d\Pi_R = \Pi_T F_\Pi G^2(\theta, \varphi) \frac{\lambda^2 e^{-4\alpha r}}{(4\pi)^2 r^4} d\sigma_{bs} \tag{2.8}$$

is the change in received electrical power.

To progress, it is further assumed that the scattering of objects within the volume V_{obs} is uniformly distributed, so that $d\sigma_{bs}$ can be used everywhere in V_{obs}, meaning that backscatter is essentially the same for objects anywhere in the transducer beam. Integration of Equation 2.8 over this volume, and substituting in Equation 2.6 produces

$$\Pi_R = \int_{V_{obs}} \Pi_T F_\Pi G^2(\theta,\varphi) \frac{\lambda^2 e^{-4\alpha r}}{(4\pi)^2 r^4} S_v dV \tag{2.9}$$

where $dV = r^2 dr d\Omega$.

The present continuous-wave analysis also applies to the steady-state portion of transient signals. Assume the observation volume V_{obs} in the far field is insonified using a tone burst of time duration τ_p and angular carrier frequency ω. The spatial extension of the pulse is $c_0\tau_p$. Assume $c_0\tau_p << r_{max} - r_{min}$. Within the spherical shell volume V_{obs}, the tone burst will then cover a spherical shell subvolume, V_p, contained within ranges r_x and r_y. Consider backscatter from V_p. At the transducer, the arrival times of the start and stop of the tone burst are $\frac{2r_x}{c_0}$ and $\frac{2r_y}{c_0}$, respectively. By defining $dr_p = r_x - r_y$ as the thickness of the spherical shell volume V_p, one gets $dr_p = \frac{1}{2}c_0\tau_p$. Consequently, $dV = \frac{1}{2}c_0\tau_p r^2 d\Omega$. Substitution of this expression into Equation 2.9 yields

$$\frac{\Pi_R}{\Pi_T} = F_\Pi \frac{\lambda^2 e^{-4\alpha r} c_0\tau_p}{(4\pi)^2 r^4 2} S_v \int_{4\pi} G^2(\theta,\varphi) d\Omega \tag{2.10}$$

By solving the above equation with respect to S_v the following formula is obtained

$$S_v = \frac{32\pi^2 r^2 e^{4ar} \Pi_R}{G_0^2 \psi \lambda^2 c_0 \tau_p F_\Pi \Pi_T} \tag{2.11}$$

where

$$\psi = \frac{1}{G_0^2} \int_{4\pi} G^2(\theta,\varphi) d\Omega \tag{2.12}$$

$$G_0 = G(0,0))\nu \tag{2.13}$$

are the equivalent two-way solid beam angle of the transducer and the axial transducer gain, respectively.

The volume backscattering from the finite spherical shell volume V_{obs}, between ranges r_{min} and r_{max}, is obtained by measuring S_v for a continuous sequence of gated volumes, V_g, and integrating S_v over the range of these gated volumes, giving the area backscattering coefficient

$$S_a \equiv \int_{r_{min}}^{r_{max}} S_v dr \tag{2.14}$$

This represents the backscattering cross section per unit area (dimensionless), within V_{obs}.

Time Variable Gain (TVG) is a way of automatically having the unit adjust the gain selectively based on how long it takes the ping to return. For the pings that take the longest to return it adds a gain to them, before displaying them, and it may reduce the gain a bit for

the pings that take the shortest to return. There are primarily two different types of TVG: TVG 40 log R and TVG 20 log R. The former is normally used to detect individual fish while the latter is used to detect schools of fish.

As objects move away from the center of the acoustic beam, their echos become weaker. Maximum Gain Compensation amplifies the signal coming from targets that are located off-center of the acoustic beam. Thus, with an increase in gain compensation, the perceived FOV of the transducer increases.

It is natural that the same object is identified in several sequential echos, due to their inherent movement speed. Maximum Phase Deviation removes all pings that have too large phase difference between sequential samples. Thus, if the phase deviation is low, echos will be filtered away.

2.2 Optical data

The main goal of this section is to show the acquisition and labeling of the optical data that will be utilized throughout this thesis. In the pre-project and related paper [51] a method for auto-labeling images, utilizing the bare minimum of manual labeling, was conducted. Almost 100.000 images were classified with a mean Average Precision (mAP) of approximately 0.88, utilizing the third version of You Only Look Once (YOLO) [43]. A brief overview of the algorithm, labeling, and structure of the optical data will now be presented.

The YOLO algorithm is one of the most efficient and accurate algorithms for object detection in complicated scenes [43]. So far, the algorithm has been adopted in many applications including chemical sensing and detection of gas emission [35], anthracnose lesion detection on plant surfaces [53], small target detection from drones [61], traffic monitoring [7], plate recognition [29], pedestrian detection [40], and autonomous driving applications [10].YOLO is a Fully Convolutional Network (FCN) [43]. It uses a feature extractor with residual blocks consisting of 53 convolutional layers. One unique feature of this algorithm is that the detections are conducted at different depths throughout the network.

In Figure 2.4 the entire structure of the network is shown. On the far left of the network one can see the layer through which the input images are fed in. This is followed by a gray box indicating YOLO's feature extractor. The feature extractor, as the name implies, is responsible for extracting features from the input. It consists of 23 residual blocks, each of which are built up of convolutional layers with 3×3 and 1×1 kernels. Batch normalization is applied in every convolutional layer to regularize the model, thus avoiding overfitting without the invocation of dropout [42]. 3×3 kernels with stride 2 are used when downsampling the feature map. YOLO uses no form of pooling in contrast to most other FCNs [63]. This is because pooling is often attributed to loss of low-level features [26].

Since YOLO is a FCN, it is invariant to the size of the input images. However, for mere convenience (for example in batch processing of images and parallelization on GPUs), the dimensions of all the images are kept the same. Detections are made at layer 82, layer 94 and layer 106. By the time the input image transverse down to the first detection layer, its size shrinks by a factor of 32. Thus with an input image of size 416×416 the feature map

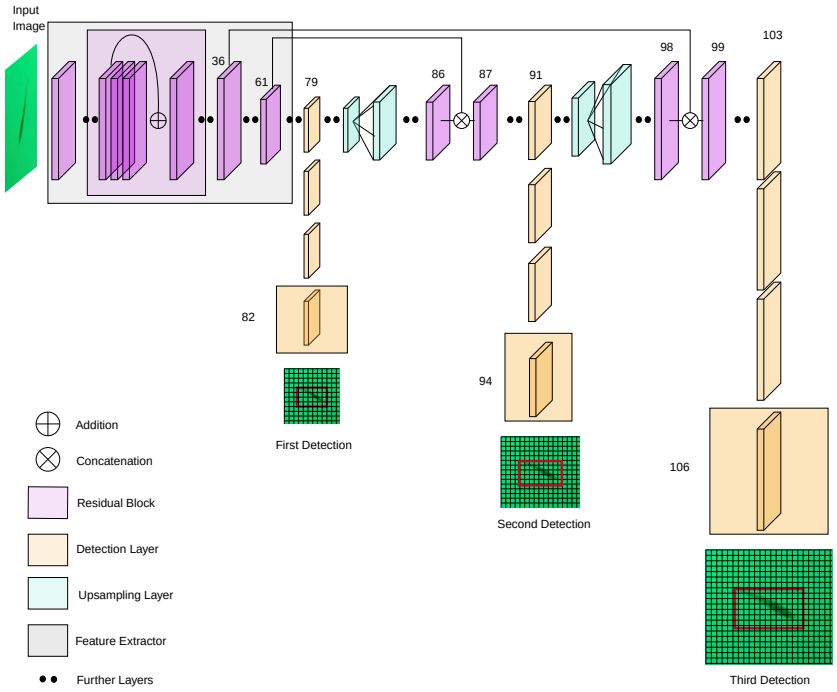

Figure 2.4: The structure of the entire YOLO v3 network.

at this layer will be 13×13. After the first detection, the layer prior to the detection is upsampled by a factor of 2. In the figure this corresponds to taking the last purple layer before the first orange layer. After a few more convolutional layers the current layer is concatenated with a feature map from an earlier layer having identical size. In Figure 2.4 this is shown as concatenation and we see that layer 61 and 86 are concatenated to produce layer 87. Then, at layer 94, YOLO again extracts detections. The exact same procedure repeats once more. If the input image was 416×416, the feature maps in layer 94 and 106 would be of size 26 and 52, respectively. Extraction of detections at three locations is an added feature of the third version of YOLO. According to the authors of YOLO it improves the detection of small objects since it is able to capture more fine-grained features [43]. The output of the network is formulated as a 3D tensor and its dimensions are presented in Equation 2.15.

$$\text{Output} = S \times S \times [B * (5 + C)] \tag{2.15}$$

where S is the number of grid-cells, B the bounding boxes per grid cell and C the number of classes to detect. In Figure 2.5 we see an illustration of a feature map in a detection layer. A bounding box is displayed as a red rectangle and the orange square is the grid cell that is at the center of the bounding box. This cell contains a long row of values. (t_x, t_y) are the center of the box relative to the bounds of the grid cell the box belongs to. (t_w, t_h) are the width and height of the box relative to the whole image.

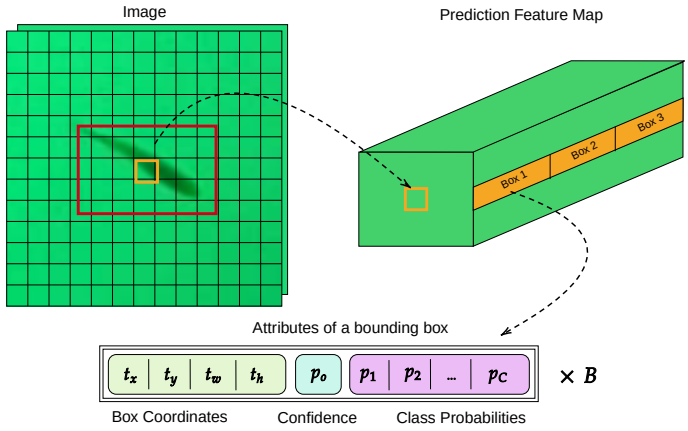

Figure 2.5: Explanation of YOLOs output tensor.

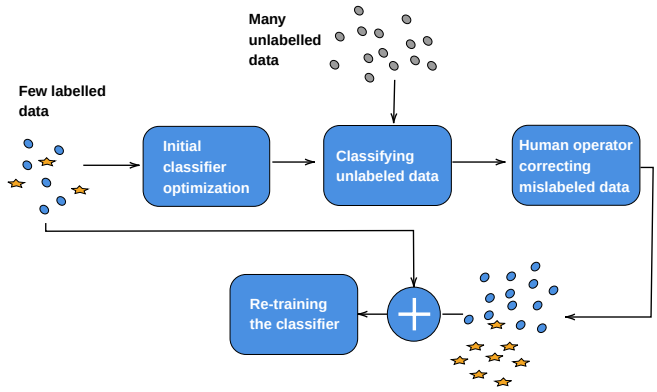

Figure 2.6: The training process.

Utilizing the network just described, pseudo-labeling of images was performed. The training of the classifier was performed in two steps in the manner indicated by Figure 2.6. In the first stage 500 hand-labeled images were fed to the network and the network was trained. Using this trained network, 2500 novel unlabeled images where fed into the network and classified. Then any deviations in these newly, automatically labeled images where manually corrected and fed back into the network. The network was then retrained, initialized with the weights from the previous training session. With the, now fully-trained network, the rest of the dataset, consisting of nearly 100.000 images, were classified automatically. Some examples of optical data, with automatically generated labels, are shown in Figure 2.7.

