Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Mechanical and Industrial Engineering

Jens Buvik Bugge

TCN-based Machine Learning Analysis of DOFS Data for Subsea Acoustic Source Localization

Investigation into the Practicability of using Temporal Convolutional Networks on Distributed Optical Fibre Sensor Data

Master's thesis in Mechanical Engineering
Supervisor: Kaspar Lasn
June 2023

NTNU
Norwegian University of
Science and Technology

Jens Buvik Bugge

TCN-based Machine Learning Analysis of DOFS Data for Subsea Acoustic Source Localization

Investigation into the Practicability of using Temporal Convolutional Networks on Distributed Optical Fibre Sensor Data

Master’s thesis in Mechanical Engineering
Supervisor: Kaspar Lasn
June 2023

Norwegian University of Science and Technology
Faculty of Engineering
Department of Mechanical and Industrial Engineering

NTNU
Norwegian University of
Science and Technology

Abstract

Distributed Optical Fibre Sensors (DOFS) are an emerging technology for in-situ structural and environmental monitoring. Optical Fibre Sensing provides high-resolution strain and temperature measurements over long distances and with minimal intrusion to the surroundings. Temporal Convolutional Networks (TCN) are a novel type of Artificial Neural Network architecture that has been proven to outperform LSTM and other earlier algorithms in tasks involving sequential data types. Recent work has demonstrated that TCN can be successfully utilised for the classification of acoustic frequencies using DOFS strain measurements on both free Optical Fibre (OF) and on OF attached to solid structures. This thesis explores the capabilities of DOFS and TCN further, applying them for sound source localisation in a real subsea environment.

Current investigation develops a set of categorical and regression TCN models for classification and continuous value predictions of the distance and direction of an acoustic source relative to the location of the DOFS. The TCN models were trained using subsea DOFS strain measurements collected in a local harbour field experiment. For this data collection in real-life conditions, a test setup was created allowing for the controlled positioning of a submerged buzzer device serving as the acoustic source. The buzzer was placed at horizontal distances between 1 and 3 m in a 120° sector horizontally relative to two vertically suspended DOFS, with measurements taken using Optical Backscatter Reflectometry (OBR) while the buzzer was active. Investigations into the TCN model stability and the effect of sample size and pre-filtering of the strain measurements on model performance were also performed.

The results show that TCN classification models are able to classify the position of the acoustic source correctly. For TCN regression models, the predictions have too large a variance to be practically usable. Both model types are stable for varying dataset sequences and initial parameters, with no significant performance differences for models predicting one or both position parameters. It is further shown that any pre-filtering beyond a slight reduction in strain value peaks leads to reduced performance, indicating that pre-filtering leads to a loss of vital information.

As this is the first known use of the TCN approach for DOFS data analysation outside a controlled lab environment, issues regarding the use of reference measurements needed for strain extraction were revealed. Significantly revised or completely new practices for recording reference measurements are likely necessary if similar subsea monitoring approaches are to be employed going forward.

Sammendrag

Distribuerte Optiske Fibersensorer (DOFS) tilbyr høyoppløselige tøynings- og temperaturmålinger med lang rekkevidde og som har en minimal innvirkning på måleobjektene, og derfor i økende grad benyttes for måling av både konstruksjoner og omgivelser. Temporale Konvolusjonsnettverk (TCN) er en relativt nyutviklet struktur for kunstige nevrale nettverk som har vist seg å overgå LSTM og andre kjente algoritmer i oppgaver som tar for seg analysering av sekvensiell data. Det er tidligere vist at TCN kan brukes til klassifisering av akustiske frekvenser ved hjelp av tøyningsmålinger fra både fritthengende DOFS og DOFS integrert i objekter. Denne oppgaven utforsker potensialet til DOFS og TCN videre ved å anvende dem for lokalisering av lydkilder i et realistisk undervannsmiljø.

Det ble utviklet et sett med TCN-baserte kategoriske og regresjonsmodeller for hhv. klassifisering og prediksjon av kontinuerlige verdier for posisjonen til en akustisk kilde i form av avstand og retning. TCN-modellene ble trent ved hjelp av tøyningsmålinger samlet inn av DOFS under vann i et eksperiment i den lokale småbåthavna. For denne in-situ datainnsamlingen ble det laget et testoppsett som tillot kontrollert posisjonering av en nedsenket summer som fungerte som den akustiske kilden. Summeren ble plassert på avstander mellom 1 og 3 m i en 120° sektor horisontalt relativt til to vertikale DOFS, med målinger tatt av DOFSene ved hjelp av Optisk Tilbakereflektometri (OBR) mens summeren var aktiv. Stabilitetsverifisering av TCN-modellene ble også utført, samt undersøkelser av effekten et varierende antall datapunkter og forfiltrering av målingene brukt til trening hadde på modellenes ytelse.

Resultatene viser at TCN-baserte klassifiseringsmodeller klarer å klassifisere den akustiske kildens posisjonen til en tilfredsstillende grad, mens predikasjonene til de TCN-baserte regresjonsmodellene alle har for stor spredning i verdier til å være praktisk anvendbare. Begge modelltypene er stabile for et varierende antall datapunkter brukt i trening og for variasjoner i de initielle modellparameterene, med kun små ytelsesforskjeller mellom modeller som forutsier en av eller begge posisjonsparametrene. Det vises videre at enhver forfiltrering utover demping av de aller største tøyningsverdiene fører til redusert ytelse, noe som indikerer at forfiltrering fører til tap av vital informasjon.

Dette er den første kjente bruken av TCN-baserte modeller for predikasjon av lydkildeposisjon utenfor et kontrollert labmiljø. Det ble avdekket problemer tilknyttet bruken av referansemålingene nødvendig for beregning av tøyningsverdier. Større endringer eller nye metoder er nødvendig dersom lignende arbeid skal gjennomføres i fremtiden.

Preface

Thank you to Kaspar Lasn for giving me the opportunity to work with Machine Learning and data analysis and for all the guidance along the way. These two last semesters of my five years studying Mechanical Engineering have been an adventure that has allowed me to combine my interest in programming and data science with my studies, and given me knowledge and experiences to lean on as the next chapter soon starts.

I am still not sure what first sparked my interest in mechanical engineering, or computers and programming for that matter. It is not unlikely that motorising a soapbox car with my uncle and the hours in and outside the workshop with my parents at our cabin had something to do with it. All I know for sure is that I wanted to become an engineer since I interviewed my aunt about what one was for a school project in 8th grade. Either way, a big thank you to you all!

My biggest thank you goes to all of the other members of the student teams I've had the joy of being a part of, both in Fuel Fighter at NTNU here in Trondheim and TUFast e.V. Racing Team at TUM in Munich. These opportunities that have allowed me to accumulate experience, knowledge, and friendships both within my own and across other fields of study would not exist without the valuable contributions of each student volunteer, who generously offered their time, energy, and expertise for us to reach the ambitious goals we set.

Finally, thank you to Trondhjems Seilforening for allowing me to borrow their pier at Skansen for the data collection.

Jens Buvik Bugge

June 2023
Trondheim, Norway

Table of Contents

List of Figures

List of Tables

Abbreviations

OF Optical Fibre.

OFS Optical Fibre Sensors.

FBG Fibre Bragg Gratings.

DOFS Distributed Optical Fibre Sensing or Distributed Optical Fibre Sensors. Also referred to as Distributed Acoustic Sensing (DAS) and Distributed Optical Vibration Sensing (DOVS), depending on the application in question.

OBR Optical Backscatter Reflectometry, a technology for measuring DOFS.

SELD Sound Event Localisation and Detection.

DOA Direction of Arrival.

SED Sound Event Detection.

ML Machine Learning.

DL Deep Learning.

ANN Artificial Neural Networks.

CNN Convolutional Neural Networks.

RNN Recurrent Neural Networks. Not to be confused with Recursive Neural Networks.

LSTM Long Short-Term Memory Networks.

TCN Temporal Convolutional Networks.

CM Confusion matrix.

Chapter 1.

Introduction to DOFS, ANNs and SELD

In the following chapter, the concepts behind Artificial Neural Networks (ANNs) and Sound Event Localisation and Detection (SELD) are introduced and put in context with Distributed Optical Fibre Sensing (DOFS). Parts of the ANN and DOFS sections are reworked from the project thesis [1]. The use of DOFS is a promising technology for structural health monitoring and surveillance, among which is threat detection, a topic of increasing importance, illustrated by recent events such as the Nord Stream I sabotage [2]. Several works have shown that DOFS technologies can be utilised for event detection, classification, and localisation along linear structures. Improved structural monitoring, and surveillance of its surroundings, allow for valuable insights into structural health, lowered false nuisance alarm rates and earlier detection of potential threats or failures. This thesis aims to further expand on the knowledge in this field by investigating how an Artificial Neural Network can be combined with DOFS data to locate acoustic sources in a subsea fluid environment.

1.1. Problem Description

Current work on the use of Optical Fiber Sensors (OFS) for acoustic event localisation is mainly focused on acoustic source localisation along linear structures. Determining the distance and direction of acoustic event arrival from the source is necessary to localise an acoustic source in 3D space. Earlier work has been done on estimating the distance between the optical fibre sensor and acoustic source, such as by Landrø et al. [3]. These methods rely on DOFS technologies with sampling rates in hundreds of hertz, at the cost of lower spatial resolution. The high sampling rate is not insignificant, as this allows for the sampling of higher frequencies in line with the Nyquist-Shannon sampling theorem, facilitating the direct use of classical analytical and numerical methods for data analysation. For the detection of more localised sources, DOFS with higher spatial resolutions are likely needed. Technologies such as Optical Backscatter Reflectometry

(OBR) offer this but at the cost of significantly lower sampling rates. For a typical OBR measurement, the sampling rate is 0.1 Hz, i.e. ca. one sample for every 10 seconds. The low sampling rate excludes the possibility of localising acoustic sources with frequencies above this threshold with classical analytical and numerical methods. Therefore, it is suspected that deeper pattern extraction and recognition methods are needed to analyse the high-resolution but low sampling rate OBR measurements.

This master thesis investigates whether it is possible to locate acoustic sources with frequencies in the hundreds of hertz using DOFS measurements performed with OBR, which has a sampling rate of 0.1 Hz, when deep learning models are used for data processing and predictions on experimental data collected in real-life conditions.

1.2. Background

1.2.1. Distributed Optical Fibre Sensors (DOFS)

In the following sections, Distributed Optical Fibre Sensors (DOFS) are explained. The structure and functioning of optical fibres is briefly described, followed by an introduction to the use of optical fibres for structural monitoring applications and a description of relevant measurement technologies.

Distributed Optical Fibre Sensors are a subset of Optical Fibre Sensors (OFS) used for distributed strain and temperature measurements. Distributed strain measurements mean that a DOFS can function as a quasi-continuous strain gauge with up to several kilometres of measuring range, depending on the technology used. The small size of optical fibres, combined with their ability to function both as sensors and signal carriers, means that DOFS has become a highly relevant technology for measuring large structures with minimal intrusion. These properties have led to extensive utilisation of DOFS for structural health monitoring and surveillance, e.g. for threat detection of pipelines [4] and impact detection of composite pressure vessels [5]. Additionally, several recent investigations have demonstrated the use of existing subsea optical fibre networks for environmental sensing, such as earthquake monitoring [6] and the work on whale, ship and storm tracking by Landrø et al. [3].

Optical Fibres (OF)

An Optical Fiber (OF) typically consists of three parts, as illustrated in Figure 1.1. The central core transmits the optical signal. Surrounding the core is a primary layer called cladding, which has a different refractive index than the core. The difference in refractive index enables to keep of the optical signal inside the core due to the phenomena described by Snell's law (Equation 1.1). Here n_1 and n_2 are the refractive indices of the core and cladding respectively, while θ_1 and θ_2 are the resulting input and output angles of a light beam travelling through the material transition interface. By altering the relative

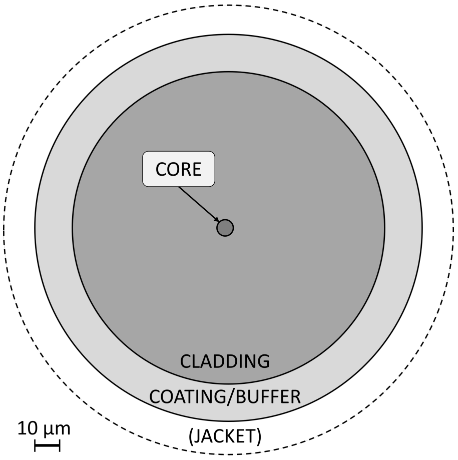


Figure 1.1.: Illustration of the cross-section of a typical single-mode optical fibre

refractive index of the core and the cladding, the input and output angles of a beam hitting the material interface must also change. For a certain ratio between the refractive indices, the beam is reflected inside the core instead of leaving into the cladding at certain incident angles. To protect the surface of the cladding from mechanical damage, a layer called coating surrounds the cladding. This mechanical protection enables it to handle otherwise very brittle glass fibres and extends the strain range it can be subjected to while monitoring, preventing fracturing at low strains.

$$n_1 sin\theta_1 = n_2 sin\theta_2 \tag{1.1}$$

Optical fibre cores and claddings are typically made of silica glass, but studies in recent years have looked into the use of polymer-based OFs. According to Peters [7], polymer-based OF increases strains to failure compared to the traditional silica-based OF, as well as allowing for easier tailoring of the behaviour of polymer-based OF when subjected to varying external factors. Unlike silica glass, polymers can however not be considered inert in all environments, making the silica-based OF superior, and sometimes the only option for harsh measurement conditions. It is also pointed out that a silica glass core is generally more homogenous than its polymer version, reducing the need for calibration and tailoring of equipment based on the specific fibre being used. Higher fragility of silica glass OF does however means that protection against mechanical damage is needed. Therefore, In addition to coating, an outer polymer layer called a jacket is often added for cases where the OF is handled roughly or directly exposed to external factors.

An important classification of OF is done based on the size of the core. Those with smaller diameter cores are typically referred to as single-mode optical fibres, as the small cross-sectional size only allows for a single beam of light to travel through it at any given time without interference. OFs with sufficiently increased diameter can allow for several beams of light to travel through simultaneously and are thus referred to as multimodal optical fibres. Multimodal optical fibres however display increased noise and a decreased range and are therefore generally reserved for shorter measurement ranges. This thesis focuses on utilising single-mode optical fibres for distributed strain sensing.

Optical Fibres (OF) for Sensing Applications

Optical Fiber Sensors (OFS) can be used for measuring strain or temperature, characterising external objects or events. These measurements are based on variations in how the optical signal is altered as it passes through the sensing region. OFS are commonly divided into two exclusive categories depending on their measurement principle [8]. *Extrinsic* OFS uses external components inserted in series, where the external components are the actual sensing component altering the optical signal, while the optical fibre serves as the signal carrier. *Intrinsic* OFS removes the need for external components. Instead, it measures variations in how the optical signal is altered as it passes through the sensing region due to refraction properties in the optical fibre itself. Some intrinsic OFS technologies use specially modified optical fibres, such as Fiber Bragg Gratings (FBG) [9], in order to create identifiable reflections of the optical signal. FBG allows for tailoring the alteration of the optical signal and the position of these alterations to the specific task at hand, but limits the sensing region to the positions at which these modifications are made. For sensing over larger regions, other technologies are utilising inherent variations in the refractive index and light scattering in the optical fibre. These technologies are referred to as *Distributed Optical Fiber Sensors* (DOFS), as the whole length of the fibre can be used as a sensor instead of the sensing region being limited to specific points.

The variations in refractive index and light scattering utilised by DOFS technologies occur due to small, random defects such as variations in density, foreign particles such as dust and changes in the material structure throughout the optical fibre. Additionally, reflections are created in the optical fibre when larger changes in refractive index occur, such as when a splice in the OF (i.e. a welded fusion joint in glass) is misaligned or due to cracks in the core. The result of these effects causes alterations in the phase, frequency and intensity of the optical signal, and are known as the optical scattering effects.

Measurement Technologies

To perform measurements with OFS, a known optical signal is sent through the OF from one end. For interferometer-based technologies, the optical signal is then measured at the other end of the fibre. As an alternative, interrogator-based technologies only require access to one end as they measure backscatter, which are the parts of the optical signal reflected back due to the previously discussed scattering effects. Technologies such as Optical Time-Domain Reflectometers (OTDR) measure the travel time of single optical pulses. In contrast, Optical Frequency-Domain Reflectometers (OFDR) utilise optical signals in the form of sweeps over several frequencies, applying Fourier transform and frequency-based measurement techniques on the measured backscatter to establish the distance to different events along the fibre.

1.2.2. Deep Learning (DL) for Data Analysis

In the following sections, an introduction to Machine Learning (ML) focusing on ANNs for sequential data analysis is given.