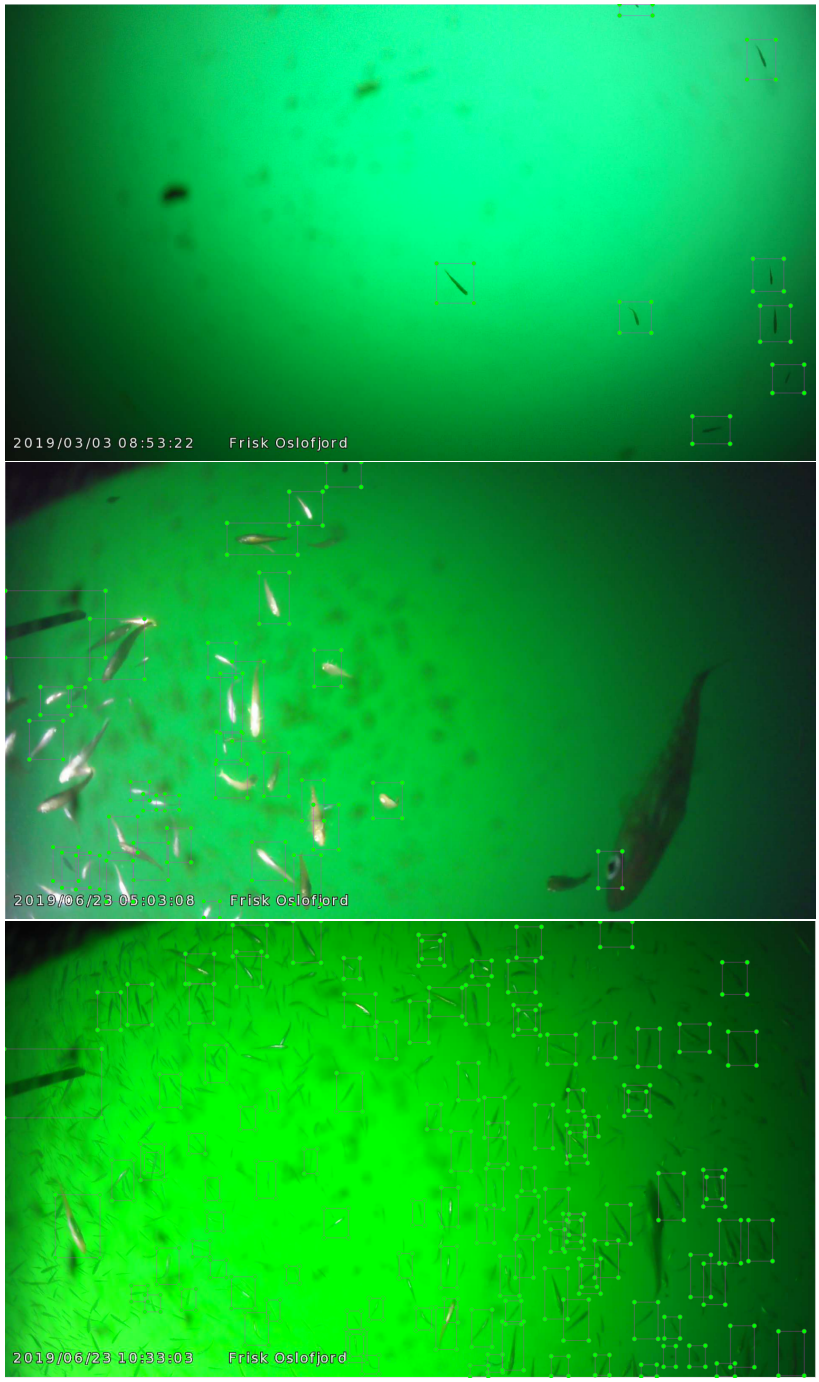

Figure 2.7: Some examples of labeled optical data.

2.3 Feedforward Neural Networks

Deep feedforward networks, also called feedforward neural networks, or multilayer perceptrons (MLPs), are networks capable of approximating some function f that maps an input $\boldsymbol{x}$ to a desired output $\boldsymbol{y}$ [19]. A feedforward network defines the mapping $\boldsymbol{y} = f(\boldsymbol{x}; \boldsymbol{\theta})$ and learns the values of the parameters $\boldsymbol{\theta}$ that produce the best function approximation. NNs are loosely based on the biological neural networks found in brains [58].

A network consists of neurons connected together in an acyclic directed manner as shown in Figure 2.8, where it is implicit that information flows from left to right. The neurons are also grouped together in layers. Each neuron in a network is in itself a function

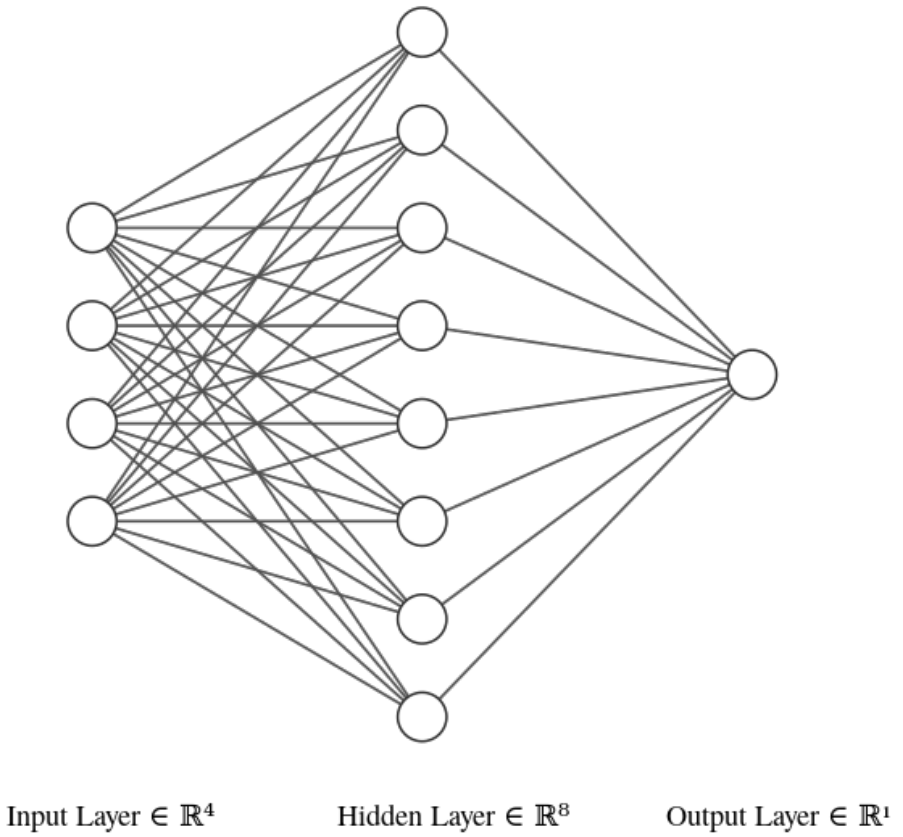

Figure 2.8: A standard neural network.

that calculates its output using $Y = g\left(\sum wx + b\right)$, where g is some activation function. A visualization is shown in Figure 2.9. The neuron takes some inputs x_i that are weighted by the weights w_i and added together with a bias b, and then processed by an activation function.

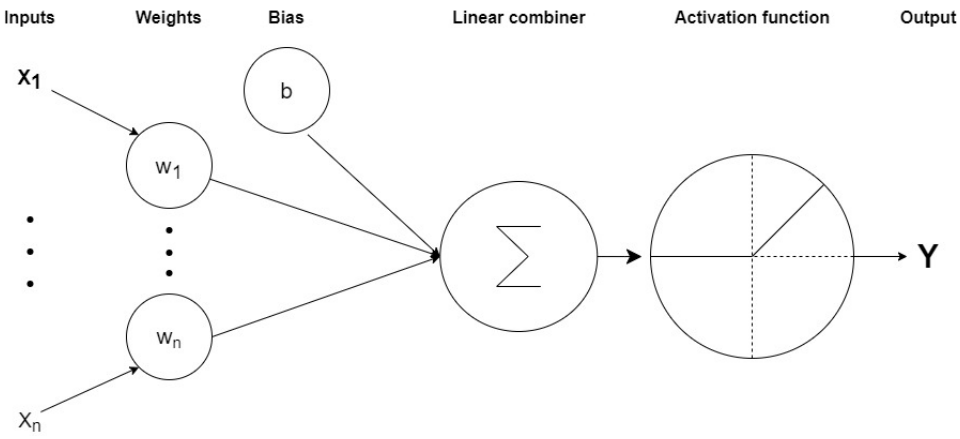

Figure 2.9: A standard neuron.

Activation functions The most commonly used activation functions are Sigmoid, Tanh and ReLU [19]. They are all nonlinear. Common problems with Sigmoid and Tanh are vanishing gradients. Tanh has a much steeper gradient than Sigmoid. ReLU has the advantage of giving sparsity to the output due to its horizontal line [58]. ReLU is also less computationally expensive to compute. The most common problem with ReLU is dying ReLU, which occurs when the activations of a neuron is 0. When this happens the neuron weights will never updddate because the gradient calculated from 0 is 0. This occurs for all negative inputs to the ReLU function. Variations of ReLU, like Leaky ReLU, seek to combat this problem.

Layers A layer in a network consists of many neurons grouped together. Each neuron is modeled as described above, with its own activation function, weights, and bias. In a layer topology, the inputs to each individual neuron is the output of the entire previous layer.

Loss The loss function, also called the cost function, is used to calculate the error of predictions made by the network. The loss used for the system proposed in this thesis, is binary cross-entropy. This function calculates the binary cross-entropy between the training data and the model distribution. The function is given by:

$$L_p(\theta) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)) \tag{2.16}$$

where y is a label and $p(y)$ is the predicted probability for that class.

Regularizers A common problem when training is overfitting. Overfitting occurs when the network memorizes the training set. There are some common clues to look for, in order to see whether the NN is overfitted. Mainly, looking at the difference in training set prediction accuracy and validation set prediction accuracy. If the accuracy on the training set is

high while the accuracy on the validation set is low, the network is overfitted. Regularizers exist to combat overfitting. Two examples:

- **Dropout** is a simple method that can be used to avoid overfitting. Dropout turns neurons on or off in a layer by a probability p, meaning the output value of the neuron is set to 0. The probability is often user defined.

- **Batch Normalization** was developed to handle internal covariate shift. Internal covariate shift describes the change in the distribution of network activations due to the change in network parameters during training. Batch normalization layers handle this problem by shifting it to zero mean and unit variance for every batch, resulting in normalized input.

Optimizers The role of an optimizer is to update the weights and biases such that the loss function is minimized. Most optimizers calculates some form of gradient for every few training cycles, updating the weights and biases, so that a lower cost is achieved. Often they get stuck in local minimums. Finding the global minimum is next to impossible, except for trivial problems. The optimizer used for this thesis is *adam* [27]. This is an optimizer for first-order gradient-based optimization of stochastic objective functions, based on adaptive estimates of lower-order moments. The authors claim that *adam* is computationally efficient, has little memory requirements, is invariant to diagonal rescaling of gradients and is well suited for large problems with respect to data and/or parameters. A simple, unoptimized version of *adam* is displayed in algorithm 1.

Algorithm 1 *adam*

Require: α: Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector
$m_0 \leftarrow 0$ (Initialize 1^{st} moment vector)
$v_0 \leftarrow 0$ (Initialize 2^{nd} moment vector)
$t_0 \leftarrow 0$ (Initialize timestep)
while θ_t not converged **do**
    $t \leftarrow t + 1$
    $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Get gradients w.r.t. stochastic objective at timestep t)
    $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Update biased first moment estimate)
    $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Updates biased second raw moment estimate)
    $\hat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Compute bias-corrected first order moment estimate)
    $\hat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Compute bias-corrected second raw moment estimate)
    $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \hat{m}_t / (\sqrt{\hat{v}_t} + \epsilon)$ (Update parameters)
return θ_t (Resulting parameters)

2.4 Metrics and PCA

There are many metrics that can be used when evaluating the quality of an ML algorithm. The most common and ubiquitous metrics are presented here. The definitions are obtained from [20], [16] and [14].

If a prediction is equivalent, to some satisfying degree, to a ground truth stating *true*, then this is called a True Positive (TP). If the prediction contradicts the ground truth and predicts *false* when the ground truth states *true* then its a False Negative (FN). If the prediction states *true* while the ground truth states *false* its called a False Positive (FP). If both prediction and ground truth agrees on *false* its called a True Negative (TN). All these possibilities are displayed in Figure 2.10.

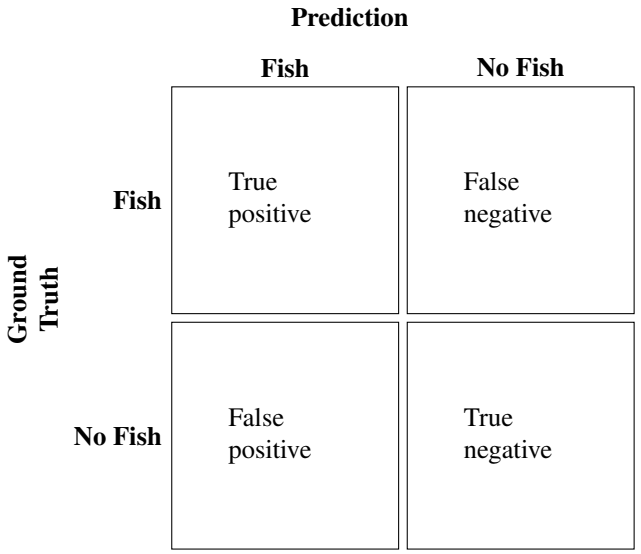

Figure 2.10: A confusion matrix relating TP, TN, FP, and FN.

Accuracy, given by Equation 2.17, is the ratio of correct predictions to all the predictions.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.17}$$

Precision, given by Equation 2.18, is a measurement of how precise the predictions are. It yields the percentage of predictions that agrees with the ground truth.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2.18}$$

Recall, given by Equation 2.19, describes how well an algorithm remembers all the TPs in an image.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2.19}$$

Precision and recall are strongly related. High precision and low recall means that is likely that the detected objects are detected correctly. Low precision and high recall means that all the objects are detected, but also that a lot of junk has been labeled incorrectly. For most applications it is desirable to find the parameters that lead to the best combined precision and recall. The F1-score (Equation 2.20) achieves this by simply combining the two:

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.20}$$

2.4.1 Principal Component Analysis

PCA is an unsupervised technique for identifying patterns in data, and expressing the data in such a way as to to highlight their similarities and differences [48]. Most commonly it is used as a dimensionality reduction method [25]. The fundamental idea is to represent a dataset using fewer variables than the original dataset, while retaining as much information as possible. With this approach, eigenvectors of the covariance matrix, explaining the majority of the variance of the dataset, are called principal components. In practice, these eigenvectors are ordered by the amplitude of corresponding eigenvalues containing main characteristics of the dataset. In order to obtain the principal components, an orthogonal linear transformation of the dataset must be defined. By identifying the direction of maximum variation, in the feature space, the problem definition can be reduced to finding the eigenvectors of the covariance matrix C associated with the dataset.

$$C = W \Lambda W^{-1} \tag{2.21}$$

The eigenvectors are orthogonal and span the N-dimensional subspace that explains a significant amount of the variance in the dataset. Let $\boldsymbol{x}$ be the original feature vector in the dataset and $\boldsymbol{w}_n$ an eigenvector associated with the n-th largest eigenvalue. The principal component is then given by:

$$PC_n = \boldsymbol{w}_n^T \boldsymbol{x} \tag{2.22}$$

An advantage with PCA is that there are no limitations on how many components the dataset can be reduced to. A disadvantage is that the algorithm does not preserve the class labels when finding the projected subspace.

CHAPTER 3

METHODS

The topics presented in this chapter:

- The acquisition and extraction of optical and acoustic data, demonstrating the setup for data procurement as well as highlighting its challenges.
- Derivation of an opti-acoustic relationship, presenting necessary assumptions and defining a fusion scheme for optical and acoustic data.
- Verifying the opti-acoustic relationship empirically, displaying soundness of methods.
- Training a NN on frequency data and utilizing cross-correlation with images, indicating that frequency data can be analysed with the aid of labeled optical data.

3.1 Data Acquisition and Extraction

The measurement station at Fulehuk in Norway can be seen in Figure 3.1. The station has a camera, a sonar and an artificial lighting source. It is deployed on the ocean floor 14 meters below the water surface, oriented upwards, looking up at the water surface. The camera is a Goblin Shark and records in 1080p at 30 fps with a horizontal angle of view of $92°$ [24]. The sonar system consists of the transducer Simrad ES200-7CDK Split [33] together with the transceiver WBT mini [2].

The transducer is a compact, composite, split-beam transducer. It has three sectors of composite materials able to transmit and record acoustic waves. The beamwidth is $7°$ degrees at nominal operational frequency. The nominal frequency is 200 kHz and its total frequency range is from 185 to 255 kHz. The transmitter, Simrad WBT Mini, is a wideband transceiver capable of transmitting and receiving pulses over a wide range of frequencies. Combining this wideband transceiver with the Simrad ES200-7CDK Split wideband transducer it is possible to make sweep transmissions (chirp) where the frequency continuously increases throughout the transmitted pulse. The software used to control and interface the sonar system is Kongsbergs Simrad EK80.

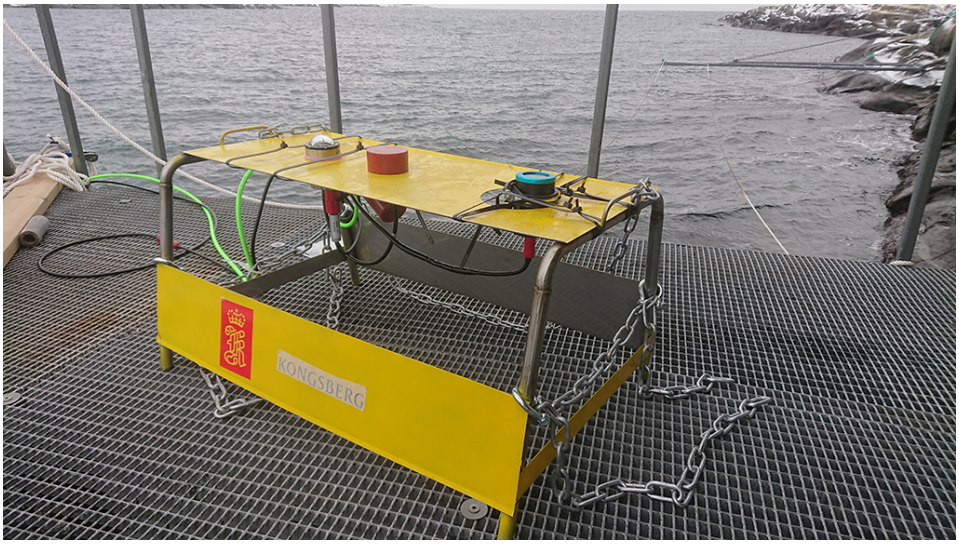

Figure 3.1: The measurement station. The leftmost glass dome is the camera. The red cylinder is the sonar and the blue and black cylinder on the right is the artificial lighting source.

Images were recorded between March and August, while sonar data was recorded between February and May 2019. The hardware was initially configured such that the camera and sonar would continuously capture data, while artificial lighting would be enabled during nighttime. In order to lessen the data burden, images were uploaded to the storage container at 6 or more seconds intervals during March, and much more infrequently (minutes to hours) during June, July and August.

The sonar data is stored in a proprietary .RAW-file format, where each file is 100 MB in size and contain roughly 6 minutes of data. While analysing the data with EK80 it was found that several sections of data were missing. An example is shown in Figure 3.2, where an echogram produced by EK80 is observed. There are approximately 7 minutes in between the left and the right hand side of this figure.

Due to faulty equipment, there exist several, different timestamps for every image. An image might have both the timestamp 2019-03-03 10:00:00 and 2019-03-03 10:00:27, and this difference is not consistent. In fact, it drifts throughout the year. Neither the optical nor the acoustic data explicitly contain information about the time zones wherein they were captured, making the temporal information within both types of data potentially erroneous. The inconsistencies with respect to temporal information is a major challenge and will be handled in detail in section 3.3.

During the acquisition of the data, the recorded data was uploaded to an online blob storage container, which is only available for the Windows operating system. With credentials, the desired period of images and transducer data can be downloaded to a local machine manually. Proceeding, is a short explanation of how labeled optical data and semi-raw

acoustic data is extracted:

From the work done in this thesis' preceding pre-project, and the resulting paper [51], approximately 100.000 labeled images were readily available. On a remote private server, the fully trained YOLO v3 algorithm was hosted, ready to create bounding boxes for fish images. A module was then written, piping the data from the Azure storage container to the private server over ssh. There it ran the YOLO algorithm on the images, extracting their corresponding labels, and finally sending the labels and the images to a local environment for further handling [50].

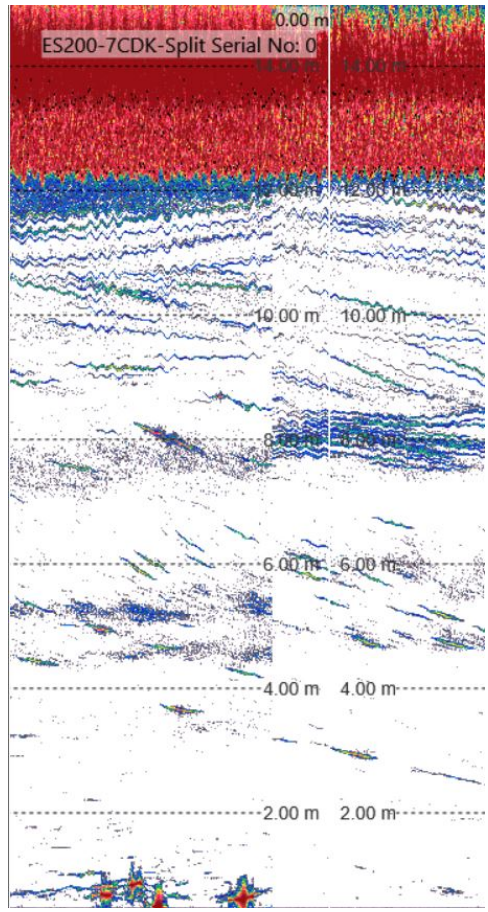

Figure 3.2: An example of missing sonar data.

In order to extract data from the .RAW-files they must be parsed by EK80 and simultaneously piped to a desired workspace. It would be highly desirable to know the format of the .RAW-files such that the information in them could be retrieved without the aid of EK80, but unfortunately this could not be made available.

The foundation for the piping tool is the EK80 extractor [52] written by Terje Nilsen at Kongsberg Maritime [1]. This tool, which is a python module, contains the bulk of the communication protocols, allowing for extraction of data from the EK80 software. Since some necessary functionality was found to be missing, we wrote an embellished module available at fishynet [50]. This embellished module also deals with some bugs in the piping tool. In order to get the desired data EK80 must load and parse the desired .RAW-file. While it parses the data, datagrams are piped to the local server over UDP. In total EK80 allows for subscribing to 10 different datagrams. In the embellished module [50] there are snippets that handles these different datagrams. In general, when working with EK80 there is not a lot of humanly tangible resources to work with.

Since EK80 has to run in a windows environment while the piping module needs to run in a Linux environment Windows Subsystem for Linux (WSL) was setup with an Ubuntu 18.04 distribution.

3.2 Opti-acoustic Methodology

Vision based sensors have been extensively used in autonomous underwater vehicles applications. The value in optical sensors comes from their high detail which can also include colour information [15]. There are however several drawbacks like needing texture, light attenuation, water turbidity and algae presence to name a few. Artificial lighting can alleviate some of these problems, but without homogeneous lighting it may itself be a problem. Sonars are in general more robust, and can pierce much farther into light prohibiting mediums, such as water. Typically, cameras record less than 20 meters underwater [15]. The main drawback of sonars is that, even with recent breakthroughs, they simply do not provide the level of detail a camera can. By fusing optical and acoustical data from the camera and sonar, respectively, it might be possible to harness to strenghts of each sensor system. In this section a method for such fusion will be presented.