Machine learning algorithms have been utilised for data analysis tasks such as image recognition for several years already. However, projects such as ChatGPT [10] or Google Bard [11] have increased the availability of ML to the masses, increasing their popularity and enabling their possibilities to new groups of users. These projects demonstrate how powerful machine learning algorithms have become with the development in recent years, with the ability to handle increasingly complex tasks and problems. This is partly because, unlike traditional analytical and numerical methods, ML-based methods do not require extensive detailed knowledge of all causal mechanisms involved in the modelled phenomena to accurately produce the desired predictions. Instead, ML leverages large datasets combined with statistical methods to discover patterns in the data for the prediction of corresponding outputs [12]. The discovery of underlying patterns and the adaption of the model to them is known as *training* of the model, or that the model is *learning*. If the correct output of the data points in the set used for training is known, the process is referred to as supervised learning. In that case, the training typically attempts to minimise the discrepancy between the model prediction for given data points and their known true labels. For cases where the data is not labelled, i.e. when the true output is unknown, unsupervised learning is performed. Training of the model then aims at minimising a defined loss function serving as a score of model performance.

This thesis uses supervised learning of Temporal Convolutional Networks (TCN) for data analysis. TCN is a new set of Artificial Neural Network (ANN) architectures first proposed by Lea et al. in 2017 [13]. It has been proven to outperform Long Short-Term Memory (LSTM) and other previously used models for analysation of sequential datasets. This performance advantage makes TCN also interesting for application in tasks involving DOFS data. The application of TCN on DOFS data has been previously demonstrated in the project thesis by the author of this master thesis [1]. There, it was shown that a TCN could successfully classify acoustic disturbances on two tubular structures for frequencies up to hundreds of Hertz, using DOFS measurements with a sampling rate of 0.1 Hz.

Artificial Neural Networks (ANN)

Deep Learning (DL) is a subset of Machine Learning (ML) methods that cover models created to find deeper patterns in data. Unlike traditional ML, DL models are created so that no prior feature extraction is necessary. Instead, DL models learn to extract relevant features from the raw data during training, a concept known as *feature training.* This is possible due to its use of complex Artificial Neural Networks (ANN), which are a subset of ML models that, as the name suggests, are built to resemble how the neurons in the human brain are connected and work [14].

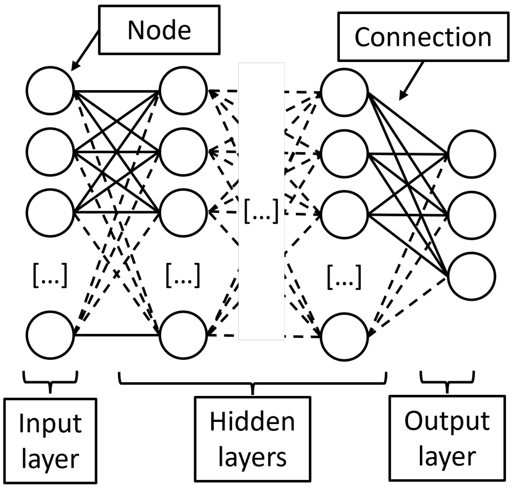


Figure 1.2.: Example of a generic, fully connected feed-forward ANN structure. Adapted from [1].

An ANN consists of *nodes*, each computing a *weighted* sum of a specified set of inputs and an extra value known as a *bias*, with both the weights and the bias specific to each node. The structure of ANN nodes depends on the model architecture being used, but generally, the nodes are structured in sequential layers as described in Figure 1.2. Each node in a given layer takes the outputs of the previous layer as its inputs and outputs the result of its calculation to the next layer. Which nodes are used as inputs from a previous layer depends on the architecture in question and are referred to as its connections. Standard terminology further refers to the first layer as the *input layer* and the last layer as the *output layer* of the model. The structure of the input layer of a model is determined by the structure of the data it should analyse. Similarly, the number of nodes in the output layer corresponds to the number of output parameters the model should return. ANN models used for DL have several layers between these input and output layers, referred to as *hidden layers*, as the values here are not immediately accessible outside the model. These hidden layers enable the model to learn 'deeper' features from the raw training data, as the combination of several hidden layers and their connections to previous layers allows the model to recognise complex patterns in the given input. Traditional ANNs follow this pattern of feeding the outputs of the prior layer as inputs in the next sequentially and are known as *feed-forward neural networks.*

Common Model Architectures

As previously discussed, the structuring of and connections between the nodes of an ANN model, known as the *model architecture*, decides its behaviour and capabilities. While numerous architectures exist, there are two main categories that define how a model handles feature extraction. Convolutional Neural Networks (CNNs) cover architectures commonly used for image recognition and similar types of pattern recognition [14, pp. 296–329]. To achieve this, CNNs have at least one hidden layer that performs convolutions on its inputs in place of the weighted summation normally used. The size of these convolutions is typically defined by a kernel size, with the resulting convolution termed a *kernel* or a *filter.* Each node can have a stack of these kernels, all with different values. Kernels are then applied to all input values, resulting in a scaling of the number of inputs by the number of kernels in each node, with each kernel allowing for the node to extract a different feature from its input data. Before passing values to the next layer, a weighted sum of these scaled inputs is computed and normalised as for other ANN nodes.

While CNNs were developed to recognise visual patterns, Recurrent Neural Networks (RNNs) were developed to recognise patterns in sequential data of varying lengths [14, pp. 333–336]. In its simplest form, this is done by adding feedback loops between layers of a traditional feed-forward neural network, allowing the result of previous calculations to influence the next. The feedback loop is often added as a hidden state vector, serving as the 'memory' of the model, keeping it between steps in the input sequence. The Recurrent part of the RNN name further points to how it handles inputs of varying lengths. This is done by duplicating the network structure with all its weight and biases to match the input sequence length, unravelling the feedback loops into a structure that instead feeds the output from one step into the next.

Current Research in Optical Fibre Sensing (OFS) and Deep Learning (DL)

There is a large use of DOFS for infrastructure monitoring, with several recent works utilising ML and DL-based approaches for the processing of DOFS data [4], [15]–[19]. These generally aim at developing systems for threat detection where disturbances, i.e. events along the optical fibre or structure, are detected, classified and located by analysation of DOFS data using DL-based approaches to give early warnings of potential threats and other problems. For pipelines, Peng et al. [20] have demonstrated how DOFS sensing systems can be extended to include the detection of pipe degradations such as corrosion. Huynh et al. [21] have developed a system for leak detection in addition to external intrusion events. Zhongqi et al. [22] developed a framework based on CNN and LSTM for intrusion detection on a high-speed railway using an optical fibre cable buried between two tracks. This method of utilising an external optical fibre cable as the DOFS for surveillance has also been demonstrated in several other works [23]–[25]. Similar to the railway system by Zhongqu et al., Khan et al. [26] proposed a system for tracking vehicle movements on roads, using a Generative Adversarial Network (GAN) for denoising the DOFS measurement data.

The use of optical fibres and DL-based processing for structural monitoring on a smaller scale has also been demonstrated, such as a CNN-based machine learning framework by Li et al. [27] for monitoring lithium-ion batteries using optical fibres with Fiber Bragg Gratings (FBG). Similar work has been done by Dhanalakshmi et al. [28] for temperature monitoring of solar photovoltaic panels using FBG for measurements and random forest regressor modelling for analysation. Zhuang et al. [29] demonstrated the use of embedded FBG and various ML models for small-scale impact monitoring in helmets to give early detection of concussive events.

In addition to the investigations demonstrating practical applications of DL-based modelling combined with DOFS measurements, several works address the requirement of processing large training datasets that affect most DL-based approaches. Shiloh et al. [30] propose using a secondary GAN to generate additional training data from a smaller labelled dataset. Shi et al. [31] utilised Transfer Learning to reduce the dataset need. Here, the ANN model is pre-trained with a more general, but still similar, dataset. The pre-trained model can then be further specialised to the specific use case afterwards with a specialised dataset, reducing the size requirement of the dataset specialised to the particular application. They demonstrated that by freezing the parameters in the first layers after pre-training using the bigger general dataset, the model could quickly be specialised by training on the smaller specialised dataset. Further performance improvements have also been shown in the work by Shi et al. [32], implementing a Support Vector Machine (SVM), an ML classification method, as the final layer in a CNN model to improve the overall model performance. Bublin [33] took this a step further, combining DL models with classical machine learning and knowledge-based approaches to improve the overall model performance. Finally, some works have demonstrated the use of DL for pre-processing before applying traditional knowledge-based methods, such as the investigation by Liehr et al. [34] using CNNs for real-time denoising of long-distance strain measurements using DOFS.

1.2.3. Sound Event Localisation and Detection (SELD)

Sound Event Localisation and Detection (SELD) covers methods and systems that aim to establish the location and active time of one or more acoustic sources, referred to as *monophonic* and *polyphonic* systems respectively [35]. This is done by dividing the process into two parts. Sound Event Detection (SED) aims at identifying the temporal location of acoustic events, e.g. by determining start and stop times. Localisation is typically done by establishing the Direction of Arrival (DOA) to identify the spatial location of the acoustic source of interest. Monitoring systems can additionally include classifying the acoustic event as a part of their operation. Which of these are implemented in the resulting system depends on the use case. For example, voice assistants (e.g. Google Assistant or Alexia) require SED to detect prompts for which to generate responses, while surveillance systems can benefit from both SED, DOA and classification for identifying potential threats.

The methods used by SELD systems to detect acoustic signals depend on the specific use case but typically include some form of an acoustic sensor array. While SED in principle only requires a single acoustic sensor, a larger set of sensors is needed for DOA. For 360° localisation in the horizontal plane, full azimuth arrays are needed, with additional sensors necessary to detect elevation and distance to obtain a full three-dimensional DOA system. The type of sensor also depends on the use case, with microphones typically used for SELD in the air. For SELD in solid materials, Surface Acoustic Wave (SAW) sensors can be used to detect the propagation of acoustic waves along the material surface. In water and other liquids, hydrophones or other piezoelectric sensors are typically needed. Data from acoustic sensors is then analysed to perform classification, DOA or SED estimations using machine learning or knowledge-based mathematical, statistical and analytical methods such as the MUSIC or Approximate Maximum Likelihood (AML) algorithms.

ML Methods

According to a review on sound source localisation methods by Desai and Mehendale [35], the use of Deep Learning (DL) and Artificial Neural Networks (ANN) is an emerging and promising approach to accomplish SELD. DL-based SELD methods do not require detailed mathematical and statistical models about investigated phenomena to perform DOA and SED estimations. As Desai and Mehendale further point out, DL-based methods reduce the need for pre-processing, such as feature extraction and denoising. But they also discuss that most DL systems developed are more sensitive to environmental changes, such as noise images, than knowledge-based methods.

Gurirguis et al. [36] developed the SELD-TCN model architecture, claiming it to be the first use of the Temporal Convolutional Network (TCN) ANN structure for SELD applications, in place of CNN and RNN structures typically used in similar earlier investigations. This model has further been improved on by Song et al. [37], who discovered that decoupling the training of the SED and DOA parts of the model individually improves the overall model performance. These preceding works have demonstrated that TCN-based ANNs are a promising approach for achieving efficient SELD systems.

DOFS for Acoustic Source Localisation

In addition to DL-based methods, there are several other threat detection systems for linear infrastructure monitoring utilising classical statistical or analytical methods for processing of DOFS data [20]. The detection systems are, in practice, all SELD systems with the DOFS functioning as a SAW detector for measuring acoustic waves in the form of vibrations in the structure itself or in nearby vicinity through ground vibrations.

Recent works by Landrø and Bouffaut et al. [3], [38], along with similar work by Rivet et al. [39], have demonstrated the use of unused optical fibres in subsea cables, also known as *dark fibres*, as DOFS for tracking whales and ships. Both projects convert measured strains to acoustic pressure waves, the frequencies of which are then analysed to establish position and velocity using classical numerical methods. Their results demonstrate the potential of using DOFS for subsea SELD applications.

1.3. Scope

This project aims to establish whether Deep Learning in the form of Artificial Neural Network models can be utilised to localise acoustic sources in a subsea environment, specifically estimating the distance to the source and the direction of arrival in the horizontal plane. The investigation is limited to source localisation and does not include event detection. Further, this project investigates how data pre-filtering, sample sizes and the number of output nodes affect the performance of these ANNs.

The data used for these investigations is recorded from a field experiment using Optical Backscatter Reflectometry (OBR) equipment from Luna Technologies in the form of sets of quasi-continuous strain measurements along the DOFS.

The ANN models trained are categorical and regression models, all with a structure based on Temporal Convolutional Networks (TCN). Two hyperparameter sets were developed to assess the robustness and reliability of categorical and regression model results. TCN models were not further optimised as this was considered not to bring additional insight related to the main goals of this project, as such optimisation is specific to each problem and highly resource-intensive.

1.4. Structure

This thesis investigates the TCN model performance for acoustic source localisation. The work progresses from data collection through preprocessing and model training to the final result analysis and evaluation. For this, the thesis is structured as follows.

chapter 2 covers the materials and methods used, as well as the description of specific implementation for each part.

Distributed optical fiber sensors (DOFS) are explained, including the procedures for measurement and strain extraction. Problems related to measurement noise and filtering are discussed.

Temporal Convolutional Networks (TCN) and its specific architecture and operation is covered. The process of hyperparameter optimisation, training, post-processing and evaluation is explained. An overview of the infrastructure and the process used for model development and training is also presented.

Experimental setup describes the setup used for data collection, including structural parts, the acoustic source and the two DOFS used. The location of and procedure for data collection are also described.

chapter 3 presents the results from analysing DOFS data with final TCN models and the studies related to sample size and filtering.

chapter 4 discusses the results in the context of the project objectives.

Finally, an overall **conclusion** is presented in **chapter 5**, along with thoughts and suggestions on **further work** in **chapter 6** to expand the knowledge on TCN and DOFS for the localisation of acoustic sources.

Chapter 2.

Materials and Methods

2.1. Distributed Optical Fiber Sensors (DOFS)

2.1.1. Optical Backscatter Reflectometry (OBR) and Optical Fibre (OF)

Optical Backscatter Reflectometry (OBR) is an interrogator-based measurement technique for DOFS, which utilises Rayleigh Backscatter for measuring pertubation in the optical fibre [41], a phenomenon which arises when the inhomogenities in the fibre core that scatter the optical signal back are smaller than the wavelength of the signal. Due to Rayleigh Backscattering being an elastic scattering phenomenon, meaning that no energy is drained from the optical signal due to scattering, the frequency of the optical signal is not altered [42, p. 215]. OBR utilises an implementation of Optical Frequency-Domain Reflectometry (OFDR) for its measurements, with an additional polarisation of the optical signal for improved accuracy.

The interrogator device used for data collection is the OBR 4600, produced by Luna

Table 2.1.: Spesifications for the SMB-E1550H optical fiber used in the experiment [40].

Parameter	**Value**
Core diameter	6.5 μm
Cladding diameter	125 ± 2 μm
Coating diameter	155 ± 15 μm
Jacket diameter	~ 2 mm
Central wavelength	1550 nm
Attenuation	≤ 0.7 dB/km
Operating temperature	-65 to 300 °C
Short-term bend radius	≥ 10 mm
Long-term bend radius	≥ 17 mm

Table 2.2.: Selected parameters for the Luna Inc. OBR 4600 interrogator device, adapted from [43].

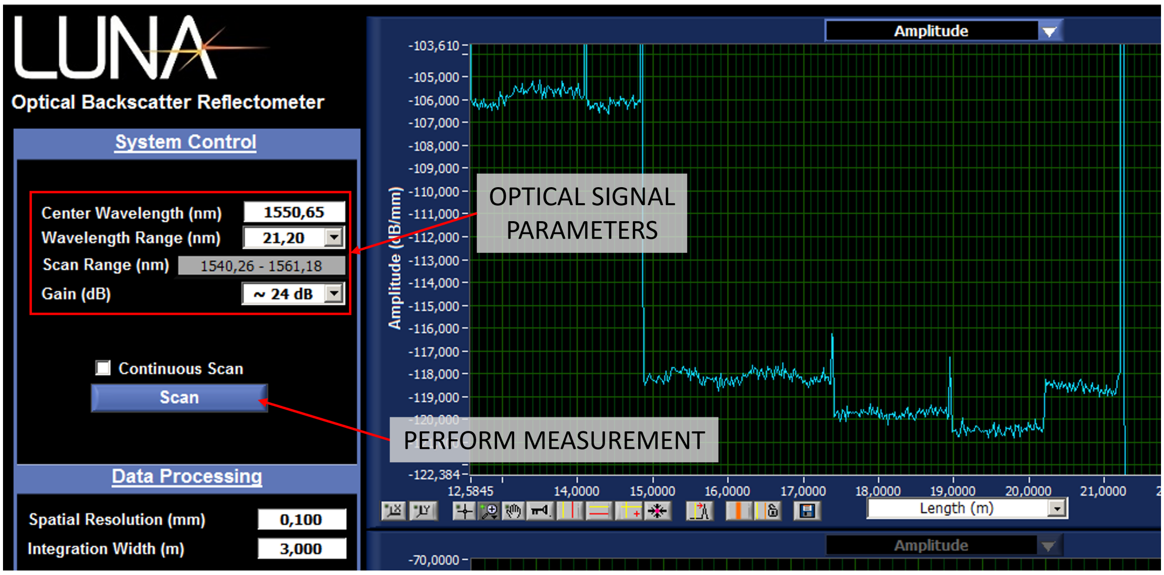
Parameter	**Value**	**Explanation**
Max range	2 km	Maximum measurement distance
Standard range	70 m	Typical measurement distance
Sampling resolution (70 m)	20 μm	Precision of data points in 70m range
Dead zone	2x resolution	Area behind each sampling point where no measurements can be performed
Spatial resolution	± 1.0 cm	Precision of location measurements
Temperature resolution	$\pm 0.1°C$	Precision of temperature measurements
Strain resolution	± 1.0 $\mu\epsilon$	Precision of strain measurements
Avg. scan time (70 m)	9.7 s	Typical time to scan a 70m long DOFS

Technologies. Table 2.2 lists relevant parameters for the capabilities of this device, along with characteristic values and short explanations. Specifications for the Optical Fibre used as DOFS is given in Table 2.7. The results in the project thesis [1] found that using a free jacketed optical fibre as the DOFS for frequency classification resulted in the highest performing Artificial Neural Network (ANN) models, and it was therefore decided to use jacketed OF for the experiment in this thesis.