In Figure 3.3 the basic setup of the camera and sonar is presented. It is of interest to be able to identify an object in one sensor system, and then have a correspondence, such that the same object can be found in the other sensor system. To make this feasible the FOV overlap of the optical and acoustic sensor systems must be calculated. Both systems are stationary. In order to proceed two assumptions are made:

1. The sonar has a perfectly, cone-shaped FOV.

2. All objects of interest are located at the ocean surface.

As stated earlier the emitted acoustic pulses spread spherically in a homogeneous medium. With assumption 1 it is assumed that all echos outside of a perfect cone are filtered out. With the setup of measurement systems like in Figure 3.3, where the sonar and camera are located at the ocean floor looking up at the surface, assumption 2 indicates that all objects of interest are located at the point where the cross section FOV is at its largest. The assumption that all objects of interest are at the surface is obviously not true, but is necessary and has the benefit of simplifying calculations. The necessity stems from the fact that the camera captures no information about the depth of objects in the given environment. In Figure 3.4 the measurement systems are seen as from the ocean surface.

Since a split-beam transducer is utilized the sonar system is able to identify targets in world coordinates. Meaning that the complete location of objects can be determined. The optical sensor operates in 2D giving no information depth of targets. In essence, a map from 3D world coordinates to 2D image coordinates is required.

The subscripts a and o represents measurements in the acoustic and optical coordinate systems, respectively. The superscript i is used when units are in the image plane. If no superscript is present, the units are in world-coordinates or coordinate free, based on context. The scheme that is derived here assumes that the horizontal FOV of the camera is supplied. The procedure that ensues, follows the basic four steps:

1. Calculating the size of the camera rectangle at the ocean surface.

2. Calculating the radius of the sonar circle at the ocean surface.

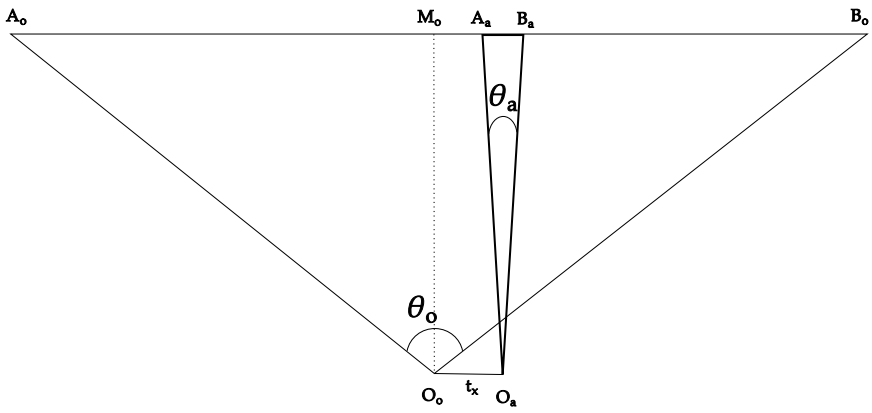

Figure 3.3: Cross section of the FOVs of the camera and sonar.

3. Calculating the scaling factor between the camera rectangle in world coordinates and image coordinates.

4. Applying the scaling factor to the sonar circle and its offset.

The origos of the camera and sonar are denoted as O_o and O_a, respectively. From Figure 3.3 it is observed that the measurement systems project out a cone and a rectangle that in 2D are equivalent to triangles. Since the camera projects a rectangle, it is seen that Figure 3.3 and Figure 3.4 are related by $x_o = 2\overline{M_oR_o}$. By utilzing the law of sines

$$x_o = 2\overline{M_oR_o} = 2\frac{\overline{M_oO_o}\sin\left(\angle M_oO_oB_o\right)}{\sin\left(\angle M_oB_oO_o\right)} \tag{3.1}$$

Since the aspect ratio of images by nature is invariant to the coordinate system it is expressed in

$$\frac{I_x^i}{I_y^i} = \frac{x_o}{y_o} \tag{3.2}$$

which yields $y_o = \frac{I_y^i x_o}{I_x^i}$. Thus, both width and height of an image in world coordinates is obtained. By using the same arguments as for x_o it is observed that the radius of the sonar cone can be expressed as

$$R_a = \frac{\overline{M_aO_a}\sin\left(\angle M_aO_aB_a\right)}{\sin\left(\angle M_aB_aO_a\right)} \tag{3.3}$$

Since the image width is known in both world and image coordinates a scaling factor is procured

$$S = \frac{I_x^i}{x_o} \tag{3.4}$$

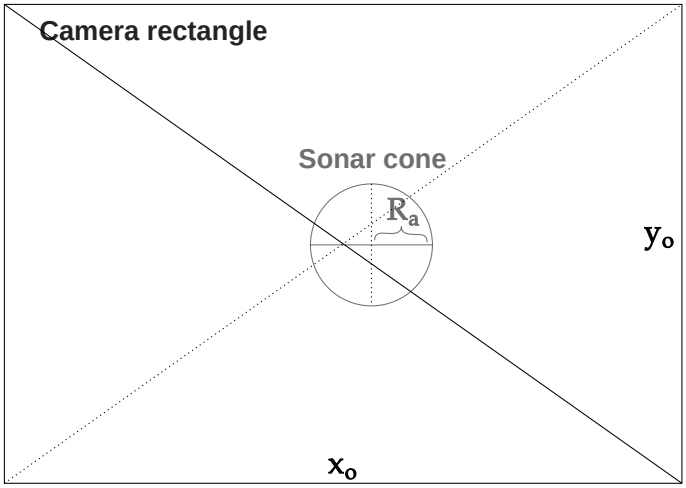

Figure 3.4: The beams from the camera and sonar as seen from the ocean surface.

With the scaling factor S, the sonar radius can be transformed into image coordinates

$$R_s^i = R_s S \tag{3.5}$$

Since both sensors have parallel FOVs it is trivial to calculate the offset of the sonar circle inside the camera image.

$$t_x^i = t_x S \tag{3.6}$$

The value t_x^i is the number of pixels the sonar circle should be offset the centre of the camera image. Since the illustrations are consistent with the physical measurement station this corresponds to directly moving the sonar circle t_x^i pixels to the right in an image. Due to the geometric symmetry of the system, results reached in 2D world coordinates, can be extrapolated to 3D world coordinates.

3.3 Verification of the Opti-acoustic Relationship

The camera and sonar are situated $t_x = 21$ cm apart from each other. The camera has a horizontal FOV of $92°$ and the sonar $7°$. From deployment of the measurement station is is known that the equipment is located 14 meters below the water surface. Utilizing the equations developed in section 3.2 the radius and offset for the acoustic region in the optical data is procured:

$$R_s^i = 56.701 \tag{3.7}$$

$$t_x^i = 7.702 \tag{3.8}$$

where the superscript indicates that the values are in pixels. An example of the acoustic region is displayed in Figure 4.2. A fish is deemed to be within the acoustic region if the center of its bounding box lies within the region.

Before proceeding with building an object detection architecture utilizing cross-correlation of opti-acoustic data, the spatial and temporal relationship within the opti-acoustic data must be verified empirically in accordance with the description in section 3.2. In other words, to show that a satisfactory amount of fish are located at corresponding locations and timestamps, in both the optical and the acoustic sensor data. However, due to uncertainty in the timestamps of the data, it is not sufficient to just verify the geometric relationship defined in section 3.2 - the temporal relationship must also be determined. Furthermore, since location and time are coupled, they must be verified simultaneously. To combat this problem in a tangible manner, a few key structures must be defined. The optical data is structured as in Equation 3.9.

$$\mu_o = \{(t_1, I_1), (t_2, I_2), ..., (t_n, I_n)\} \tag{3.9}$$

where I_n is an RGB image matrix of dimensions $(1080 \times 1920 \times 3)$. The intervals between $t_1, t_2, \ldots, t_n$ is not consistent in the dataset. From Equation 3.9 detections of fish within the acoustic region is extracted, in order to get a more convenient dataset. This means that for every image and timestamp pair in Equation 3.9, the detected fish within the acoustic region of the images are found, according to the explanation in section 3.2. Subsequently, the data is structured as:

$$\mu_o' = \{(t_1, n_1), (t_2, n_2), ..., (t_n, n_n)\} \tag{3.10}$$

where $n_1, n_2, \ldots, n_n$ is the number of fish located at the corresponding timestamp. Equation 3.10 is illustrated in Figure 3.5. In this figure the number of fish located within the acoustic region between 7 AM and 5 PM on the third of March 2019 is seen.

The acoustic data is structured as in Equation 3.11.

$$\begin{aligned} \mu_a = \big\{&(t_1, d_1, \theta_1, \phi_1, Sa_1, \boldsymbol{\gamma_1}), \\ &(t_2, d_2, \theta_2, \phi_2, Sa_2, \boldsymbol{\gamma_2}), \\ &\vdots \\ &(t_n, d_n, \theta_n, \phi_n, Sa_n, \boldsymbol{\gamma_n})\big\} \end{aligned} \tag{3.11}$$

where d is the depth, θ the alongship angle, ϕ the athwartship angle, S_a is as explained in chapter 2, and $\boldsymbol{\gamma}$ is the frequency response, which is an array of 1000 numbers that contain the amplitude of the echo for the frequencies from 185 to 255 kHz. The amplitude is denoted in decibels. It is important to note that the elements in μ_a are all *targets* according

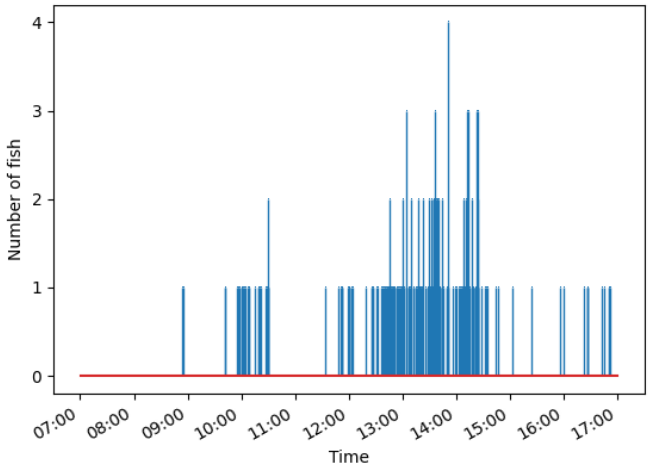

Figure 3.5: All fish found in camera within the sonar region the third of March 2019.

to EK80. This means that they all have an echo response over a certain *Threshold, MaxGainComp,* and *PhaseDeviation*. A priori, the true Threshold for the species of fish in the images, is unknown. Thus, this threshold value is regarded as an unknown parameter that must be estimated. The three values are listed as parameters of μ_a.

$$(\mu_a : \text{Threshold}, \text{MaxGainComp}, \text{PhaseDeviation}) \tag{3.12}$$

MaxGainComp and PhaseDeviation are set fixed, at 3dB and 25° degrees, respectively. This is explained in section 4.1. Thus, only Threshold is regarded as a value to be estimated.

In Figure 3.6 the same time window as for the optical fish in Figure 3.5, is plotted for different threshold values. Based purely on these figures it is difficult to tell which threshold produce the best match with the optical detections. In general there are a lot more detections made by the acoustic system than the optical. This makes sense since the camera takes an image roughly every 6 seconds, while the sonar system makes detections nearly continuously. Since the sonar system makes detections continuously and only records the echos over a certain threshold the the timestamps $t_1, t_2, ..., t_n$ do not have a fixed distance between them. The timezone of the acoustic data is also unknown and must be matched with the timezone of the optical data.