2.1.2. Measurement Procedure

The OBR 4600 interrogator is controlled from a PC using the software *OBR v3.13.0* from Luna Technologies, which stores the recorded measurements as proprietary binary *.obr*-files. Figure 2.1 shows the interface of the program used for controlling measurements, while Table 2.3 gives an overview of the parameters for controlling the optical signal.

For each specific measurement, a reference measurement has to be performed while the DOFS are in their non-excited state. While recording the reference, none of the



Center Wavelength (nm) 1550,65

Wavelength Range (nm) 21,20

Scan Range (nm) 1540,26 - 1561,18

Gain (dB) ~ 24 dB

□ Continuous Scan

Scan

Spatial Resolution (mm) 0,100

Integration Width (m) 3,000

Figure 2.1.: OBR v3.13.0 interface for controlling the OBR 4600 interrogator.

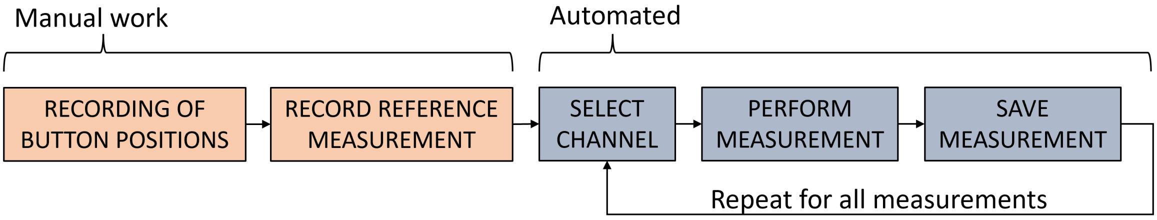

Manual work
Automated
RECORDING OF BUTTON POSITIONS
RECORD REFERENCE MEASUREMENT
SELECT CHANNEL
PERFORM MEASUREMENT
SAVE MEASUREMENT
Repeat for all measurements

Figure 2.2.: Overview of OBR measurement procedure.

events one wishes to measure the effect of should be present and influencing the DOFS. Reference defines the unaffected state of the DOFS, and is later used to calculate strains. Measurement traces are then recorded while the DOFS are submitted to the different desired events, i.e. measurement states. The recording of measurement traces was automated using a Python script. Figure 2.2 gives an overview of the complete process for taking measurements with the OBR.

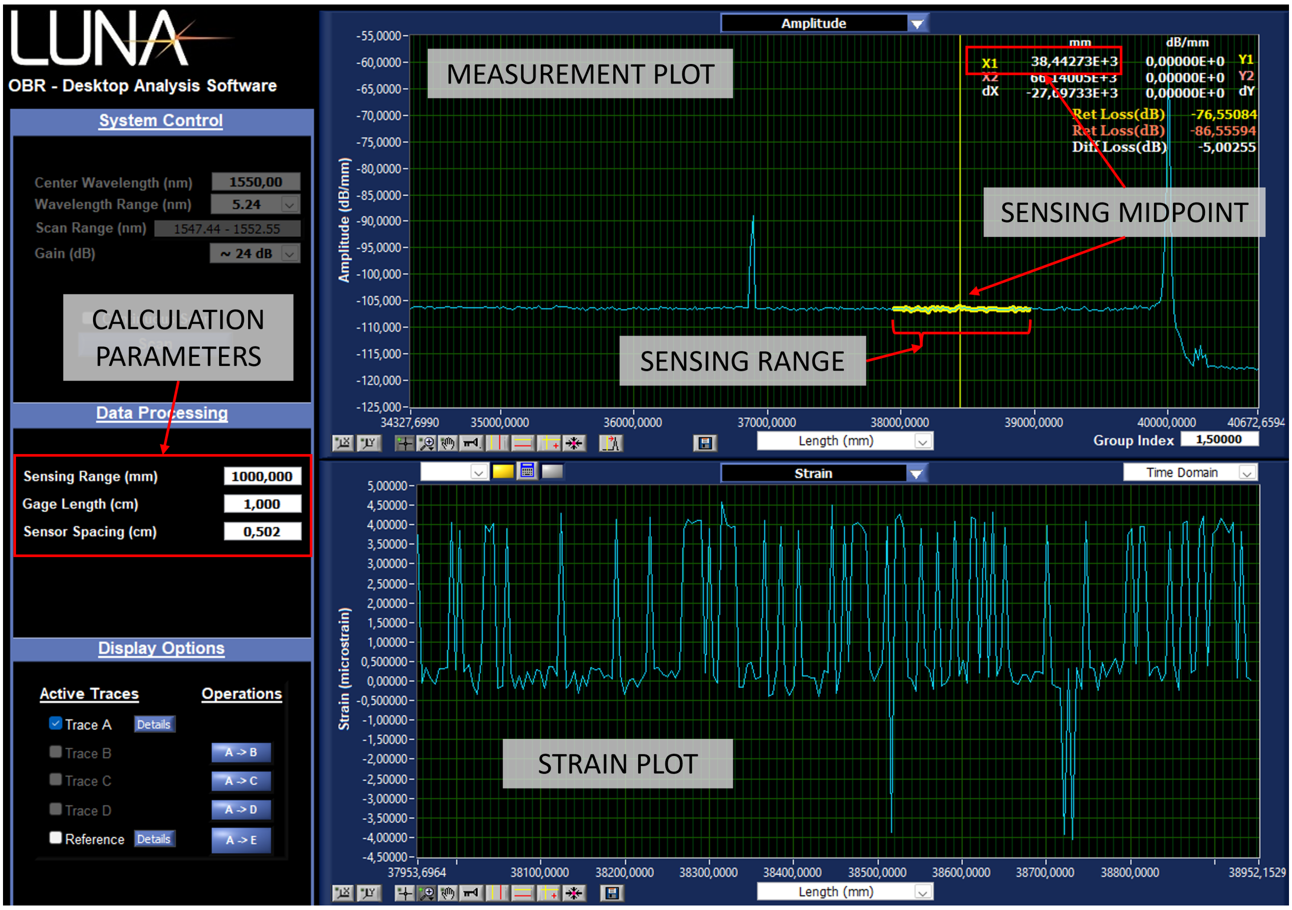
2.1.3. Strain-Extraction in Data Pre-Processing

The recorded measurement traces in the form of *.obr*-files are analysed using a second software from Luna Technologies, *OBR Desktop v3.13.0*. An overview of the interface used is shown in Figure 2.3. For each DOFS, the corresponding reference trace is loaded. Measurement files are individually loaded, calculating strains for each measurement state using the parameter values defined in Table 2.3 and exporting the output as CSV files. Note that the sensing range and midpoint location are specific to each DOFS and were therefore set to their respective value in Table 2.7. The strain extraction process was automated with a Python script similar to that for the measurement procedure.

Table 2.3.: Parameters for measurements and strain calculations using the OBR 4600 device and accompanying software.

Measurement		
Parameter	**Value**	**Explanation**
Center Wavelength	1550.65 nm	Center wavelength of the optical signal
Wavelength Range	21.20 nm	Width of the wavelength range of the optical signal
Gain	~ 24 dB	Intensity of the optical signal
Strain extraction		
Gauge length	10 mm	Width of each virtual strain gauge where strain is averaged over
Sensor spacing	5 mm	Spacing between the centers of virtual strain gauges
Sensing range	-	Width around the sensing midpoint for which strains are calculated
Sensing midpoint	-	Midpoint for strain calculation region

Note: The value of sensing range and midpoint are dependent on the specific DOFS being used and can be found in Table 2.7.



OBR - Desktop Analysis Software

System Control

Center Wavelength (nm) 1550,00
Wavelength Range (nm) 5.24
Scan Range (nm) 1547.44 - 1552.55
Gain (dB) ~ 24 dB

Data Processing

Sensing Range (mm) 1000,000
Gage Length (cm) 1,000
Sensor Spacing (cm) 0,502

Display Options

Active Traces
☑ Trace A
□ Trace B
□ Trace C
□ Trace D
□ Reference

Group Index 1,50000

Figure 2.3.: OBR Desktop v3.13.0 interface for strain extraction.

Figure 2.4 gives an overview of the overall procedure used for strain extraction.

As all measurements were taken in the span of a few hours per session, with a reference measurement for each new position, temperature changes were assumed to have a negligible effect on the strain output, and were therefore not taken into account in the strain extraction process.

2.1.4. Methods for Analysation

Extracted quasi-continuous strains along the fibre allow for the use of DOFS as an alternative to strain gauges, a use case demonstrated in several works, e.g. Wang et al.[44]–[47]. Further, combining DOFS strain data with detailed positioning for each DOFS allows to create detailed maps of the structure. As an example, Sæter et al. [5] demonstrated how the strain measurements from DOFS embedded in a composite pressure vessel were used to create relatively detailed maps of the impact damage in the structure. On a much larger scale, the application of SELD methods on DOFS data was used to track whales, ships, storms and earthquakes in [3][38].

As presented in Section 1.2.2, several Machine Learning-based (ML) approaches have

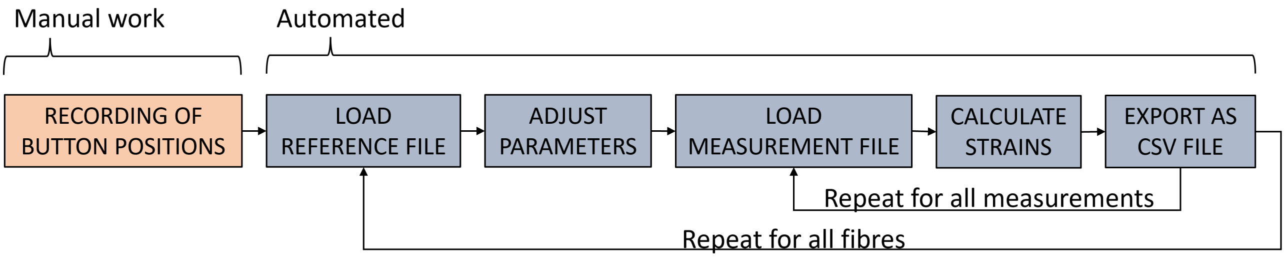


Figure 2.4.: Overview of the OBR strain extraction procedure.

been developed for the analysation and use with DOFS data replacing knowledge-based analytical and numerical methods. While some ML approaches, such as in the work by Chen and Xu [25], depend on filtering and denoising similar to the classical methods, it has been shown that ML methods typically can be utilised without the need for noise removal in pre-processing. In the master thesis of Usenco [48], it was shown how the ML algorithms classified several different sources of acoustic disturbances by analysing the raw, un-filtered DOFS strain data. The reduced number of steps in pre-processing due to the inclusion of denoising as a part of feature extraction is an advantage for ML methods. By creating models that are able to extract information from noisy data, the problem of information loss as discussed in Section 2.1.5 is addressed as the model in practice ignores the parts of the data that only contains noise, rather than having to removing them manually.

2.1.5. The Problem of Measurement Noise

Wang et al.[44] address noise-related problems of strain measurements in transition regions, where the fibre ingresses and egresses from a solid structure. An easy solution to noisy measurements is to manually remove outliers and significantly noisy regions. While this has been proven as a viable solution for basic analysation of DOFS strain measurements, there is the obvious risk of removing useful information along with the outliers. As Wang et al. further discuss, an alternative is to develop specific filtering to deal with the noise in DOFS measurements. This does, however, contain the risk of removing useful information from the data when filtering.

The models developed in this thesis only take a series of strain values as input, and does not specifically consider the spatial position of these, as they were the same for each measured value across all measurements. Removing outliers was therefore not considered a feasible option, as this would shift the overall signal and not just dampen any outliers, potentially removing useful measurement data. Instead, Gaussian filters with different sigma values were added to the data loading stage in the pipeline as an attempt to remove potential outliers. A Gaussian filter calculates a weighted average for a range of values to either side of the current value being calculated, with the weights being the value at the Gaussian curve with standard deviation σ and centred on the current value. As a result, any outliers in the form of spikes in sequential data will be dampened without shifting the remaining strain signal.

A frequency band filter was also implemented to investigate whether specific frequency ranges contained vital information used by the models for predictions, similar to the Gaussian filter. For the band filter, the spatial strain measurement was first converted to the frequency domain using a Fast Fourier Transformation (FFT). A low-pass and a high-pass filter was then applied to remove spatial frequencies outside the defined band before the remaining signal was converted back to the spatial domain by using an inverse FFT. The result is a sinus wave approximation of the original signal, but only containing frequencies within the defined frequency band. It should be noted that the frequencies here are spatial frequencies, where the inverse of the spatial sampling resolution r_s of the equipment, defined in Table 2.3, is defined as the sampling rate f_s, $f_s = \frac{1}{r_s}$. Because of this, the frequency values do not directly translate to temporal frequencies, but must rather be seen as a scaled version, with values ranging from 0 to the highest detectable spatial frequency possible when applying the FFT on the discrete spatial signal, i.e. the strain measurement.

The effect of both the Gaussian and the band filter is illustrated in Figure 2.5, where a randomly selected DOFS strain measurement is plotted along with the same signal treated either with Gaussian filters with different sigma values or band filters for different frequency bands.

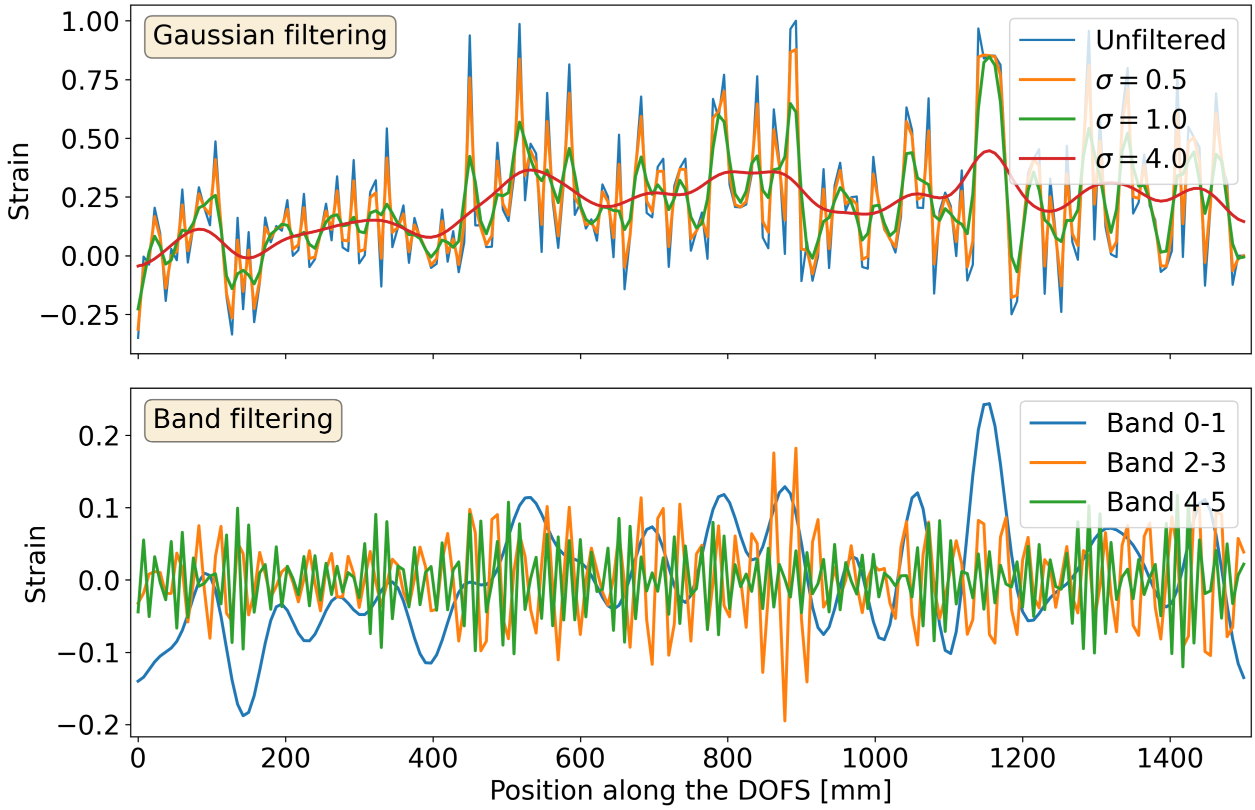


Figure 2.5.: Plot of a randomly selected strain measurement where Gaussian and band filters with different parameters are applied.