In order to find the optimal threshold and temporal relationship within the opti-acoustic data a somewhat crude approach is taken. In essence, a sliding window approach is used. The idea is to find, for a given threshold, the temporal shift that produces the highest value of common detections, called correspondences, between the optical and acoustic measurements. To simplify the search-space, the optical data is frozen in time, while the

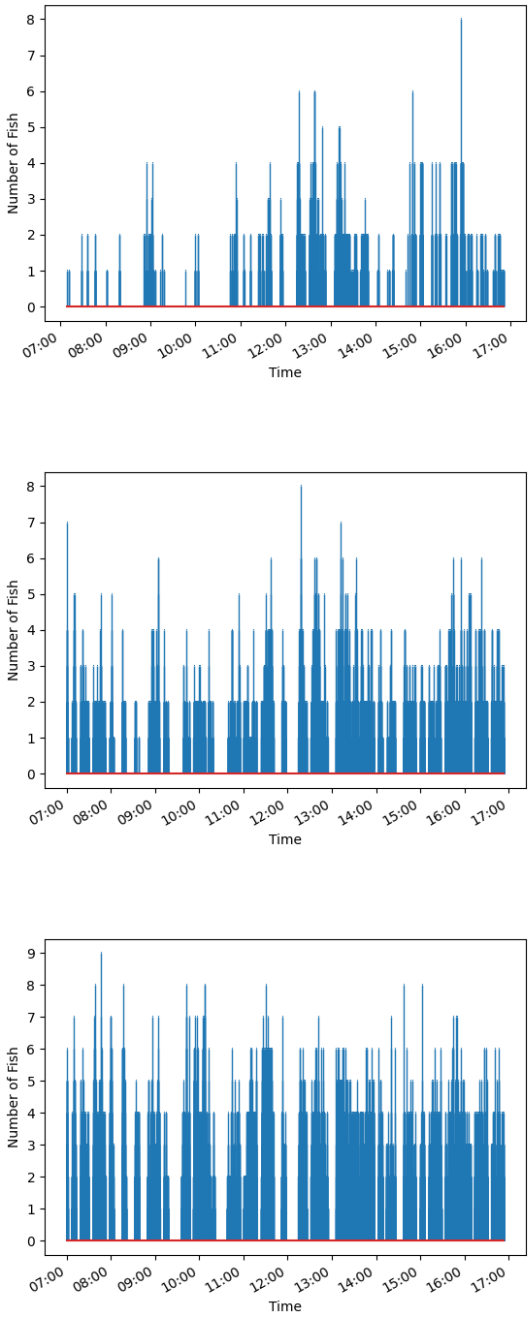

Figure 3.6: Fish located by the sonar for different thresholds. Top: -60dB, middle: -65dB, and bottom: -70dB.

acoustic data source is shifted. Define the subsets O and A of μ'_o and μ_a, respectively, such that

$$O(t_a, t_b) = \boldsymbol{t} \qquad \forall\, t \in \mu'_o \text{ where } t_a < t < t_b \tag{3.13}$$

$$A(t_a, t_b) = \boldsymbol{t} \qquad \forall\, t \in \mu_a \text{ where } t_a < t < t_b \tag{3.14}$$

where t_a and t_b are any arbitrary timestamps. Let C be the number of common timestamps of O and A, defined as the cardinality of the subset O of A.

$$C = |O \subseteq A| \tag{3.15}$$

where an element $o \in O$ is a member of A if $o = a$ for any element $a \in A$. With these constructions it is possible to define a constrained optimization problem that identifies the ideal temporal shift. The optimization problem is shown in Equation 3.16.

$$\max_{t_a, t_b} C = \max_{t_a, t_b} |O(t_a, t_b) \subseteq A(t'_a, t'_b)| \text{ s.t. } t_a - t_b = t'_a - t'_b \tag{3.16}$$

The results are presented and discussed in subsection 4.2.2.

3.4 FCN on acoustic data utilizing cross-correlation with optical data

Now it is desirable to create an algorithm that is capable of predicting whether a sonar measurement corresponds to the presence of a fish or not. The input to the algorithm will be the data shown in Equation 3.11, but without time. It is not desirable to train the algorithm to predict the presence of a fish based on the current time. Furthermore, it would be ideal if the algorithm itself could filter out any meaningless information, making it easier to use. To this end, it is fed with data having a Threshold value of -100dB, MaxGainComp of 3dB and PhaseDeviation of 25° degrees. The structure of the input is shown in Equation 3.17:.

$$\begin{aligned} x = \big\{ & (d_1, \theta_1, \phi_1, Sa_1, \boldsymbol{\gamma_1}), \\ & (d_2, \theta_2, \phi_2, Sa_2, \boldsymbol{\gamma_2}), \\ & \vdots \\ & (d_n, \theta_n, \phi_n, Sa_n, \boldsymbol{\gamma_n}) \big\} \end{aligned} \tag{3.17}$$

which is a matrix of dimensions 27675×1004. In other words there are 27675 datapoints which each has 1004 features. The 27675 datapoints correspond to three days of data, sampled between 08:00AM and 17:00PM for each day. The corresponding labels y are extracted from Equation 3.10. These are then turned binary by only evaluating if there are one or more fish present or not.

3.4.1 Pre-processing and Dimensionality Reduction

Too reduce the numbers of estimation errors and calculation times, the data is normalized prior to entering the NN [49]. For every column in the input matrix shown in Equation 3.17 the mean value and standard deviation is calculated and used to normalize the input

$$Z_j = \frac{x_j - \bar{x}_j}{\sigma_j} \tag{3.18}$$

where j denotes column.

Dimensionality reduction is most commonly used to decorrelate features and acquire insight into how well each feature performs. The pheomena known as the curse of dimensionality implies that using a few good features is beneficial for classification. The phenomena refers to the fact that classifiers often degrade in performace when presented with too many features compared to samples [56]. It is assumed that significant noise is present in frequency portion of the dataset. PCA is applied to reduce the dimensionality of the frequency portion, such that the effect of noise is diminished. This further has the benefit of reducing training times as well. In the next chapter, results are shown from using the entire dataset, a dataset with only 2, and 10 frequency components.

Too further reduce training times only a subset of the 27675 datapoints are used. Within the data there are significantly more TNs than TPs. Among the TNs, only 1 in 15 is kept. This reduced the dataset down to about 3000 datapoints.

3.4.2 Architecture and Training

The design of the network was established mostly through rules of thumb and trial-and-error. The basic procedure for finding a desirable network consisted of starting with a few layers on the same size as the input. Thus with 1004 features, the layers would be of roughly size 1000. Then both expanding the size and number of layers were tried and diminishing the size and number of layers, retraining between every change. A network on the form of Table 3.1 was found to perform more or less satisfactory.

Table 3.1: Network architecture

Size	Type	Activation
1024	dense	sigmoid
512	dense	relu
256	dense	relu
128	dense	relu
64	dense	relu
32	dense	relu
16	dense	relu
8	dense	relu
4	dense	relu
1-2	dense	sigmoid/softmax

Not every aspect of designing a network can be encompassed within this text without it expanding forever, and to only focus on a few relevant key aspects, the three following experiments was run:

1. Two or one output neurons using sigmoid and softmax, respectively, in the output layer.
2. Withholding S_a.
3. Using a various number of PCA components.

Regarding 1. it is of interest to check whether a network with two outputs is preferable over a network with one output. For a network with two outputs the y values are one-shot encoded to work with the fully connected NN structure defined in Table 3.1. By withholding S_a it is checked if the network is capable of finding any pattern without utilizing heavily pre-processed data such as S_a. This variable is a pseudo-measure of the amount of biomass present within the FOV of the sonar, and thus slightly defeats the purpose of the task set forth to accomplish by the NN. It is desirable for the network to be able to classify objects based on as pure frequency information as possible, as opposed to heavily pre-processed data by EK80.

The loss specified for the network is the binary cross-entropy loss described in Equation 2.16. The optimizer used is *adam*. The following parameters for adam perform satisfactory for the problem at hand, enabling the networks to converge to reasonable minimums with appropriate training times:

$$\begin{aligned} \alpha &= 0.001 \\ \beta_1 &= 0.9 \\ \beta_2 &= 0.999 \\ \epsilon &= 1e-07 \end{aligned} \tag{3.19}$$

ReLU is chosen as the intermediary activation function due to its simplicity, calculation efficiency and obvious popularity.

The network is created using Keras version 2.3.1 in python utilizing the GPU using Cuda version 10.0 Several other auxiliary software packages are utilized. A few key packages are:

- Matplotlib v. 3.2.1 $\rightarrow$ Plotting, visualizing
- Opencv-python v. 4.2.0.34 $\rightarrow$ Image manipulation
- Pandas v. 1.0.3 $\rightarrow$ Mathematics, linear algebra
- Scikit-learn v. 0.23.1 $\rightarrow$ Statistical metrics
- Tensorflow v. 2.2.0 $\rightarrow$ Machine learning

All the networks are trained 10 times each to obtain the best weights. The split between training and validation is 20 %. All the data is randomly shuffled at all stages. The batch-sizes is always set to the same size as the dataset that is being trained on. Due to the computer specs and size of the data this is no issue.

CHAPTER 4

RESULTS AND DISCUSSION

In this chapter the results procured in the previous section are presented and discussed. The topics presented in this chapter:

1. Results and discussion regarding the data acquisition and extraction.
2. Results and discussion regarding the derivation and verification of the opti-acoustic relationship.
3. Results and discussion regarding training a NN to perform classification on opti-acoustic data.

4.1 Data Acquisition and Extraction

An error that could not be properly circumvented was EK80 transmitting faulty data. Experimentation showed that the faster EK80 parsed data the more erroneous messages were transmitted. To circumvent this problem, EK80 was set to parse and transmit data at the slowest possible setting, which was real-time. This means that getting three days worth of data took three days. Even when doing this, faulty data was still transmitted. The inhibiting factor was the speed at which messages could be stored on the receiving end. Measures were implemented to combat this, for example by reducing the amount of redundant information the piping module was displaying, but to limited avail. The still faulty data was simply discarded on the receiving end, producing a reduced dataset.

In order to achieve the goals set forth in this thesis, it would be desirable to extract as much raw data as possible from the sonar system. For example, it would be desirable to obtain the data recorded by the three distinct listening sections of the transducer. However, parsing the .RAW-files produced by the sonar directly, was found to be impossible due to the restricting proprietary format they were stored in. In order to extract as raw data as possible, without actually parsing the .RAW-files manually, it would have been desirable to set Threshold, MaxGainComp and PhaseDeviation to their maximum values. This would yield significantly more measurements from the sonar system, as targets would not be filtered by Threshold, MaxGainComp and PhaseDeviation. Since the amount of missing

messages became ludicrous as these values were set higher and higher, it was decided that MaxGainComp and PhaseDeviation were kept at their default values, and only the Threshold increased. This was a great limiting factor to the richness and quality of the dataset.

Labeled data from June, July and August are of relatively poor quality compared to data from March. This is partly due to algae growth on the camera lens. A team of divers occasionally cleaned the camera lens, but not frequently enough. This, and other reasons, are explained in depth in [51]. Because of this, and the fact that extracting data from EK80 is time consuming, only data from March was utilized. Furthermore, only daytime data was used, because of a physical discrepancy with the artificial lighting, that rendered it useless during March. Furthermore, the labels on the data from March is not perfect. The labels generated by YOLO only exhibit a mAP of 0.88. This will naturally affect the results of the proceeding experiments.

Additionally, a portion of the data from March is quite noisy. An image from 14:14:11 on the third of March is seen in Figure 4.1. Almost all images between 13:00 and 17:00 of any day during March look like this image. In this figure a significant amount of tiny fish are observed. With such an extreme amount of tiny fish, it is difficult to verify the validity of the shifting procedure. This is because fish will enter and exit the acoustic region extremely frequently. Thus, making exact temporal information critical, which due to the uncertainty in time stamps of both optical and acoustic data, difficult to achieve.