2.2. Temporal Convolutional Networks

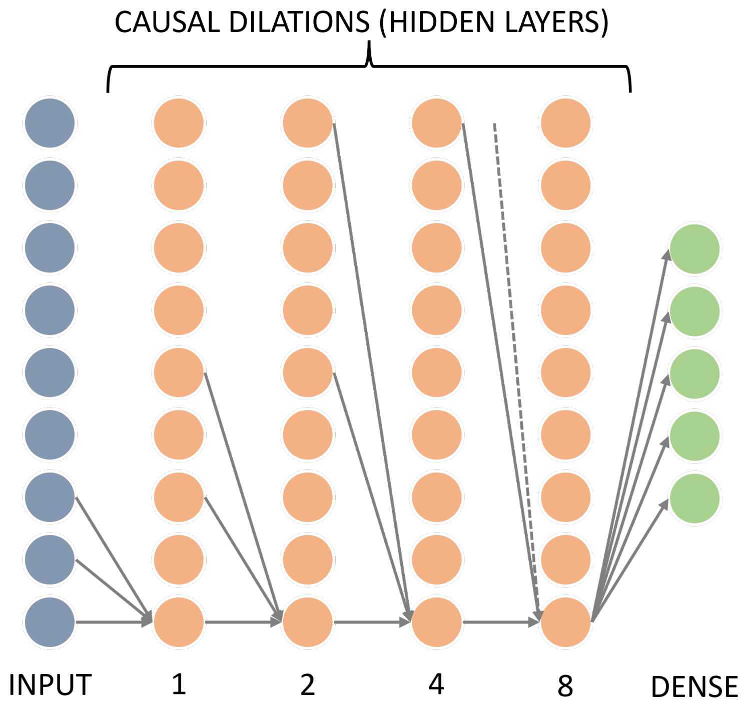


Figure 2.6.: Illustration of connections between layers of nodes in a simple TCN structure with kernel size 3. The number below each hidden layer denotes the amount of dilation for the respective layer. Adapted from [49].

Temporal Convolutional Networks (TCN) are a set of ANN architectures that combine the principles from Recursive Neural Networks (RNN) and Convolutional Neural Networks (CNN) to achieve deeper pattern recognition on sequential data. However, rather than utilising feedback connections to recognise patterns like typical RNNs, each quantity is input in parallel as separate neurons in the input layer. This allows for parallel processing of each quantity rather than keeping the model waiting for previous quantities to finish processing before it can continue with the next. As a result, TCN has been proven to outperform other, more traditional ANNs for temporal, spatial and other sequential types of data [50].

An illustration of a simple TCN structure can be seen in Figure 2.6. The structure of the model is as follows. (1) Input layer with size equal to the largest entry of the dataset. (2) Convolutions occur through several hidden layers, with each neuron containing a set of filters and each filter being a kernel of a given size for feature extraction from previous layers. (3) For each convolution layer, dilation is increased, meaning a given number of nodes are skipped between each connected node in the previous layer. Typically these dilations are a power of 2, increasing incrementally for each layer. As a result, the number of neurons in each layer and the number of connections between layers are constant, but with an increasing spacing between each connected node to a given node in the next layer. For connections that reach outside the length of the previous layer as the dilation increases, the input value of this connection is set to zero, allowing for inputs of varying lengths. Sequential duplicates of dilation layers, termed additional *stacks*, can be added to increase the receptive field of the model further. (4) The final dense layer serves as the output of the model.

The aforementioned receptive field of a TCN model describes how 'far' an output node of the TCN block in the model is able to 'see', i.e. the number of time steps or neighbouring input nodes that affect the value of the specific output node. In order for the model to see the whole input data series, the receptive field must be at least equal to the size of the largest sequence in the data set to be analysed. Equation 2.1, adapted from the Python Library keras-tcn [51], shows how to calculate the receptive field R_{field} of a TCN block. Here, K_{size} denotes the kernel size of the model, N_{stack} the number of dilation stacks and d_i the dilation of layer i in D, the set of all dilation layers in one stack.

$$R_{field} = 1 + 2 \cdot (K_{size} - 1) \cdot N_{stack} \cdot \sum_i d_i \ \forall \ i \in D \tag{2.1}$$

2.2.1. Hyperparameters

Hyperparameters are non-trainable parameters of an ANN model that are used for defining its overall structure and behaviour [14]. Unlike other model parameters, such as the weight and bias parameters for each node, hyperparameters are not defined as trainable and are kept constant during the training process of the model. They are instead optimised in a pre-training process, where different hyperparameter values are tested to find the optimal parameter set for the specific application. Hyperparameter values are determined by monitoring the performance of a test model as they are adjusted, selecting the one that results in the highest performance. For sets of several different hyperparameters, grid searches are typically performed. Sets of values for each hyperparameter to be optimised are defined, and models are then tested using different combinations of these sets, again selecting the set that results in the highest-performing test model.

The values of some hyperparameters are in practice given by the task, such as the number of input and output nodes, which are determined by the size of the input data series and the number of outputs needed, respectively. Others have a range of possible values, where the specific choice of value typically affects model performance to some degree. For models relying on feature extraction from the input data, such as the TCN, hyperparameters related to *generalisation* can have a significant effect on performance. The generalisation of a model refers to its ability to perform well on new data, i.e. data that was not part of the training process. For a TCN, this is mostly related to how deep the pattern recognition and feature extraction from the input is. Too low generalisation means an overly detailed feature extraction, where the model interprets occasional noise in the data as valid inputs. As a result, the model would be *overfitted* to the training data. This is indicated by lower performance in validation tasks, despite the training performance likely being higher compared to models with higher generalisation levels. In the opposite case, an overly generalised model means the feature extraction is not detailed enough, resulting in a model that is *underfitted*, not able to distinguish important features in the data, and thus has an overall lower performance. For TCN, the main hyperparameters affecting the model generalisation level are:

- filters affect the number of unique features a model is able to extract;
- kernel size controls the size of local patterns the model is able to detect;
- dilations are the biggest contributor to the receptive field of the model and a measure of how much of the input data series can affect a given data point;
- dropout rate allows for increasing the generalisation of the model by ignoring the output of randomly selected nodes.

In the project thesis [1], three hyperparameter sets were established for a similar problem, each with the basis in the recommendations and standard values from the documentation in the Python library Keras-TCN [51]. These hyperparameter sets aimed for (i) reduced computational cost, (ii) low generalisation, and (iii) high training effort, respectively. Testing of models using these three sets showed relatively equal performance, with only minor differences in overall accuracy and loss values. This demonstrates that the model architecture used is robust with respect to the selection of, and small changes in, hyperparameters. As the architectures used in this thesis are similar to that of the project thesis, it is assumed that the observed robustness is transferrable. It was therefore decided to only develop one hyperparameter set for each of the model types in this thesis, freeing computational resources for other tests. An overview of all hyperparameters defined, along with a short explanation and the values used is listed in Table 2.4.

2.2.2. Activation Functions

In addition to the weighted summation and normalisation of the inputs of a given node, *activation functions* are used to scale the output of the node. No scaling is equal to using the Identity function (Equation 2.2), but due to the exploding and vanishing gradient problems related to training, as discussed in the master thesis of Usenco [48], the output is typically normalized before it is passed on. For filtering out negative values, the Rectifying Linear Unit (ReLU) function presented in Equation 2.3 can be used. The Logistic (Equation 2.4) and Tanh functions (Equation 2.5) scale the output to $(0, 1)$ and $(-1, 1)$ respectively, both weighting the scaled value towards the ends of their respective intervals due to the inherent nature of their sigmoid shape. Figure 2.7 shows how the aforementioned activation functions perform in the argument interval $(-10, 10)$.

Table 2.4.: Overview of TCN hyperparameters for the categorisation (Cat) and regression (Reg) models respectively. Reworked from [1].

Name	**Explanation**	**Value**	
	Model hyperparameters	**Cat**	**Reg**
Classes	Number of output nodes in the final dense layer	9 (3)	2 (1)
Filters	Number of filters in each node, each calculating a differently weighted sum between the values from its input nodes	64	64
Kernel size	Number of input nodes for each filter	3	5
Dilations	The spacing between input nodes from the previous layer	1,2,4,8 16,32,64	1,2,4,8
Dropout rate	Chance of a weight of an input being permanently set to zero, in practice ignoring the value of the input	0.2	0.1
Stacks	Number of repeated sets of dilation layers	1	1
	Algorithm hyperparameters		
Epochs	Number of runs through the entire training dataset	30	50
Batch size	Number of data points to simultaneously calculate while training, the average of which is used for adjusting the model parameters	3	3
Learning rate	Determines the initial ’step size’ in the direction that improves model performance	0.0003	0.0003
Constant epochs	The number of epochs to train with the initial learning rate before starting exponential decay	5	5
Decay rate	Determines the rate of exponential decay from the initial learning rate after the number of constant epochs has been reached	0.4	0.4
Validation split	The fraction of the training data points reserved for validation in each epoch	0.2	0.2

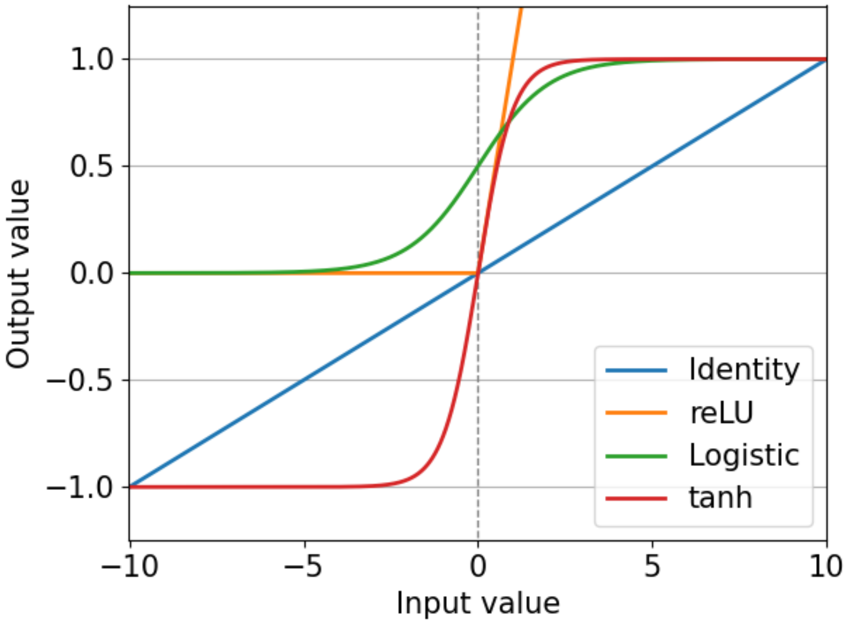


Figure 2.7.: Common activation functions used in ANN nodes, plotted for the argument interval $(-10, 10)$.

$$Identity(x) = x \ \in (-\infty, \infty) \tag{2.2}$$

$$ReLU(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases} = max(0, x) \ \in [0, \infty) \tag{2.3}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \ \in (0, 1) \tag{2.4}$$

$$Tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \ \in (-1, 1) \tag{2.5}$$

Output Scaling

It can further be noted that the output dense layer nodes are typically scaled using separate activation functions. As classification tasks are represented by *one-hot* labels, where the correct category is marked with value 1 while the rest are set to 0, a *softmax* function (Equation 2.6) is generally applied. This scales each output z_i of n nodes in the output layer such that the sum of all outputs equals 1. The scaled output values then represent how 'sure' the model is (resembling probability) that a given data point belongs in each corresponding category. For regression problems, the choice of output activation function varies more, depending on the range and type of values the model should predict.

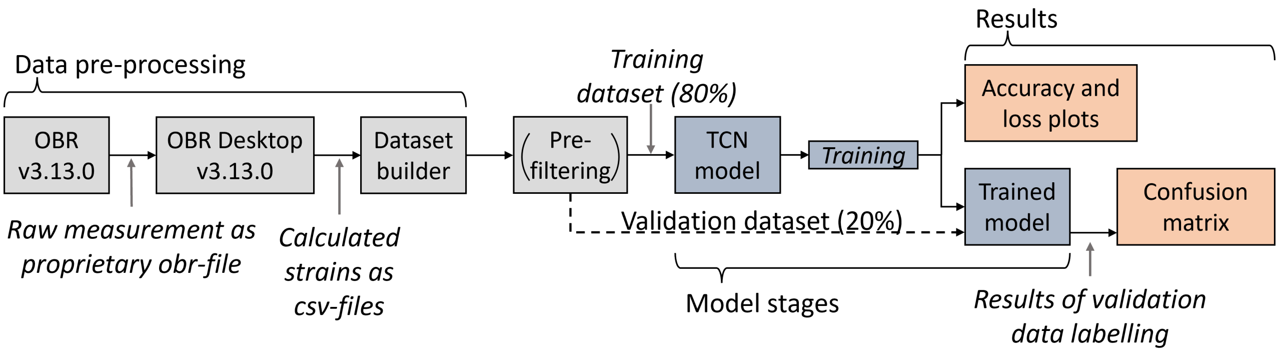


Figure 2.8.: Overview of data pipeline used for DOFS strain data analysis. Elements marked grey are related to data pre-processing, blue for elements related to TCN model development and red for post-processing. Reworked from [1].

$$softmax(z_i) = \frac{e^{z_i}}{\sum_j^n e^{z_j}} \forall i \in [1, n] \tag{2.6}$$

2.2.3. Data Pipeline

Figure 2.8 gives an overview of the whole data pipeline as designed and used in the work of this thesis. After data collection was completed, strains were extracted from the raw data as described in Section 2.1.3. The individual CSV files of strains were then loaded and normalised using absolute max scaling as shown in Equation 2.7, where ϵ_i are individual strain values, $\tilde{\epsilon_i}$ normalised strain values and $|\epsilon|_{max}$ is the maximum absolute value in the set of all strain values D for each measurement. The normalised strain values were then merged into two new datasets, one for each DOFS, each set containing all strain measurements for their respective DOFS. The two datasets were then saved as new CSV files, with one row per measurement. This archiving allowed for faster loading of datasets and corresponding labels during the training and validation process of the models. Simultaneously, separate label files were created containing the labels for each of the datasets, i.e. the distance and circumferential position of the acoustic source for each measurement, with one label set per row corresponding to the strain values in the same row of the dataset CSV file. For the investigation on the effect of input data pre-filtering on model performance, either the Gaussian or Band filter was applied as described in Section 2.1.5 during dataset loading.

$$\tilde{\varepsilon_i} = \frac{\varepsilon_i}{|\varepsilon|_{max}} \; \forall \; i \in S \tag{2.7}$$

As all Machine Learning methods leverage statistics in order to learn features in datasets, creating separate datasets for use in training and validation of a model is essential to ensure that the model predictions are statistically sound. A model will inevitably be biased towards correct predictions for data points used in training to improve the general prediction capabilities of the model. Using the same data points for validating the model would therefore not give insight into how good the generalisation of the model is, but rather just show how well the model has been trained to predict the labels for these exact data points. Instead, a set of new data points that the model has not seen during training, i.e. that have not been used to alter the model parameters during training and thus how it makes predictions, is needed to measure the generalisation ability of the model. For this reason, most supervised ANN model developments use a standard 80/20 split of data, with 80% being used for training the model, while the remaining 20% is reserved for validation of the model performance in between and after training.

Training and validation datasets were assigned by first loading the CSV files containing the relevant data points and corresponding labels as Pandas Dataframe-objects. To ensure balanced datasets, unique label sets based on distance and/or circumferential position were created, and data points were then sorted according to the set matching their respective labels. Using the sampling feature in Pandas, a given number of data points were then sampled from each label set and assigned to either the training or validation dataset, along with their corresponding labels. These were then shuffled before returning them as Numpy arrays for use in the training and validation of the models as described in the following sections.

In order to ensure repeatability of the results, and to reduce the chance of the sampling procedure resulting in a dataset that accidentally results in a high-performing model, identical static seeds are used for controlling sampling and shuffling of all datasets. The exception is for statistical analysis runs, where random seeds are used in order to ensure the obtained results are statistically sound and not just a one-off due to 'luck' when splitting the datasets.

2.2.4. Training

During training, loss functions are used for performance evaluation of the model, returning lower values (approaching zero) for increasing model performance. Training a model can therefore be viewed as a minimisation problem of its loss function, where trainable weight and bias parameters of the model are adjusted to approach this minimum. For classification models using supervised training, Categorical Cross Entropy (CCE) (Equation 2.8) is typically used, giving the sum of the logarithm of the model prediction y_i for all correct labels $\hat{y}_i$. This allows for the loss function to take the model 'confidence' into account, as the loss value will be lower the closer the predicted value is to 100% for the correct category, rewarding the model for increased confidence in correct predictions, and not only for making correct predictions. For regression models predicting one or more continuous values, Mean Square Error (Equation 2.9) is typically utilised, which encourages the model to reduce squared difference between the predicted and true values. In Equations 2.8 and 2.9, i refers to the entry, n to the total number of labels, y to the predicted label, and $\hat{y}$ to the true label of the respective entry.