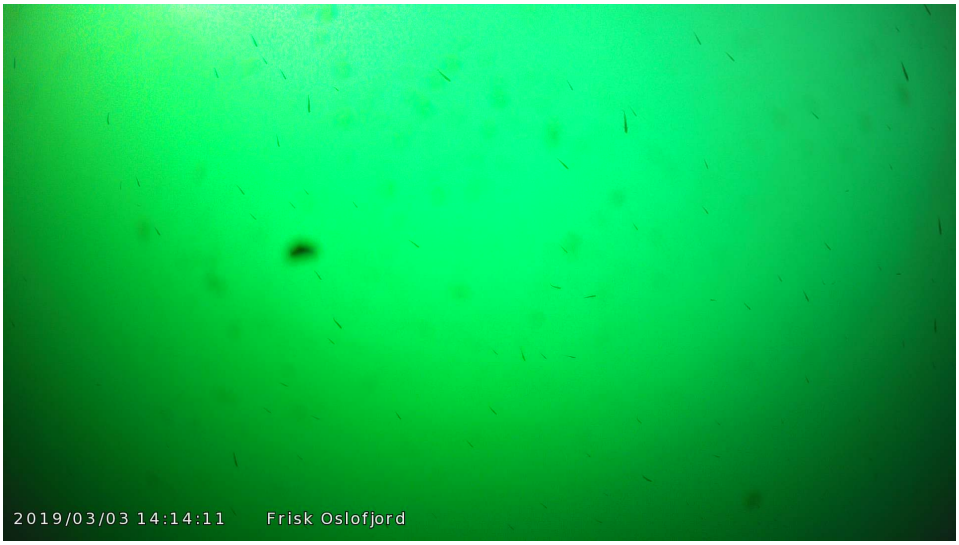

Figure 4.1: An example of abundance of fish after noon.

4.2 Derivation and Verification of the Opti-acoustic Relationship

4.2.1 The Acoustic Region

In Figure 4.2 it is observed that only a very small subset of the image overlaps with the acoustic region. Recalling the opti-acoustic theory from section 3.2 it is even noted that this is the optimistic region - the region at the ocean surface. In reality this section is most likely smaller. The sonar systems FOV is almost like a laser beam. Its opening angle is only 7°. Thus, it is evident, that with images only being captured roughly every 6-7 seconds, that the ratio of TPs to TNs, is skewed in favor of TNs. There is unfortunately no way to immediately alleviate this, as this is due to the inherent physical setup of the sonar system. In order to gain a larger FOV-overlap, a different sonar system must be set in place. For example a wide-scan sonar would be able to survey a larger volume and thus provide more valuable data. However, even though the overlap is small, it will still prove useful. There are still optical data, wherein fish are located within the acoustic region, such that progress can be made.

Figure 4.2: Illustration of the sonar region within an image. The blue circle is observed by both the optical and the acoustic sensor.

4.2.2 Verification of the Opti-acoustic Relationship

In order to lessen the computational burden, and for simplicity, not every timestamp is checked. t'_a and t'_b are set to fixed values, isolating a 20 minute interval of sonar data, while optimization is done with respect to t_a and t_b. In Figure 4.3 the result of this search is seen, when shifting the data 30 seconds backwards and forwards, and 1 hour backwards

and forwards. The number of common timestamps are shown on the y-axes, while the number of seconds shifted are shown on the x-axes. The uncertainty in the data stems from uncertainty in time zones, and the fact that the optical data has several timestamps. However, the difference is only in either hours or seconds. In Figure 4.3 it is seen that for a threshold of -60dB almost no overlapping detections are observed, no matter how the data is shifted. This indicates that a threshold of -60dB is to high. In the figures on the right hand side of Figure 4.3, where the threshold is -70dB, quite a few correspondences are found. It is observed that several maximum values of Equation 3.16 are found when shifting around 0 hours, and 20-30 seconds backwards and forwards with a -70dB threshold. However, this happens only for this specific 20 minute interval of acoustic data and is not a trend in general. These prolific matches are most likely caused by noise in this specific subset of the data.

Several matches are found for various shifts, but a 100% correspondence is not achieved. This is not ideal, and further strides should be made to improve on this. We hypothesise the following reasons for the results:

1. The geometric relationship between the sensors is wrong.
2. The search-space is not wide enough.
3. The data is faulty.

The first hypothesis might be caused by an error somewhere in the derivations or assumptions of the opti-acoustic relationship, as put forth in section 3.2. It might, for example, be caused by physical measurements of the measurement station being slightly wrong, or that the assumption that all fish are located at the ocean surface leads to the inclusion of too much noise. The second and third hypotheses are strongly connected. Perhaps the specific interval of shifting, displayed in Figure 4.3, contains too much missing or incorrect acoustic data. However, several different intervals, testing different portions of the dataset, were investigated, but yielded similar results.

Given the preceding results, it is concluded that progress should be made without shifting the data from their original time stamps. Furthermore, the threshold value will be increased. It is evident from Figure 4.3 that the fish present in the dataset produce echos with strength -65dB or lower. In order to be certain that as many as possible are recalled, the threshold will be set to -100dB for future experiments. This ensures that all fish should be recalled, however, at the cost of introducing significant noise into the dataset. It is however more desirable to feed the ensuing NNs with too much information rather than too little.

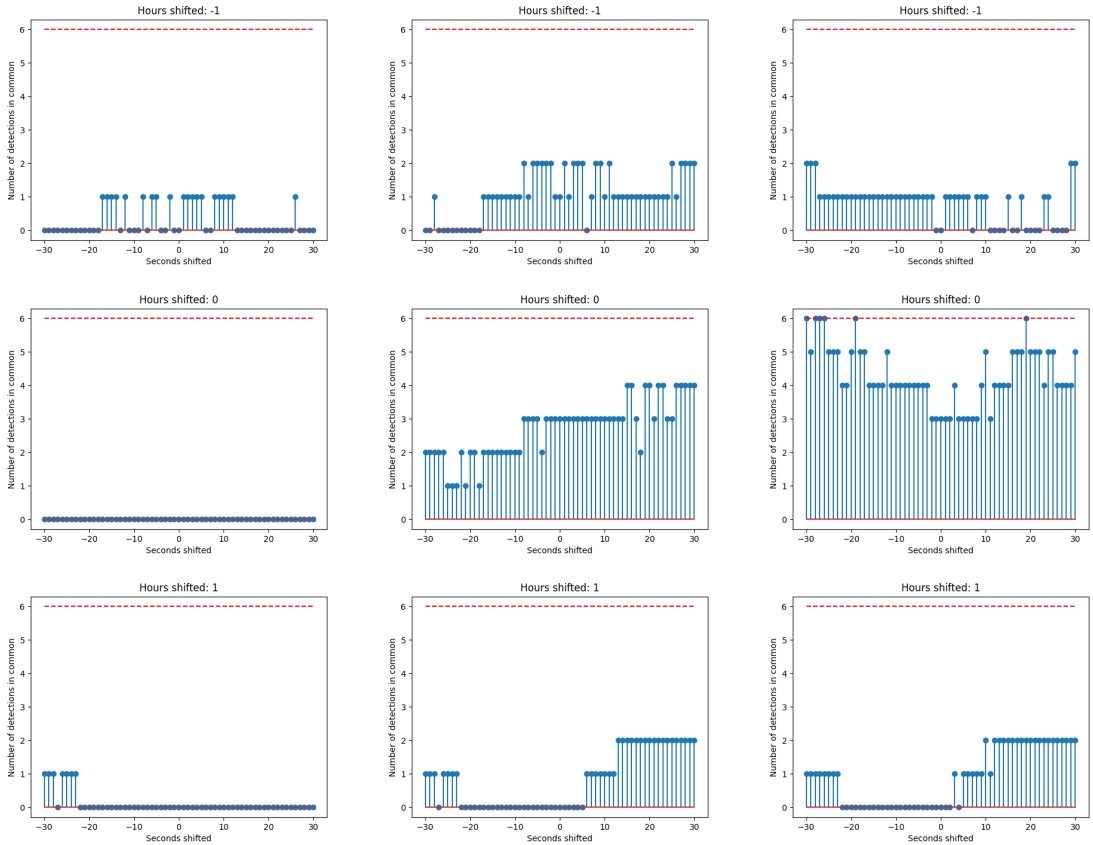

Figure 4.3: Examples of correspondences with shifted data. Columns from left to right: -60dB, -65dB, -70dB.

4.3 FCN on acoustic data utilizing cross-correlation with optical data

Number of PCA components	Final activation function	Output neurons	Inclusion of S_a	Validation			Evaluation accuracy
				Precision	Recall	F1-score	
0	Sigmoid	1	yes	0.526	0.588	0.555	0.593
2	Sigmoid	1	yes	0.563	0.584	0.573	**0.623***
10	Sigmoid	1	yes	0.519	0.561	0.539	0.585
0	Sigmoid	1	no	0.543	0.535	0.539	**0.604***
2	Sigmoid	1	no	0.498	0.539	0.517	0.565
10	Sigmoid	1	no	0.514	0.61	0.558	0.581
0	Softmax	2	yes	0.546	0.599	0.571	0.611
2	Softmax	2	yes	0.590	0.610	0.600	**0.648***
10	Softmax	2	yes	0.541	0.561	0.551	0.604

Table 4.1: Overview of training. * indicates the best results within its category. The validation metrics are *only* given the presence of fish and not the converse.

In Table 4.1 an overview of validation results are presented given different training parameters. From the setup of the experiment, it is recalled that there are two classes being validated. Class 0: no fish are present within the acoustic region, and class 1: fish are present within the acoustic region. The results under *validation* are only for class 1. The evaluation accuracy is calculated based on both classes. The results are divided into three blocks, divided by a thick black line, and the best results within each block is highlighted. A ”0” in the PCA column indicates that PCA was not used. The neural networks were trained 10 times each for every row.

It is observed that for all the experiments that either having two components or not using PCA at all yields the best results. However, the difference is slight, so it is difficult to ultimately conclude that one is better than the other. This difference is most likely caused by variations in the initial weights and convergence to slightly different local minimums. Simultaneously, observing that the results are not significantly distinct, indicates that PCA only yields the benefit of speeding up calculations. This speedup is however, quite significant and reduces the training times a tenfold.

When withholding S_a from training, keeping all 1000 frequency components, produce the best results. Again, the difference is small with respect to the validation metrics. It is however satisfactory that classification is at all possible without the inclusion of S_a. This indicates that it is possible to extract some tangible, useful information based solely on the distance, angle and frequency response of an object. This is a major result.

It is observed that having two output neurons with two PCA components yields the best result out of all the experiments. With the architecture that was utilized, it is evident that having two output neurons with softmax, produce better results. The difference is again slight, and it is thus difficult to make a final conclusion. A potentially, plausible hypothesis is that, since the classes are mutually exclusive, that a network with two output neurons gets penalized harder for incorrect predictions by the binary cross-entropy loss function.

It of importance to note that no other preprocessing than standardization and PCA has been applied to the data. It is evident that the network itself is capable of discarding the erroneous parts of the dataset. It is for example known, that echos produced by objects very close to the transducer, are almost pure noise, and consequently manually filtered in most cases [59]. However, it seems that the NN is able to contextualize and generalize this on its own.

In Figure 4.4 the confusion matrices produced by the top three rows of Table 4.1 is observed. When training a network on the full frequency data, with no PCA, the network classifies the absence of fish as TNs correctly, but struggles severely with the presence of fish. In fact, it mislabels more than 50 % of the data that corresponds to the presence of fish. When training on two PCA components, the network performs, quite a bit, better across the board. However, it is seen, from the validation on the presence of fish, that the predictions are split 50/50, which is far from ideal still. Using ten PCA components enables the network to finally be able to understand what constitutes the presence of a fish. This does however come at the cost of worse performance when predicting the absence of fish.