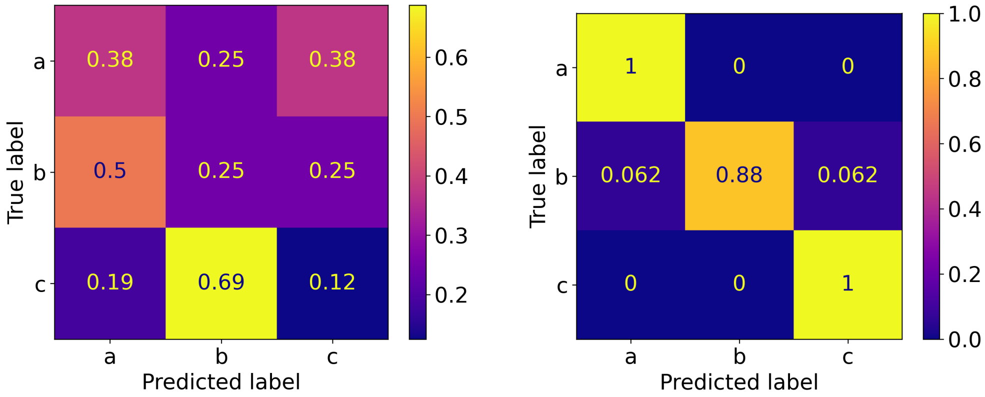
$$CCE = -\sum_{i}^{n} \hat{y}_i \cdot \log(y_i) \; \forall \; i \in [1, n] \tag{2.8}$$

$$MSE = \frac{\sum_i^n (y_i - \hat{y}_i)^2}{n} \; \forall \; i \in [1, n] \tag{2.9}$$

In order to train a model, the defined loss function such as is used to evaluate the current performance on a batch of training data. The size of this batch is a hyperparameter that is typically set so that an average of the model performance on the individual data points in the batch is used, avoiding the problem of overfitting to single data points. Adjustments are then made to model parameters, finding the direction of change for each trainable parameter that improves the average performance on the batch through a process known as *gradient descent*. The hyperparameter *learning rate* controls the 'step' the model takes in the direction found. The adjustment made is proportional to the learning rate, with a larger value allowing for larger 'steps'. While a large step size allows for initially fast improvements, it also increases the risk of overshooting the set of optimal parameters. At the same time, setting a small learning rate results in slow convergence and increased computational cost. To combat this, a common approach is therefore to have an initially larger step size and gradually reduce it as performance improves in order to locate the optimal set of trained parameters more accurately. Additionally, controlling the number of epochs, i.e. the number of iterations through the entire training dataset, is also necessary to find the optimal parameters without overfitting the model, a problem discussed previously in Section 2.2.1. For training the models in this thesis, exponential decay was used, which exponentially reduces the learning rate for each completed epoch, starting after a certain number of epochs where the initial learning rate was held constant. The decay of the learning rate r_l was calculated using Equation 2.10, where $r_{l,init}$ is the initial learning rate, r_d is the decay rate and e is the number of the current epoch.

$$r_l = r_{l,init} \cdot \exp\left(-r_d \cdot e\right) \tag{2.10}$$

For training the TCN models, hyperparameters and the model definitions were loaded in and passed on to the relevant model structure. The associated sampled training dataset was then passed to a custom training function for the model object, which defined and activated the appropriate performance measuring functions and learning rate scheduler for the given model type, as covered earlier in this section. The model was then trained for a given number of epochs. For each epoch, supervised training of the model was performed using 80% of the training dataset. The remaining 20% fraction of the training dataset was then used for validation of the model performance at the end of the epoch and the results were added to the model history before continuing to the next epoch. After the last epoch was completed, the trained model and its history were returned for further analysation and validation.



(a) Low-performing

(b) High-performing

Figure 2.9.: Confusion Matrices for two arbitrary categorical models, one low- and one high-performing, predicting data labels in classes *a*, *b* and *c*. Note the high-valued diagonal in the high-performing model.

2.2.5. Evaluation of Trained Models

In post-processing, the TCN models were validated with new sets of data that had not been used for training. For this, a standard 80/20 split of the dataset was used, thus reserving a random selection of 20% of the data points for validation for each of the unique labels in the dataset. From resulting model task predictions, accuracy for classification models was calculated as *Categorical Accuracy*, as given by Equation 2.11. As accuracy is a measure of overall performance, it does not provide insight into how the model performs in each category. Therefore, a *Confusion Matrix* (CM) was also generated from the model predictions. In a CM, each row corresponds to the true labels of the data, while the columns correspond to labels predicted by the model. As a result, correct predictions result in values on the diagonal of the matrix, while off-diagonal values represent incorrect predictions. From this, it can be revealed if there are specific labels the models struggle with categorising or if there is higher confusion between one or more subsets of labels. Figure 2.9 shows examples for two arbitrary models, where one is predicting labels randomly while the other is making correct predictions for most of the true labels.

$$\text{Categorical Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{2.11}$$

For regression models, there is no universal intuitive metric such as the accuracy is for categorical models, as the question is not *if* the prediction is correct or not, but rather *how close* the prediction is to the true value. As for a categorical model, single predictions are generally not interesting when validating a regression model. Validation aims at investigating whether a model has been able to properly generalise features and

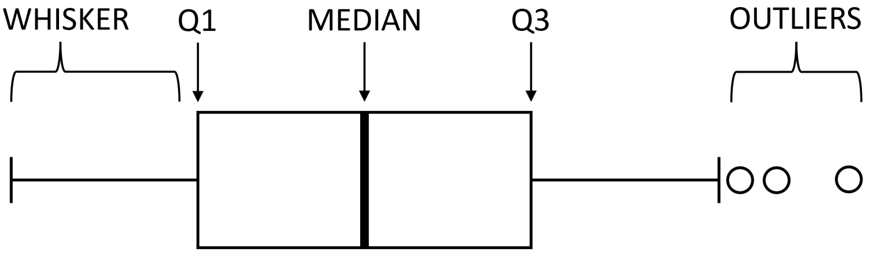


Figure 2.10.: Example boxplot with an explanation of elements of interest.

trends in a dataset. Unfortunately, there is no universal way to intuitively visualise regression model predictions for easy comparison of different models. One alternative is to use a scatter plot to show the values of individual output values. Points are then plotted, with their coordinates being the true and predicted value for each data point. As a result, the distance between the point and the 45° line between the two axes (true, predicted) represents the model performance for that point, with higher performance for shorter distances from the diagonal line. This is, however, a metric that is hard to quantify in an intuitive way and therefore more suited as a method to qualitatively compare the relative performance of several models. Due to this, a *box plot* was instead created from the results of each regression model. This gave both a known and intuitive visualisation of the distribution of model predictions, as well as quantitive values of this distribution for comparison of several regression models. For each model, since both the distance and angle parameters were one of three discrete values, it was decided to use one set of box plots per parameter. Each set consists of three box plots to show the distribution of the predicted values for their respective true value.

Figure 2.10 shows an example boxplot, marking different elements of interest. A box plot, also known as a box-and-whisker plot, is a visualisation method that displays the first and third quartile (Q_1 and Q_3) of a set of data as the sides of a box, with a centre line marking the median value. Further, whiskers in the form of thin lines extend out from each side of the box, typically marking the minimum and maximum values in the data. In the plots used in this thesis, the whiskers are extended to a length of 1.5 times the Interquartile Range (IQR), which is defined as $IQR = Q_3 - Q_1$ and represents the middle 50% of the data. This allows for the visualisation of statistical dispersion of the data, i.e. how spread out the data points are, instead of only showing the total range of predicted values. Values outside the range of the whiskers are considered outliers and therefore marked using small circular markers, as they are significantly higher or lower than the rest of the predictions. Box plots with either whiskers of different lengths or a median line that is not centred in the box, show that the data is skewed in the direction of shorter lengths. The size of the box and the length of the whiskers increases with increasing dispersion and variability in the data.

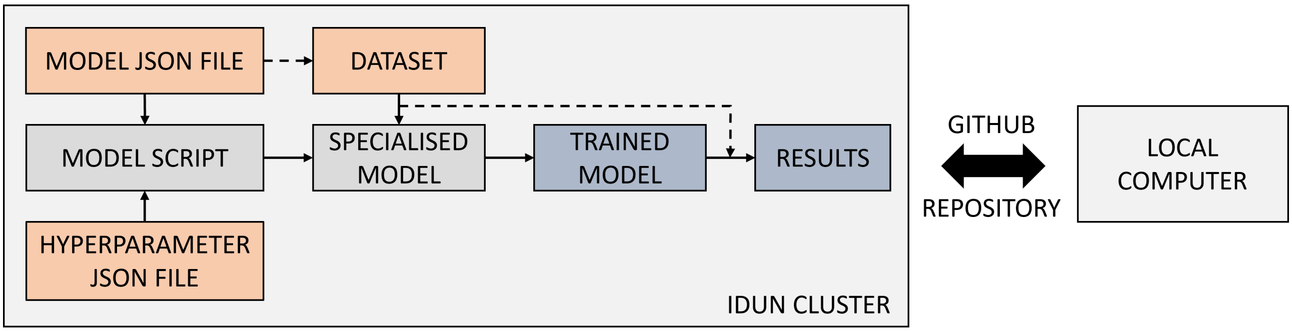


Figure 2.11.: Overview of workflow for the IDUN cluster [52] used for the development of TCN models.

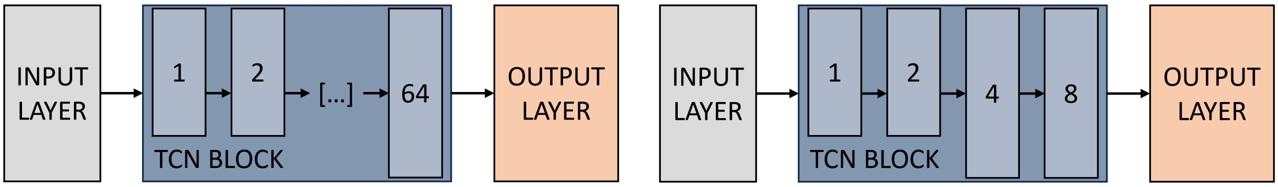
2.2.6. Infrastructure and Workflow

The cluster computer Idun [52] at NTNU (Norwegian University of Science and Technology) was used for computationally heavy tasks of training and validating the TCN models. Early testing revealed that while training and validating the models were significantly faster on the cluster compared to when running the code locally. A significant amount of time was still spent on loading and compiling relevant libraries. A workflow was therefore developed to optimize the process and even better utilise the cluster, as illustrated in Figure 2.11. Python libraries and packages used for the development of TCN models are listed in Table 2.6.

All code was developed locally and synced to a directory on the cluster computer using a private GitHub repository [53]. Hyperparameter sets for the models were defined in separate JSON files. Top-level input parameters for each model such as the dataset to be used, number of samples, version number, number of outputs etc. were defined in another JSON file serving as the 'model job' file. This allowed for creating a general Python script which then loaded one or more input parameter sets consisting of hyperparameters and a model JSON file to configure the exact architecture and behaviour of a specific model, along with the dataset to be used. After training and validation, plots were generated and synced back to the local computer using the same GitHub repository.

2.2.7. Final ANN Model Structures used for Analysis

As both categorical and regression models were used, two TCN model architectures were developed, as shown in Figure 2.12. These are relatively similar, with the main differences in the output layer. As discussed in Section 2.2.1, the number of output nodes is determined by the number of categories or the number of continuous values to predict. Additionally, the activation functions used in the output layer depend on the model type, with the softmax function (Equation 2.6) being used for categorical models, and the identity linear activation function (Equation 2.2) for regression models. For



(a) Categorical model **(b)** Regression model

Figure 2.12.: Overall architecture of ANN models.

training, Categorical Cross-entropy (Equation 2.8) was used for evaluating categorical models, while regression models were evaluated using the Mean Square Error function (Equation 2.9). The hyperparameter sets used are previously given in Table 2.4, while an overview of the final models developed is given in Table 2.5.

2.2.8. Post-processing

In post-processing, the history output from the trained model is used for plotting the evolution of loss and accuracy of the model during training. The trained models are also run with the set of validation data, and predicted labels generated by the model are subsequently compared to true labels of the validation dataset. This enables the generation of confusion matrices for further analysis.

Table 2.5.: Overview of final models used for producing the results presented in this thesis, each assigned a unique identifier, with their respective model type, the dataset used for training and their output parameters listed.

Model ID	Type	Dataset	Predicting
CH-F	Categorical	Helical	Distance, Sector
CH-D	Categorical	Helical	Distance
CH-S	Categorical	Helical	Sector
CS-F	Categorical	Straight	Distance, Sector
CS-D	Categorical	Straight	Distance
CS-S	Categorical	Straight	Sector
RH-F	Regression	Helical	Distance, Sector
RH-D	Regression	Helical	Distance
RH-S	Regression	Helical	Sector
RS-F	Regression	Straight	Distance, Sector
RS-D	Regression	Straight	Distance
RS-S	Regression	Straight	Sector

Table 2.6.: Python packages and their use in the work of this thesis

Package	**Version**	**Use**
keras	2.6.0	High-level neural network API for deep learning tasks.
keras-tcn[51]	3.5.0	Provides the TCN (Temporal Convolutional Network) layer for sequence modelling tasks using Keras.
matplotlib	3.7.0	Plotting library for creating visualizations.
numpy	1.20.3	Fundamental package for numerical computing.
pandas	1.2.4	Data manipulation and analysis library, particularly useful for working with structured data.
sklearn	0.0	Library for generation and plotting of confusion matrices.
scipy	1.6.3	Library for scientific and technical computing. Used for filtering of data.
tensorflow	2.6.0	Open-source machine learning framework for building and training various models, here serving as the back-end for keras.

2.3. Experimental Setup

The setup used for data collection consisted of two jacketed Distributed Optical Fibre Sensors (DOFS) which were connected to the OBR 4600 interrogator from Luna Technologies and the free ends submerged vertically under water. Measurements of the DOFS were recorded using an automated procedure as described in Section 2.1.2 while a submerged buzzer device was activated at different circumferential positions and distances near the DOFS. For this experiment, a linear, vertical DOFS and a helically-wrapped vertical DOFS were used. The linear DOFS was intended for establishing distance to the acoustic source, while the helically-wrapped DOFS was expected to work as a full azimuth array, thus enabling the TCN models to perform Sound Event Localisation in 3D better than with the linear DOFS version.

2.3.1. Floating Structural Frame

For the design of the experimental setup, several requirements were set. The setup had to provide (1) several possible positions for the acoustic source, where (2) both distance and circumferential position relative to the DOFS were adjustable. These two parameters (3) had to be individually adjustable. As the experiment was to take place under water, (4) the support structure had to float, and be (5) modular for easy transportation, assembling and disassembling between runs. It was further expected that moving the DOFS could introduce changes in reflections and background noise levels, complicating feature extraction for the model. Consequently, it was decided that (6) the DOFS would be kept stationary, while only moving the acoustic source.

To fulfil these requirements, it was decided to create a structural frame, shown in Figure 2.13, consisting of three concentric half-circles made of floating foam tubes with radii of 1, 2 and 3 meters. The DOFS were positioned at the centre, with the acoustic source suspended at different positions on the half-circles. Hard plastic tubes were used to keep the circular shape radially consistent. The plastic tubes were attached to the central hub, spanning outwards at 0°, 60°, 120° and 180° angles. A 3D-printed centre hub was used to connect the structural tubes at the centre, while the foam tubes were attached to the structural tubes using strips. The resulting structure provided three different distance options between the DOFS and the acoustic source. At each distance, three circumferential positions at 30°, 90° and 150° were designed, for a total of 9 unique positions for the acoustic source.

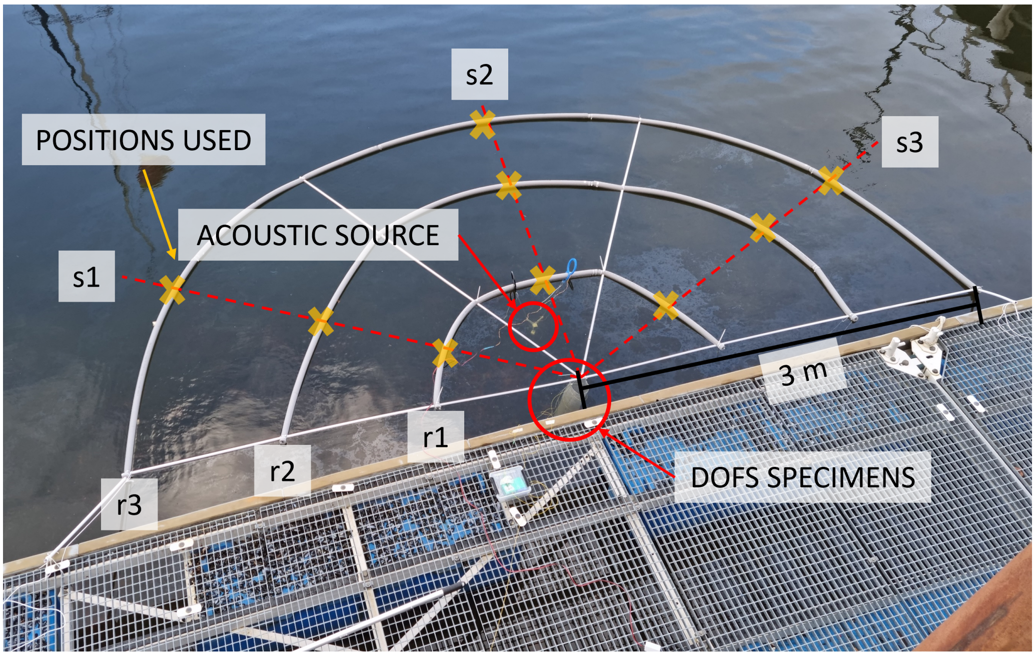


Figure 2.13.: Experimental setup in water, where distance (r1, r2, r3) and sector (s1, s2, s3) for individual positions for the acoustic source are marked.