It is evident that the networks are capable of identifying some pattern within the data. Especially it is noted that the training results are quite impressive, while the validation results are not. The networks are not truly able to grasp the ideal pattern that enables discerning between the presence and absence of fish. It might however be the case that no such pattern actually exist in the data. The rate of correct identification of fish by humans is 89.3 % according to [57]. Ideally, the algorithm presented here would be able to perform similarly, but taking into account the quality of the labeled optical data and the noisy conditions of the environment, this is not guaranteed.

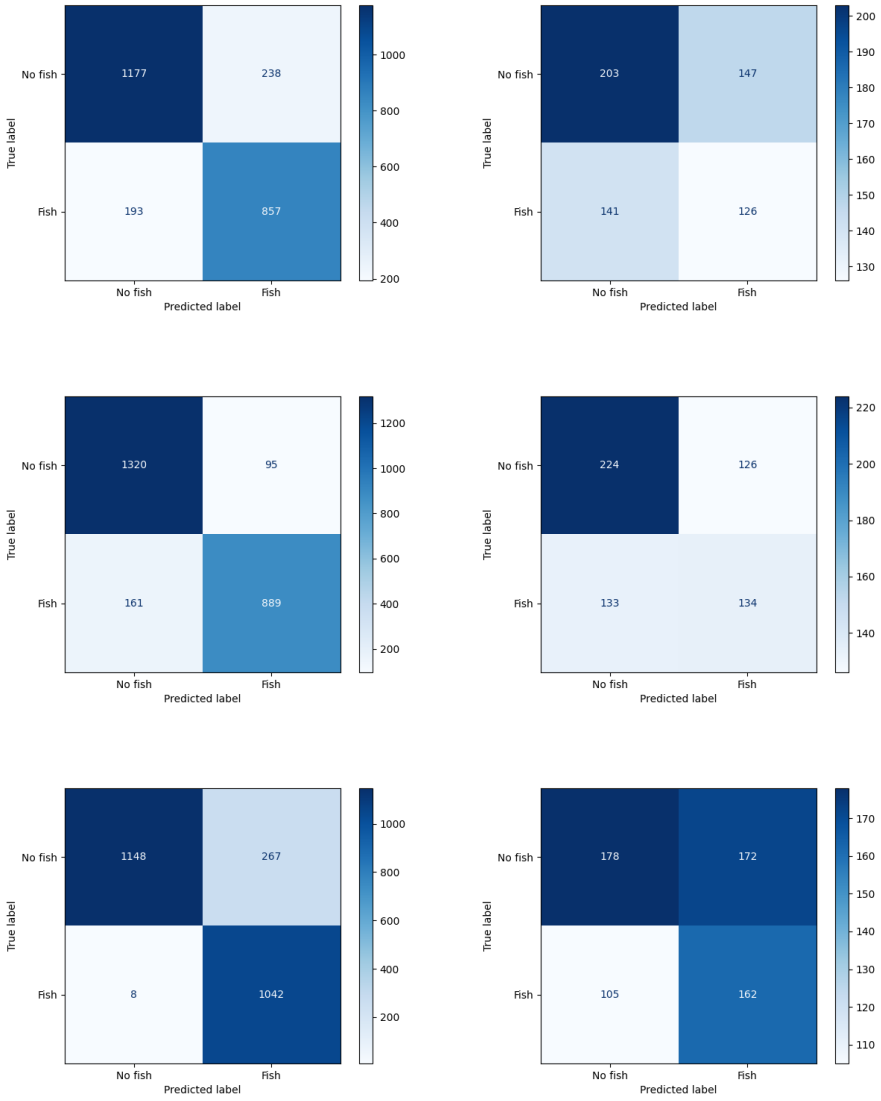

Figure 4.4: Left hand side contains training results while the right hand side contains validation results. Top row: All 1000 frequency components were used. Middle row: 2 PCA components. Bottom row: 10 PCA components used.

CHAPTER 5

CONCLUSION AND FUTURE WORK

The topics presented in this chapter:

- A summary of the entire thesis and evaluation of research questions and tasks.
- What improvements could have been made and potential future tasks.
- The implications and ramifications, that the methods embodied in this thesis, exhibit.

5.1 Conclusion

In this thesis, relevant background on optical and acoustic data, were presented, as well as introductory theory on NNs and the most relevant and ubiquitous statistical metrics. A method for merging optical and acoustic data was developed. Significant strides were made to make up for faulty data, both in terms of insufficient temporal information and spatial discrepancies. A scheme for aligning and combining the data sources was designed and empirically tested. The results were to a certain degree inconclusive with respect to ascertaining if the temporal information of the two data sources were aligned properly. Thus displaying, that there was an error in methods or an inherent discrepancy in the data sources. A NN was designed, as well as several variations of it, and trained on opti-acoustic data, in order to check if patterns could be extracted. The results showed promise, but not overwhelmingly satisfactory. The best NN produced an accuracy of 64.8 % on detecting fish in opti-acoustic data.

Ensuing, is an evaluation of the fulfillment of the research questions and objectives, as devised and presented in the introduction.

RQ1: *Does the acoustic data accommodate sufficient patterns for classification to be a possibility?*

It was shown in section 4.3, that some patterns, could be extracted from the acoustic data. The best network produced an accuracy of 64.8 %, which is promising, and quite a bit better than random guesses, but not necessarily ideal. Based on the methods presented, it is difficult to discern whether the accuracy is due to faulty methods or inherent in the data.

RQ2: *How can ML be used to extract patterns from acoustic data in real-time?*

It was shown that a supervised algorithm could interpret frequency data generated by a sonar, utilizing chirp capabilities. Together with labeled images, the algorithm was able to learn what information within an echo constitutes a fish.

RQ3: *How can optical and acoustic data be utilized in conjunction, to aid in automatic detection and classification?*

In section 3.2 a mathematical, geometric relationship, relating the spatial information of a camera and a sonar was devised. It was shown that this can be used to procure a dataset capable of assisting in training supervised agents without the need of labeled acoustic data. This is a great accomplishment, and helps alleviate the challenges connected to procuring labeled acoustic data. Specific data discrepancies, like insufficient temporal information, must be accounted for, like displayed in subsection 4.2.2.

Based on the answers to the preceding partial research questions, we can finally address the **primary research question**:

PRQ: *Is it feasible to employ deep learning to identify fish, utilizing unlabeled acoustic data in conjunction with labeled optical data?*

By empirically demonstrating that a fully connected NN was able to train on opti-acoustic data, it was shown that it indeed is feasible to utilize deep learning to identify fish.

During the theoretical and practical work necessary to answer the aforementioned questions, several specific tasks were completed. These tasks were listed as research tasks in the introduction and their fulfillment will now be reviewed:

RT1: *Creating tools for extracting acoustic data.*

Significant effort was poured into creating tools for extraction. Several challenges were confronted while piping data from the EK80 software. Most nefarious, were missing transmissions, and the transmission of faulty data. However, the tool capable of extracting sonar data is functional.

RT2: *Devising a geometric relationship between optical and acoustic data.*

A geometric relationship were devised, such as to relate the spatial information within the optical and acoustic data. Two restraining assumptions were imposed in order to make this feasible. Especially restraining was assuming that objects are located at the ocean surface. Furthermore, it was observed that the acoustic region was very small, producing few TPs. However, this was a fundamental restriction imposed by the available hardware.

RT3: *Generating a labeled dataset with opti-acoustic data.*

The dataset is composed of three days worth of data, harvested during daytime with natural lighting. Every camera image is linked with sonar measurements with corresponding timestamps. There are approximately 28.000 images with labels, coupled with corresponding sonar measurements.

RT4: *Demonstrating empirically, that optical and acoustic data can be combined to aid classification.*

Combining optical and acoustic data was troublesome, due to the specifics of the dataset, such as incomplete temporal and faulty spatial information. Measures were carried out to obtain a satisfactory spatio-temporal relationship. When investigating the shifting procedure devised in section 3.3, it became clear that the procedure was not able to ascertain a completely rigorous temporal correspondence. It was shown in subsection 4.2.2 that significant noise was present in the dataset, and thus that the temporal correspondence between the two data sources, was not necessarily completely sound. However, it was shown that a spatio-temporal relationship can in fact be devised bearing some merit, and which results would probably improve with improved data quality. The algorithm trained on opti-acoustic data made better than random predictions, proving empirically that opti-acoustic data can aid classification.

RT5: *Designing and implementing a NN capable of discerning between the presence and absence of fish, utilizing opti-acoustic data.*

Several NNs were designed, implemented, and trained on opti-acoustic data. The algorithms were capable of discerning between the presence and absence of fish.

5.2 Future work

If more time were allocated, there are several improvements that could have been made. There are also quite a few experiments that would be beneficial to perform:

1. Actually acquire raw data from the sonar system. It would be extremely valuable to repeat the experiments of this thesis with data harvested directly by the transducer. It is not certain that EK80 performs completely correct computations. EK80 relies on a significant amount of ad hoc mathematical methods, that are susceptible to errors. An example of this is S_a as described in subsection 2.1.1. However, actually obtaining pure, raw data was deemed impossible, due to the restraints imposed by EK80 and its proprietary data formats.

2. Develop a better opti-acoustic relationship with improved spatio-temporal information. With respect to time, it would be desirable to impose stricter conditions for time stamping both the optical and acoustic data. It is critical that data is obtained within the same time frame. Regarding the spatial relationship devised in section 3.2, it is desirable to relax the imposed assumptions. Especially, with respect to fish being located somewhere in the middle of the FOVs of the camera and sonar, as opposed to the ocean surface.

3. Both images and sonar data should be recorded much more frequently, and preferably labeled with millisecond timestamps. Video would be preferable over images with 6-7 seconds intervals.

4. A drawback of training a NN on the opti-acoustic data like presented here, is that the algorithm will only find fish that is visible in the optical data. Techniques should be implemented that still allow the algorithm to predict the presence of fish that is out of view from the optical sensor of the measurement station. One possible way to achieve this would be to setup an array of cameras, spanning a wide area, and using this to create an even richer opti-acoustic dataset.

5. Several improvements can be made with respect to the NNs presented in this thesis. During training of the networks, it was observed, that the given architectures were extremely susceptible to overfitting, and outputting the same prediction no matter what input they were fed. In some cases this happened in 9 out of 10 training sessions. In order to alleviate this, dropout should be implemented. Several different types of networks should also be rigorously tested.

5.3 Impact

It was shown in this thesis that it is potentially possible to utilize labeled images to train a network on acoustic data. This removes the need for hand-labeling acoustic data, which is a tremendously difficult task, requiring significant domain knowledge. This makes it easier to analyse acoustic data, which in turn will reduce costs and improve performance regarding surveying marine life. These surveys and their subsequent analysis will enable governments and businesses to understand how fishing affects marine environments and thus implement regulations that can prohibit destructive harvesting, slowing down the ever growing extinction rate. Even though the methods developed here are not necessarily directly applicable to such grandiose goals, and require further refining, they still provide a valuable contribution that others can build upon.

Training a network on fish detections is one thing, but the work presented here is by nature generalize to any type of object, be it sunken ships, lost equipment, rescue/retrieval missions, etc. Methods for treating multiple data sources with insufficient and faulty spatio-temporal information are ubiquitous in any job regarding data analysis in the real world. The techniques presented within this thesis are general and can be used for a multitude of data types, not necessarily just optical and acoustic data.

As with all research, these methods may be applicable to fields not intended, or thought of, by the author. There is always the possibility, that new technologies will be utilized in haphazard or malevolent ways. Or perhaps in making something unimaginably wonderful.

BIBLIOGRAPHY

[1] GitHub - The1only/ek80: Simrad EK80 echosounder python interface module.

[2] Simrad WBT Mini Compact wide band transceiver - Kongsberg Maritime.

[3] Sungbin . Fish identification in underwater video with deep convolutional neural network: SNUMedinfo at LifeCLEF fish task 2015, 2015.

[4] Emmanuel Abbe and Colin Sandon. Provable limitations of deep learning, 2018.