2.3.2. DOFS Fixture

To ensure a controlled placement of two DOFS, a thin-walled tubular fixture was created using a fine metal mesh, shown in Figure 2.14. The mesh was chosen as it allowed for flexible positioning of the DOFS while sinking in water and allowing acoustic waves to pass through. The mesh was further reinforced at the ends using a metal circle made of a thicker wire, with a secondary wire spanning across both circles. The tube was suspended at the centre of the floating structural frame using a thin polyester silk line, with a heavy metal bolt hanging from the bottom of the mesh tube to keep it vertical in the water. One linear and one helically-wrapped DOFS were attached to the tube as described below. Both DOFS extended 300 *mm* beyond the end of the tubular fixture to ensure that the sensing range was well out of the range of end reflection noise as found by Usenco [48]. To avoid water intrusion affecting the optical signals, the optical fibres were extended ~ 2 *m* beyond the fixture before the connectors were spliced in, ensuring they could be kept on dry land.

The initial mesh fixture was 2000 *mm* long, but it was shortened due to a change of testing location leading to shallower water than first planned. Table 2.7 describes the geometry of the fixture and the DOFS after this change. Figure 2.14 shows the final mesh fixture with both DOFS attached.

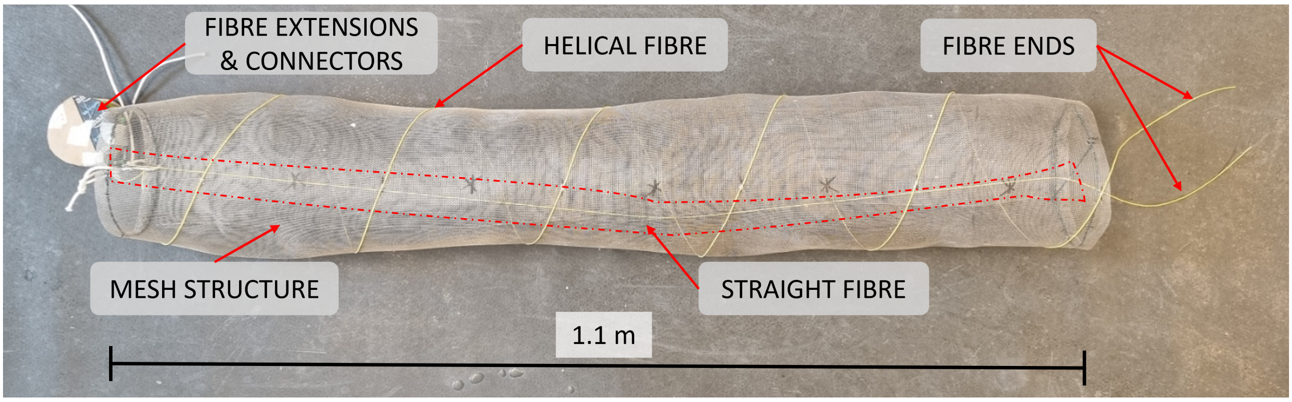


Figure 2.14.: Tubular mesh fixture with two jacketed DOFS attached.

Table 2.7.: Geometrical specifications of jacketed DOFS used in the experiment. The sensing midpoint is the distance from the interrogator to the middle of the mesh tube along the optical fibre.

Parameter	Linear DOFS	Helical DOFS
Sensing range	1000 *mm*	1500 *mm*
Sensing midpoint	39157 *mm*	51508 *mm*
Tube length	n/a	1100 *mm*
Tube diameter	n/a	145 *mm*
Pitch	n/a	200 *mm*

Linear DOFS

A straight, linear DOFS with parameters as defined in Table 2.7 was prepared with an FC connector spliced in at one end. It was positioned in the centre of the tubular mesh fixture and attached to the straight metal wire in the centre at both ends of the mesh using drops of epoxy glue. During attachment, it was ensured that there was enough slack in the DOFS to avoid pre-tensioning, but not so much that it could come into contact with the mesh sidewalls.

Helical DOFS

A helical DOFS with parameters as defined in Table 2.7 was prepared with an FC connector spliced in at one end. Before positioning it on the tubular mesh fixture, markings were made along the opposite sides of the fixture with a spacing of 200 *mm* according to the pitch defined. The DOFS was then manually wrapped around the fixture to achieve a helical shape, and attached to the mesh by using drops of epoxy glue at the previously made markings.

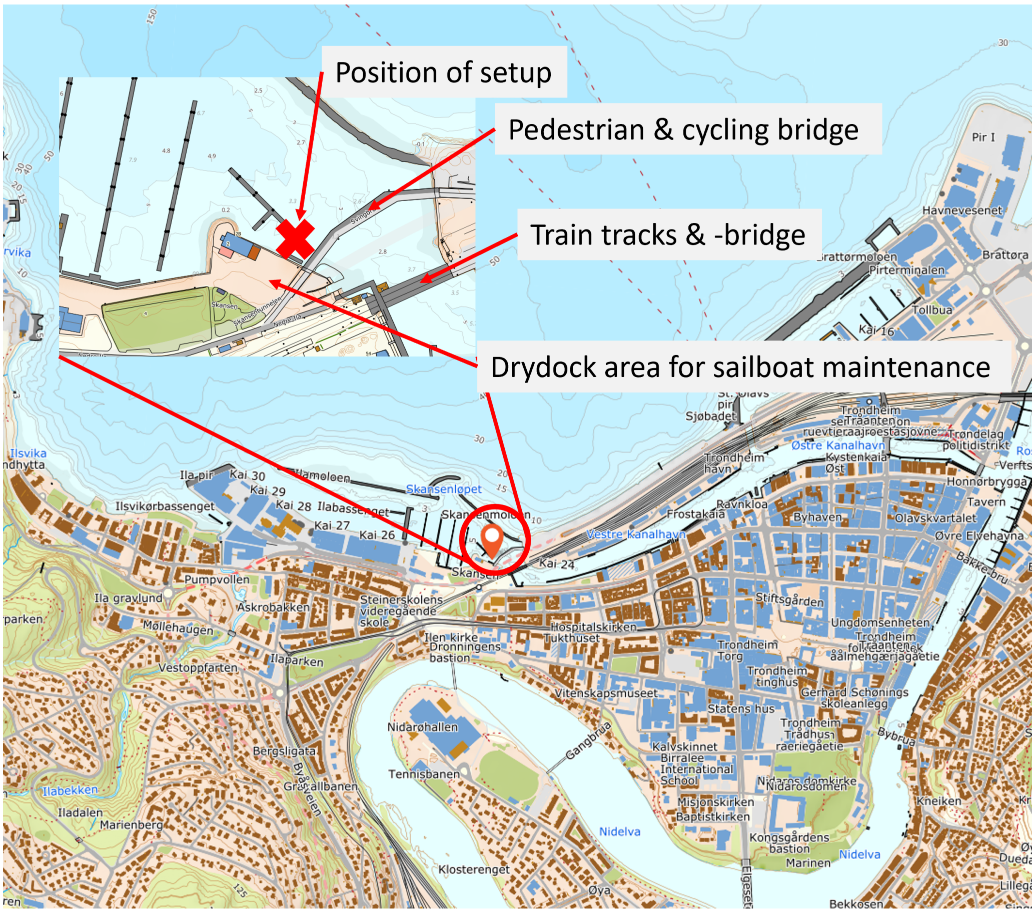

Position of setup
Pedestrian & cycling bridge
Train tracks & -bridge
Drydock area for sailboat maintenance

Figure 2.15.: Map showing location used for data collection. Adapted screenshot from Norgeskart.no [54]

2.3.3. Acoustic Source

A CI26P-16T40-C-1X piezoelectric buzzer from Challenge Electronics with a central frequency of 400 Hz was used as the acoustic source for the experiment, sending out a continuous acoustic tone while activated. A pair of electrical wires were soldered to the buzzer before it was placed in a 3D-printed casing and waterproofed using epoxy to cover any seams and other openings.

As the buzzer is directional, it was suspended off the floating frame using two 3D-printed hooks with some space in between. A weight was attached to the bottom of the casing to keep the buzzer positioning vertical in the water. This setup ensured that the buzzer was always pointed towards the centre of the DOFS mesh fixture in the centre of the floating frame. While the measurements were being performed, the buzzer was powered by a power supply at 14 V, controlled using a Buck converter connected to the wire pair between the supply and buzzer.

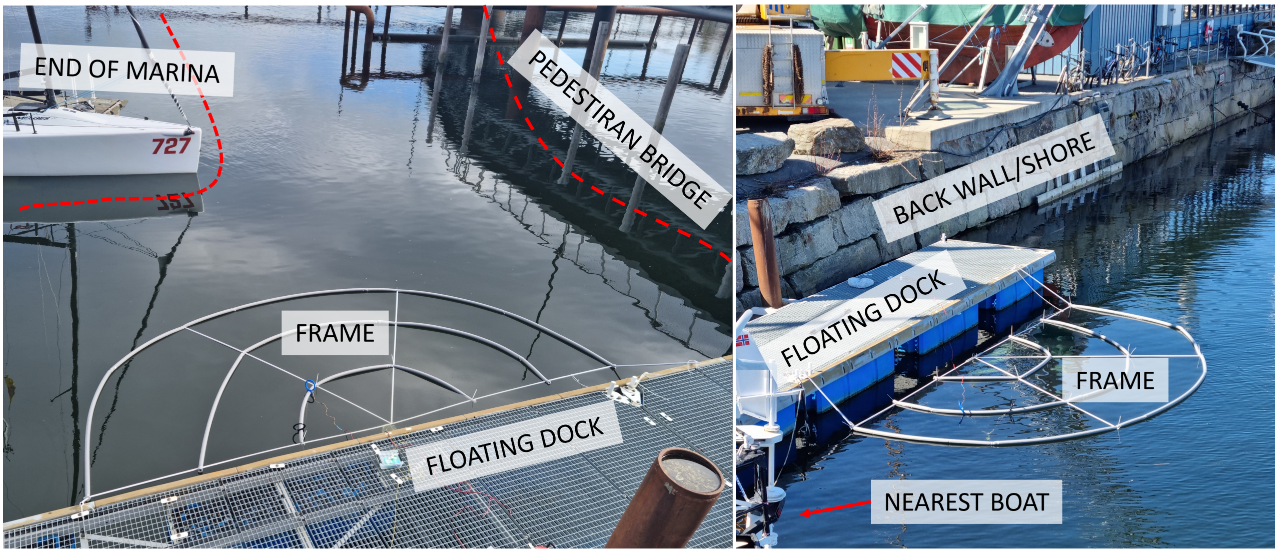


Figure 2.16.: Overview of the location used for data collection. The left picture is taken on top of the shore, looking outwards, and the right picture is taken from the pedestrian bridge with a wider perspective.

2.3.4. Data Collection Process

The location used for data collection, along with electricity for running the experiment, was kindly provided from *Trondhjems Seilforening* [55] for two days, 03/05-23 and 05/05-23. Figure 2.15 shows a map marking the location used. A floating dock was used as the working platform and anchoring point for fixing the floating frame, located at the end of a marina. The floating dock enabled a distance between the DOFS and the shore of $\sim 3\ m$, with the latter built up of large natural rocks and slightly slanted backwards. The surrounding area consisted of a breakwater opposite the setup, similar to the back wall and in practice creating a wide canal. Figure 2.16 shows the frame with surrounding elements marked. While the acoustic source ended up being pointed at the wall on the shore, it was assumed that the uneven shape of rock faces, along with the natural sea bottom, dispersed the sound waves enough that any reflection of acoustic waves would have a negligible effect on the DOFS compared to the effect of direct acoustic waves from the buzzer.

For each day, the frame was assembled, placed on the water and attached to the floating dock using polyester silk lines and two pieces of hard plastic tubes to keep the floating frame stationary. The tubular mesh fixture with the DOFS was suspended vertically at the centre of the frame, while the acoustic source was suspended at the different positions marked in Figure 2.13 for each of the measurement sets taken. The setup was then dismantled at the end of each day, with the total time used for measurements including setup, testing and dismantling being around 11 hours per day.

The first half of the first day was mainly used for initial testing of the setup, and for shortening the DOFS mesh fixture as described in Section 2.3.2 due to a lower tide than originally expected from the tidal forecast. Sets of 50 measurement samples using the linear DOFS were then taken from all positions except for *r3s2* with the following method. For each position, (1) the buzzer was moved into position and deactivated before (2) a reference measurement was taken, consequently only measuring the effect of the background noise. (3) The buzzer was then activated and confirmed working before (4) continuously measuring 50 samples from the linear DOFS. (5) The buzzer was again checked to ensure it was still activated at the end of the sample set before moving to the next position. Due to limited time, measurements from position *r3s2* were not taken on the first day.

On the second day, the measurement procedure from the first day was repeated, now taking measurements using both the linear and helical DOFS. Reference measurements were taken of both DOFS at each position before taking 50 and 100 measurement samples for the linear and helical DOFS respectively. The exception was again for position *r3s2*, for which 100 samples were taken for both DOFS to make up for the lack of measurements from the first day for this position. The resulting total dataset consisted of 100 repeated measurement samples for each DOFS at each of the nine buzzer positions.

As the location was in a public area, there were other regular and sporadic activities nearby that likely also added to the acoustic background affecting the measurements. Some of these, such as activities around the sailboats being maintained on shore, pedestrian and cycling traffic on the bridge nearby, as well as trains passing on the bridge a bit further away, were assumed to have a sufficiently smalltransient effect on the measurements that the models themselves could filter it out during analysation. Bigger disturbance events suspected to have a non-negligible impact on measurements were logged and are listed in Table 2.8, along with comments on their suspected effect and any measures taken to mitigate this.

Table 2.8.: Log of disturbance events during acoustic experiments, with time stamp and comments addressing the suspected effect on measurements, and eventual measures taken to mitigate these.

Time	**Event**	**Comment**
3/5 11:50	Noisy work on a boat at the same dock	Done between measurement sets, so there is likely no measurable effect.
3/5 13:51	Jetski passing nearby	Current measurement set was restarted.
3/5 14:20	Sensor fixture weight hit bottom	Tubular fixture itself did not touch the bottom, but might have moved a bit due to the weight shifting. No measures were taken, as a new reference is taken for each set.
5/5 12:15	Use of angle grinder on the boat in drydock next to setup	Assumed to have a negligible effect.
5/5 13:15	Pause in measurements due to work with a crane on the dock	No measurements done, i.e. no consequences.
5/5 14:00	Irregular, strong gusts of wind	Continued through the rest of the day. Assumed to have a negligible effect as the setup did not move noticeably.
5/5 18:29	Increased strength of currents in water	Could affect the position of the buzzer slightly, but are assumed to have negligible effect.

Chapter 3.

Analysis of Experimental Results

The model development and data analysis results from the acoustic experiments performed are presented in the following sections. For each part, selected results are presented, along with objective analyses and comments on what can be directly observed from the Figures and Tables presented. The remaining results not presented in the current chapter can be found in Appendix A and B.

3.1. Model Performances and Stability Study

Table 3.1 gives an overview of performances for all TCN models developed as specified in Table 2.5, with their respective final validation accuracies (Equation 2.11) and loss values (Equation 2.8-2.9) after all training epochs had been completed. Additionally, the difference between the validation and training loss is listed for each model, giving an indication of how well the model is generalised with respect to under-/overfitting. The models were trained using their respective optimal hyperparameter sets on full 80% training datasets as described in Section 2.2.7, with 30 epochs for all categorical models and 50 epochs for all regression models to reach the final values listed.

Table 3.1.: Overview of all TCN model performances after completing all training epochs. The Loss difference column gives the absolute difference between the final training and validation losses for each model

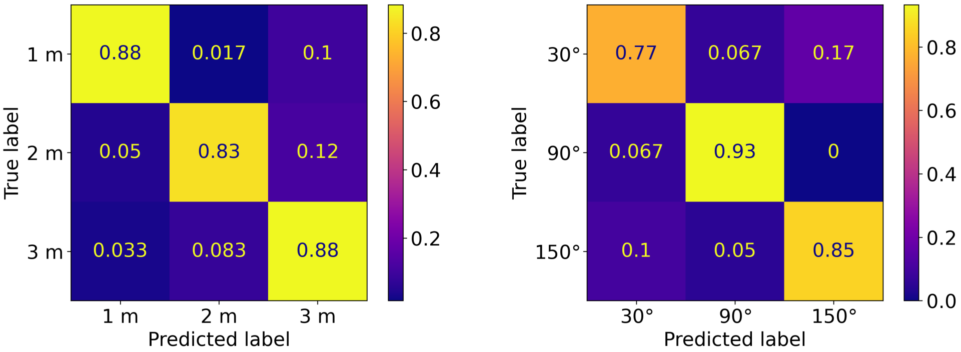
Model	Validation accuracy	Validation loss	Loss difference
CH-F	0.653	0.999	0.116
CH-D	0.750	0.629	0.042
CH-S	0.813	0.604	0.066
CS-F	0.764	0.882	0.022
CS-D	0.875	0.371	0.073
CS-S	0.778	0.632	0.083
RH-F	-	0.471	0.069
RH-D	-	0.494	0.010
RH-S	-	0.438	0.125
RS-F	-	0.479	0.063
RS-D	-	0.452	0.073
RS-S	-	0.485	0.021

3.1.1. Categorical Models

The results presented in Table 3.1 and Figure 3.2 show that all categorical models generally perform similarly, with final accuracy scores reaching 70% to 80%. From Figure 3.2 it can also be observed that the development of loss and accuracy scores throughout the training epochs follows a similar pattern, albeit with varying start and end values. An outlier here is the CH-F model, classifying both distance and sector using the helical fibre data, which has the lowest validation accuracy of 65%, along with the highest values for both validation loss and loss difference. This is reflected in the loss curves in Figure 3.3b, showing that the validation loss curve stabilises around 1.0, while the training curve flattens out at roughly 1.2.