[5] Helge Balk. Echosounder basics analysing data. `https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=2ahUKEwjfgcah4PvnAhVP1qYKHfDpA30QFjAAegQIAxAB&url=http%3A%2F%2Ffolk.uio.no%2Fhbalk%2FBioAcoustic%2F1_Lecture%25200830_1000_Echosounder%2520basics_Analysing%2520data.doc&usg=AOvVaw0vwA8_pr9r9eUiOKGvmsIK`, 2011.

[6] Anthony Barnosky, Nicholas Matzke, Susumu Tomiya, Guinevere Wogan, Brian Swartz, Tiago Quental, Charles Marshall, Jenny McGuire, Emily Lindsey, Kaitlin Clare Maguire, Benjamin Mersey, and Elizabeth Ferrer. Has the earth's sixth mass extinction already arrived? nature. *Nature*, 471:51–7, 03 2011.

[7] Johan Barthélemy, Nicolas Verstaevel, Hugh Forehead, and Pascal Perez. Edge-computing video analytics for real-time traffic monitoring in a smart city. *Sensors*, 19(9):2048, 2019.

[8] Olav Brautaset, Anders Ueland Waldeland, Espen Johnsen, Ketil Malde, Line Eikvil, Arnt-Børre Salberg, and Nils Olav Handegard. Acoustic classification in multifrequency echosounder data using deep convolutional neural networks. *ICES Journal of Marine Science*, 01 2020. fsz235.

[9] Gerardo Ceballos, Paul Ehrlich, Anthony Barnosky, Andres Garcia, Robert Pringle, and Todd Palmer. Accelerated modern human-induced species losses: Entering the sixth mass extinction. *Science Advances*, 1:e1400253, 06 2015.

[10] Jiwoong Choi, Dayoung Chun, Hyun Kim, and Hyuk-Jae Lee. Gaussian yolov3: An accurate and fast object detector using localization uncertainty for autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 502–511, 2019.

[11] C. S. Clay and H. Medwin. *Acoustical Oceanography: Principles and Applications*. John Wiley & sons, 1977.

[12] David Demer, Lars Andersen, Christopher Bassett, Laurent Berger, Dezhang Chu, Jeff Condiotty, George Jr, Briony Hutton, Rolf Korneliussen, Naig Bouffant, Gavin Macaulay, William Michaels, David Murfin, Armin Pobitzer, Josiah Renfree, Thomas Sessions, Kevin Stierhoff, and Charles Thompson. Evaluation of a wideband echosounder for fisheries and marine ecosystem science., 01 2017.

[13] David Demer, Laurent Berger, Matteo Bernasconi, Eckhard Bethke, Kevin Boswell, Dezhang Chu, Réka Domokos, Adam Dunford, Sascha Fässler, Stéfane Gauthier, Lawrence Hufnagle, J. Jech, Naigle Bouffant, Anne Lebourges-Dhaussy, Xavier Lurton, Gavin Macaulay, Yannick Perrot, Tim Ryan, Sandra Parker-Stetter, and Neal Williamson. Calibration of acoustic instruments, 05 2015.

[14] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

[15] Fausto Ferreira, Diogo Machado, Gabriele Ferri, Samantha Dugelay, and John Potter. Underwater optical and acoustic imaging: A time for fusion? a brief overview of the state-of-the-art. 09 2016.

[16] Peter Flach. Performance evaluation in machine learning:the good, the bad, the ugly and the way forward. `https://aaai.org/ojs/index.php/AAAI/article/view/5055`, 2019.

[17] Om frisk oslofjord. `https://friskoslofjord.no/om-frisk-oslofjord/`.

[18] Mina S A Ghobrial. Fish detection automation from aris and didson sonar data. 2019.

[19] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[20] Aravind Kota Gopalakrishna, Tanir Ozcelebi, Antonio Liotta, and Johan J. Lukkien. Relevance as a metric for evaluating machine learning algorithms. In Petra Perner, editor, *Machine Learning and Data Mining in Pattern Recognition*, pages 195–208, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[21] JP Greff. Simrad fisheries underwater science products sm. `https://en.calameo.com/books/00532683324913a11927d`, 2015.

[22] Roy Edgar Hansen. Course materiel to inf-geo4310, university of oslo. `https://www.uio.no/studier/emner/matnat/ifi/INF-GEO4310/h13/undervisningsmateriell/sonar_introduction_2013.pdf`, 2013.

[23] Y. Hirama, S. Yokoyama, T. Yamashita, H. Kawamura, K. Suzuki, and M. Wada. Discriminating fish species by an echo sounder in a set-net using a cnn. In *2017 21st Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES)*, pages 112–115, Nov 2017.

[24] Inge Ivesdal. Imenco: Goblin Shark. `https://imenco.no/product/imenco-goblin-shark-ip-hd-wide-angle-overview-subsea-video-camera/`.

[25] Ian Jolliffe and Jorge Cadima. Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374:20150202, 04 2016.

[26] Ayoosh Kathuria. How to implement a YOLO object detector in PyTorch.

[27] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[28] Rolf Korneliussen, Laurent Berger, Fabio Campanella, Dezhang Chu, David Demer, Alex Robertis, Réka Domokos, Mathieu Doray, Sophie Fielding, Sascha Fässler, Stéphane Gauthier, Sven Gastauer, John Horne, Briony Hutton, • Federico, Iriarte Jech, Rudy Kloser, Gareth Lawson, Anne Lebourges-Dhaussy, and Charles Thompson. Acoustic target classification. ices cooperative research report no 344, 10 2018.

[29] Rayson Laroca, Evair Severo, Luiz A Zanlorensi, Luiz S Oliveira, Gabriel Resende Gonçalves, William Robson Schwartz, and David Menotti. A robust real-time automatic license plate recognition based on the YOLO detector. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–10. IEEE, 2018.

[30] P. Leadley, Cornelia Krug, Rob Alkemade, Rashid Sumaila, Matt Walpole, Alexandra Marques, Tim Newbold, Louise Teh, J. Kolck, Céline Bellard, Stephanie Januchowski-Hartley, and Peter Mumby. Progress towards the aichi biodiversity targets: An assessment of biodiversity trends, policy scenarios and key actions, 11 2014.

[31] L. Liu, W. Ouyang, and X. et al. Wang. Deep learning for generic object detection: A survey. *International Journal of Computuer Vision*, 2020.

[32] Per Lunde, Audun Pedersen, Rolf Korneliussen, Frank Tichy, and Håvard Nes. Power-budget and echo-integrator equations for fish abundance estimation, 12 2013.

[33] Kongsberg Maritime. Simrad es200-7cdk split. `https://www.kongsberg.com/maritime/products/commercial-fisheries/td/200-khz/simrad-es200-7cdk-split/`, 2020.

[34] Md Moniruzzaman, Syed Mohammed Shamsul Islam, Mohammed Bennamoun, and Paul Lavery. Deep learning on underwater marine object detection: a survey. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 150–160. Springer, 2017.

[35] Javier Monroy, Jose-Raul Ruiz-Sarmiento, Francisco-Angel Moreno, Francisco Melendez-Fernandez, Cipriano Galindo, and Javier Gonzalez-Jimenez. A semantic-based gas source localization with a mobile robot combining vision and chemical sensing. *Sensors*, 18(12):4174, 2018.

[36] S. Negahdaripour. *Epipolar Geometry of Opti-Acoustic Stereo Imaging*, pages 1776–1788. 2007.

[37] Erlend Olsvik, Christian M. D. Trinh, Kristian Muri Knausgård, Arne Wiklund, Tonje Knutsen Sørdalen, Alf Ring Kleiven, Lei Jiao, and Morten Goodwin. Biometric fish classification of temperate species using convolutional neural network with squeeze-and-excitation, 2019.

[38] B. Fu P. Zhu, J. Isaacs and S. Ferrari. Deep learning feature extraction for target recognition and classification in underwater sonar images. *IEEE 56th Annual Conference on Decision and Control (CDC)*, 2017.

[39] Hossein Peyvandi, Mehdi Farrokhrooz, H. Roufarshbaf, and Sung-Joon Park. *SONAR Systems and Underwater Signal Processing: Classic and Modern Approaches*. 09 2011.

[40] Hongquan Qu, Tongyang Yuan, Zhiyong Sheng, and Yuan Zhang. A pedestrian detection method based on YOLOv3 model and image enhanced by Retinex. In *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–5. IEEE, 2018.

[41] Sai Rajan, James Miller, Gopu Potty, D.B. Reeder, T.K. Stanton, and Dezhang Chu. Measurements and modeling of the target strength of divers. volume 2, pages 952 – 956 Vol. 2, 07 2005.

[42] Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger, 2016.

[43] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement, 2018.

[44] Alireza Rezvanifar, Tunai Porto Marques, Melissa Cote, Alexandra Branzan Albu, Alex Slonimer, Thomas Tolhurst, Kaan Ersahin, Todd Mudge, and Stephane Gauthier. A deep learning-based framework for the detection of schools of herring in echograms, 2019.

[45] Pia Sethi, Yatish Lele, and Sudipta Chatterjee. *Loss of Biodiversity*. 12 2016.

[46] Yue Shang and Jianlong Li. Study on echo features and classification methods of fish species. *2018 10th International Conference on Wireless Communications and Signal Processing (WCSP)*, pages 1–6, 2018.

[47] Harvinder Singh and Nishtha Hooda. Prediction of underwater surface target through sonar: A case study of machine learning. 02 2019.

[48] Lindsay I Smith. A tutorial on principal component analysis. `http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf`, 2002.

[49] J. Sola and Joaquin Sevilla. Importance of input data normalization for the application of neural networks to complex industrial problems. *Nuclear Science, IEEE Transactions on*, 44:1464 – 1468, 07 1997.

[50] Stavelin. fishy_net. `https://github.com/hersta/fishy_net`, 2020.

[51] Herman Stavelin, Adil Rasheed, Omer San, and Arne Johan Hestnes. Marine life through you only look once's perspective, 2020.

[52] The1only. Ek80 extractor. `https://github.com/The1only/ek80`, 2019.

[53] Yunong Tian, Guodong Yang, Zhe Wang, En Li, and Zize Liang. Detection of apple lesions in orchards based on deep learning methods of cyclegan and yolov3-dense. *Journal of Sensors*, 2019, 2019.

[54] James F. Tressler. *Piezoelectric Transducer Designs for Sonar Applications*, pages 217–239. Springer US, Boston, MA, 2008.

[55] Jack V. Tu. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of Clinical Epidemiology*, 49(11):1225 – 1231, 1996.

[56] Michel Verleysen and Damien François. The curse of dimensionality in data mining and time series prediction. volume 3512, pages 758–770, 06 2005.

[57] Sebastien Villon, David Mouillot, Marc Chaumont, Emily Darling, Gérard Subsol, Thomas Claverie, and Sébastien Villéger. A deep learning method for accurate and fast identification of coral reef fishes in underwater images. *Ecological Informatics*, 48, 09 2018.

[58] Magnus Poppe Wang. Evolving knowledge and structure through evolution-based neural architecture search. *NTNU open*, 2019.

[59] Marcus Widmer. Exploring two approaches to classification of underwater targets, 2019.

[60] Wenwei Xu and Shari Matzner. Underwater fish detection using deep learning for water power applications, 12 2018.

[61] Zhi Xu, Haochen Shi, Ning Li, Chao Xiang, and Huiyu Zhou. Vehicle Detection Under UAV Based on Optimal Dense YOLO Method. In *2018 5th International Conference on Systems and Informatics (ICSAI)*, pages 407–411. IEEE, 2018.

[62] Manell E. Zakharia, François Magand, François Hetroit, and Noël Diner. Wideband sounder for fish species identification at sea. *ICES Journal of Marine Science*, 53(2):203–208, 04 1996.

[63] Zhong-Qiu Zhao, Peng Zheng, Shou tao Xu, and Xindong Wu. Object detection with deep learning: A review, 2018.

NTNU

Norwegian University of
Science and Technology