Further, Model CS-D, predicting only distance categories using the straight fibre data, has a significantly higher accuracy score compared to other categorical models. The loss value is also the lowest out of all the classification models, although the loss difference is in the higher range. Studying the detailed plot in Figure 3.3a reveals that higher loss difference is a result of the training stopping at a seemingly local spike in training loss, while the accuracy curves are following each other relatively closely between 0.8 and 0.9 for the converging part of the training region roughly from epoch 10.

Comparing the confusion matrices of the straight DOFS single-parameter models CS-D (distance) and CS-S (sector) in Figure 3.1, a low amount of confusion can be observed for both models due to low off-diagonal values. For CS-S, which only predicts the sector for the source position, higher off-diagonal values can be observed compared to that of CS-D. This is also reflected by the lower accuracy and higher loss values of CS-S in



(a) CS-D (Distance)

(b) CS-S (Sector)

Figure 3.1.: Confusion matrices for single-parameter classification models using straight DOFS data. Rows are true labels, while columns are the same labels as predicted by the TCN models. Values are normalised for each row.

Table 3.1.

For the confusion matrices of full categorical models CH-F and CS-F in Figure 3.4, the off-diagonal values of CH-F in 3.4a are relatively evenly distributed, indicating no specific pattern to the confusion of this model. For CS-F in 3.4b, the off-diagonal values are more concentrated around the categories at distance *r3*, indicating a higher confusion for the model when the acoustic source is furthest away from the DOFS. Overall, for individual acoustic source parameters (distance and sector) separately as well as in the full combined model, straight DOFS models perform well, predicting the source location on par or better compared to helical DOFS models.

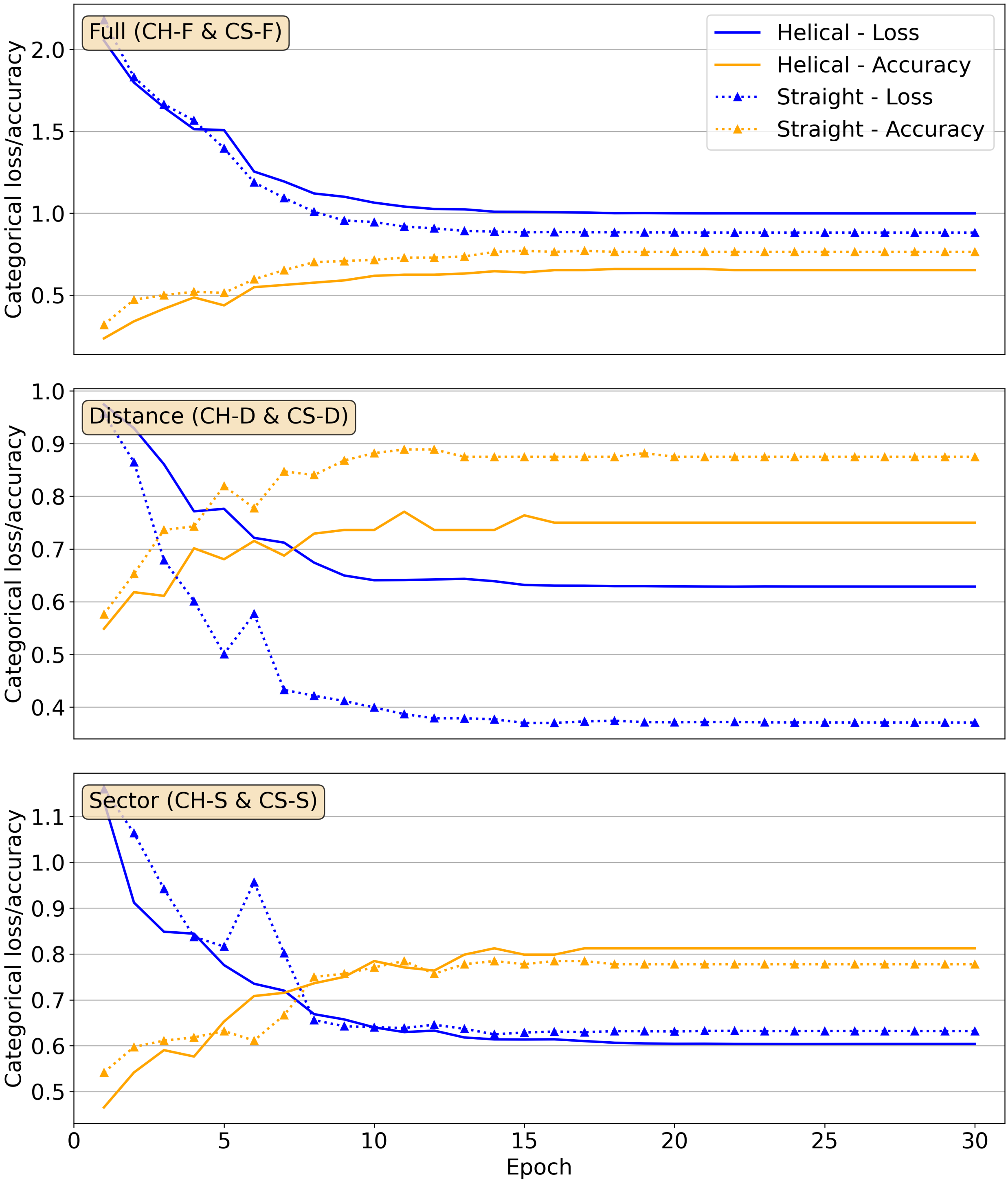
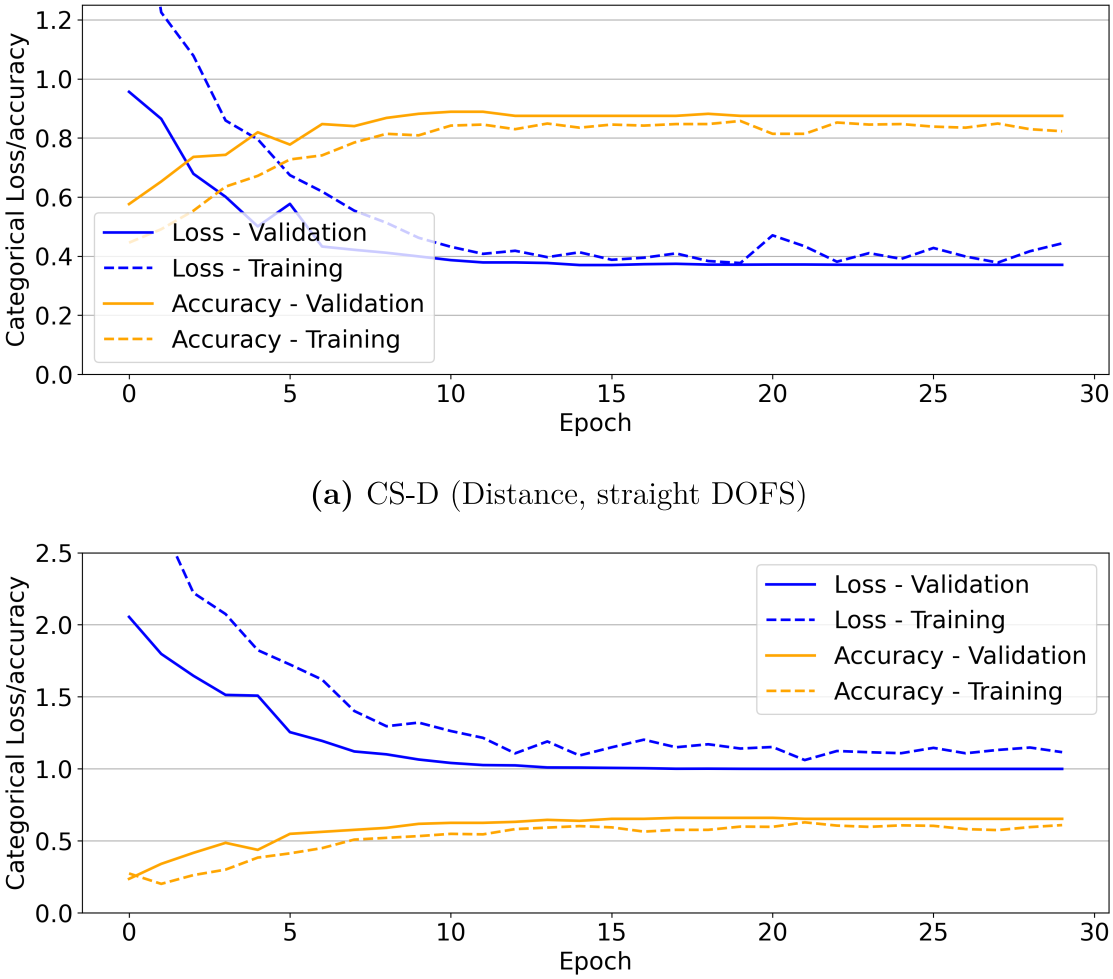


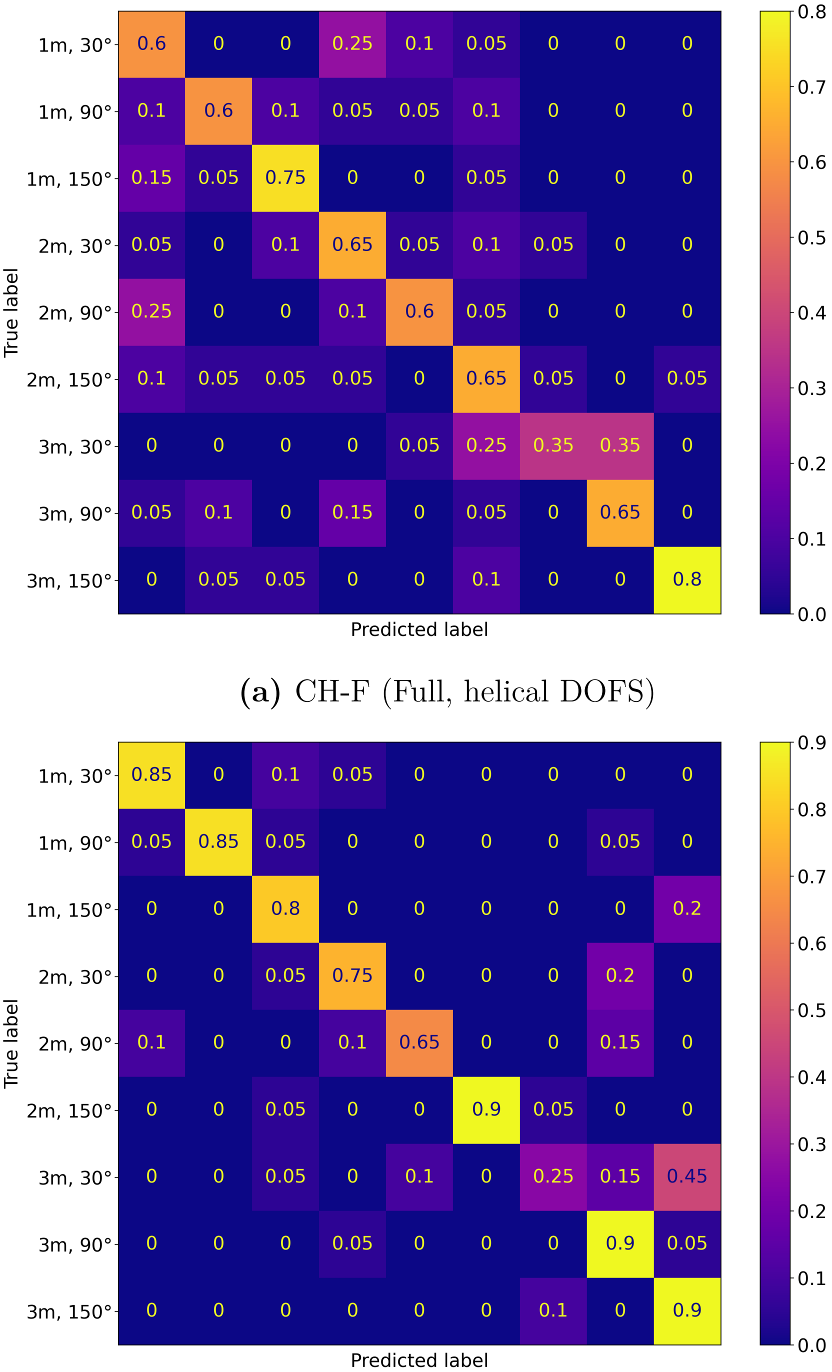
Figure 3.2.: Development of loss and accuracy values for categorical models predicting the category of either one or both of the parameters distance and sector for the relative position of the acoustic source.



(a) CS-D (Distance, straight DOFS)

(b) CH-F (Full, helical DOFS)

Figure 3.3.: Development of training and validation performance for loss and accuracy of selected categorical models



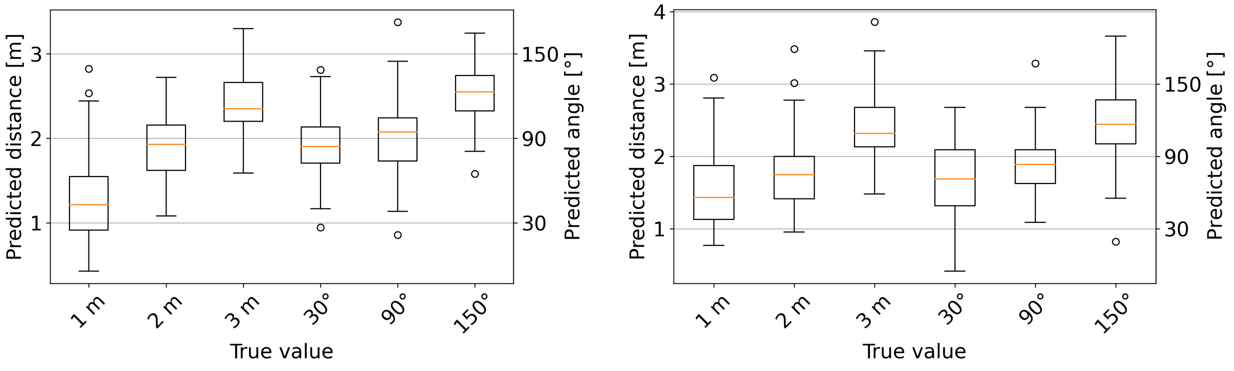
(a) CH-F (Full, helical DOFS)

(b) CS-F (Full, straight DOFS)

Figure 3.4.: Confusion matrices for classification models predicting both distance and sector of the acoustic source. Rows are true labels in the form of a unique combination of distance and sector, while columns are labels as predicted by the TCN model. The column labels were removed in order to fit the figures on one page, but the columns (left-right) are equal and in the same order as the row labels (top-down). The values are normalised for each row.

3.1.2. Regression Models

Table 3.1 and the accompanying validation loss development graphs in Figure 3.6 show that all 6 regression models have similar performance, with the loss curves generally converging towards values in the range of 0.45 to 0.50. A more unstable behaviour in early epochs can be observed for models predicting the direction of the acoustic source position, namely RH-F, RS-F, RH-S and RS-S, but the performance of these also stabilises in similar ranges to the two distance models towards higher epoch values. Further, considering the boxplots for full (both distance and direction) models RH-F and RS-F in Figure 3.5 show that while there is a visible difference between the median values of prediction of different labels, the predictions overall span most of the range of true values. Similar behaviour was found to be true also for the models only predicting one parameter (distance or direction separately), as shown with the boxplots for the remaining regression models in Figure 3.7.



(a) RH-F (Full, helical DOFS) **(b)** RS-F (Full, straight DOFS)

Figure 3.5.: Boxplots for regression models predicting both distance and sector. The box covers values between the first and third quartile, with the whiskers marking the range equal to 1.5 times the range of the box. Dots mark any predictions outside this range.

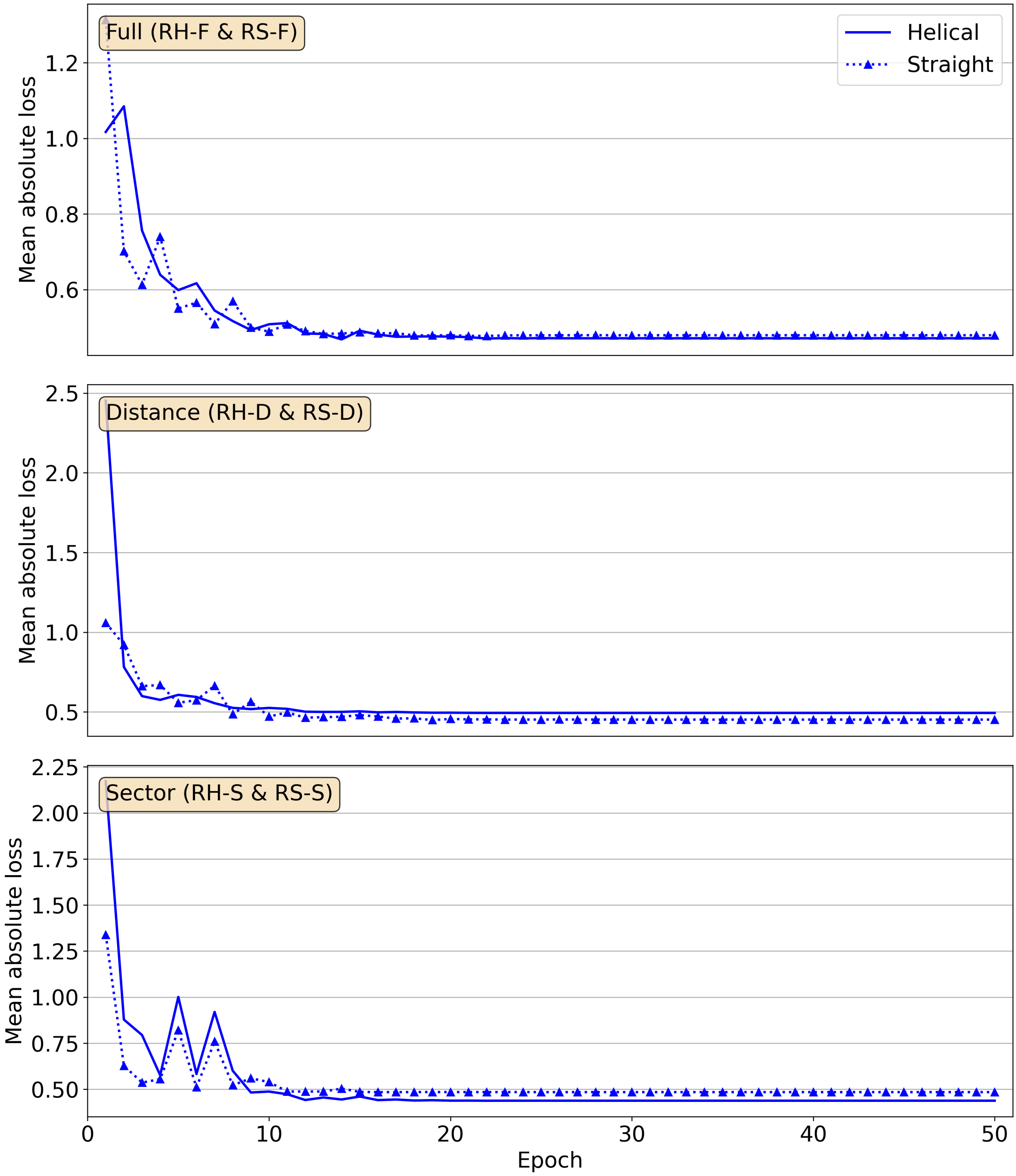
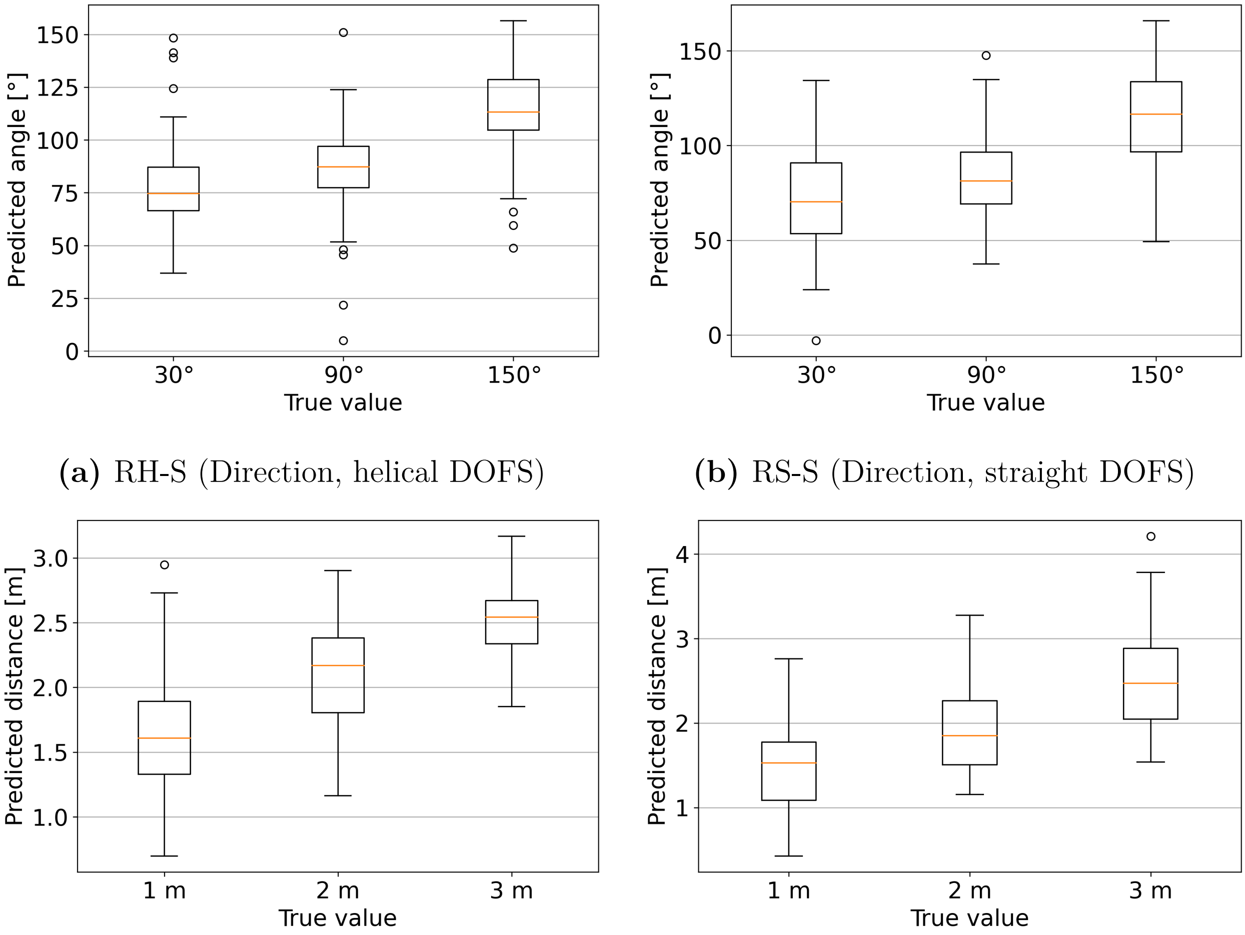


Figure 3.6.: Development of validation loss for regression models predicting either one or both parameters distance and direction as scalar values for the varied position of the acoustic source.



(a) RH-S (Direction, helical DOFS)

(b) RS-S (Direction, straight DOFS)

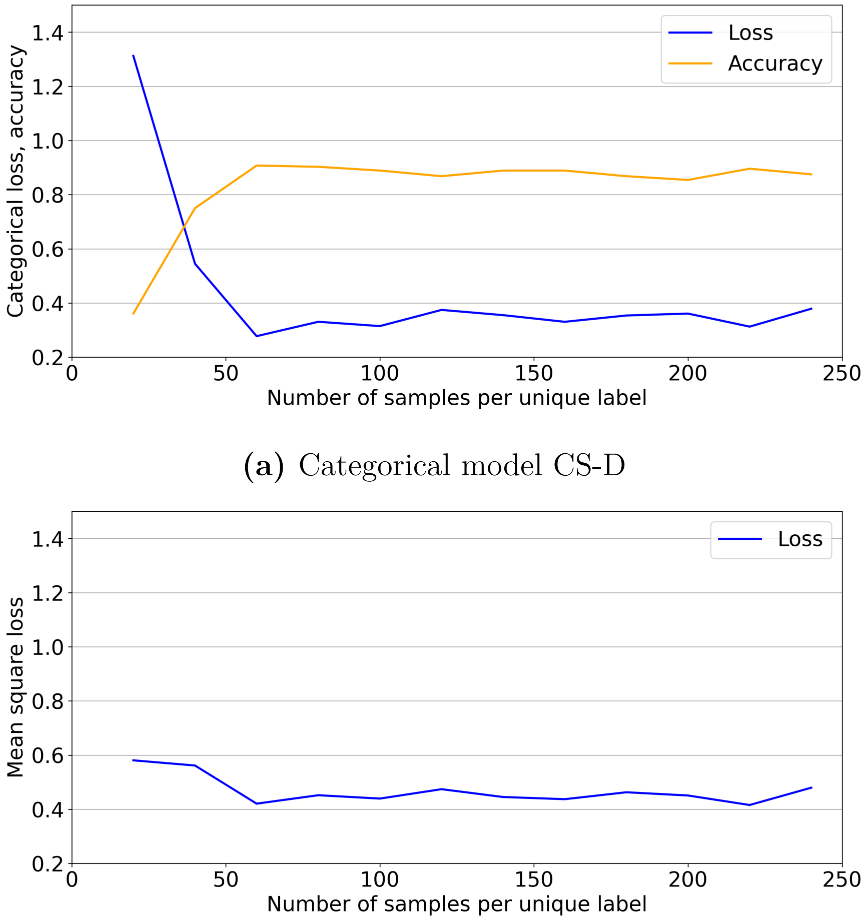
(c) RH-D (Distance, helical DOFS)

(d) RS-D (Distance, straight DOFS)

Figure 3.7.: Boxplots for regression models predicting either distance or direction to the acoustic source. The box covers values between the first and third quartile, with the whiskers marking the range equal to 1.5 times the range of the box. Dots mark any predictions outside this range.

3.1.3. Effect of Sample Size on Model Performance

In Figure 3.8, model performance for categorical model CS-D (Figure 3.8a) and regression model RS-D (Figure 3.8b) for increasing sample sizes of training data are plotted. For both models, an apparent performance plateau is reached at ca. 60 samples per unique label, where both the accuracy and the loss curves flatten out with no visible performance improvements for increasing the sample size beyond this point.



(a) Categorical model CS-D

(b) Regression model RS-D

Figure 3.8.: Model validation performances after training with varying sample size in the training data set per unique label.

3.1.4. Statistical Analysis

Figure 3.9 shows the resulting mean value and ±1 standard deviation range for validation loss and accuracy from 30 stochastically varied runs of the highest-performing categorical model, CS-D. It can be observed that the standard deviation remains relatively constant throughout the training process, with a slight reduction as the mean curves flatten out

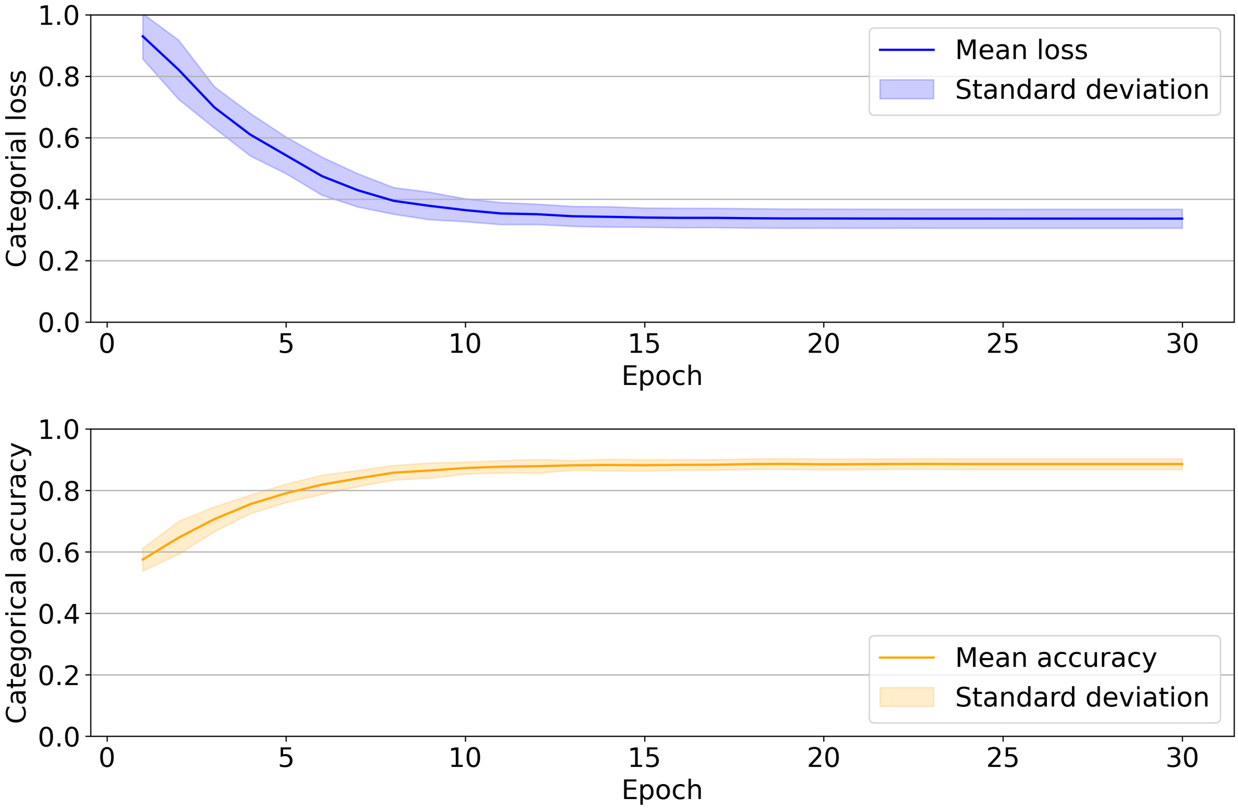


Figure 3.9.: Plots showing the mean and standard deviation values of validation loss and accuracy for each epoch, calculated using values from 30 runs of the categorical model CS-D in Table 2.5, using non-deterministic values for initialising weight and biases, as well as for shuffling the datasets.

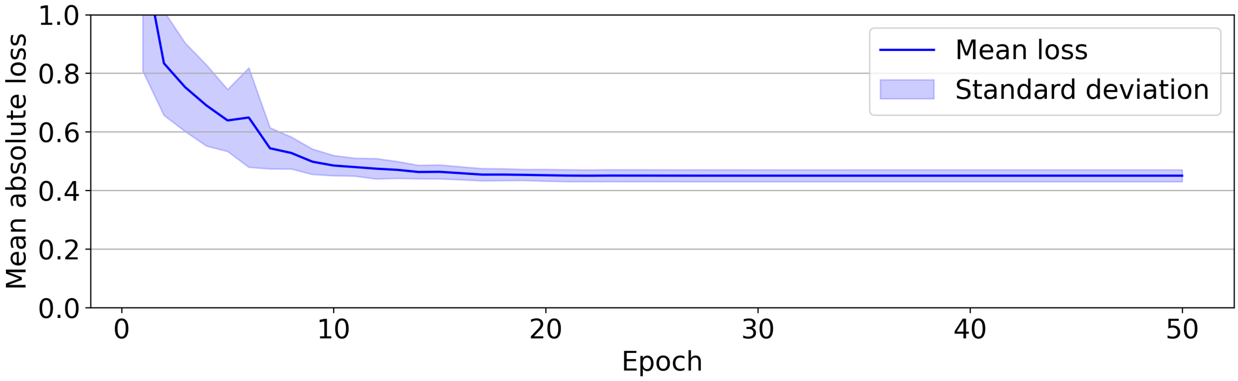


Figure 3.10.: Plot showing the mean and standard deviation values of validation loss for each epoch, calculated using values from 30 runs of the regression model RS-D in Table 2.5, using non-deterministic values for initialising weight and biases, as well as for shuffling the datasets. Note that the graph is cut off at 1.0 to better show the values for the later training stages.

towards 0.35 for loss and 0.9 for accuracy. A similar trend can be observed for the plot of statistical analysis of 50 stochastically varied runs of regression model RS-D in Figure 3.10. While the standard deviation during initial training epochs is larger than for the categorical model, a significant reduction can be observed as the mean loss curve flattens towards 0.45. This indicates that the stability of the regression model is comparable to that of the categorical model.

3.2. Effects of Data Pre-filtering

Figure 3.11 and Figure 3.12 show the model validation performance scores for the high-performing CS-D model trained with pre-filtered DOFS strain data points. Figure 3.11 shows the result of applying a Gaussian filter with increasing sigma values on the strain data. For sigma values up to 1.0, a performance comparable to or even slightly exceeding that of the model trained using unfiltered data can be observed, followed by a decrease in performance for increasing sigma values. A similar performance drop can be observed in the plots in Figure 3.12, where different band filters were applied instead of the Gaussian filter. The band filters had a spatial frequency range width of 1.0 $\frac{1}{m}$, but with increasing centre value in the total frequency range from 0 to 4.9 $\frac{1}{m}$ of the straight DOFS data set. All band filters resulted in an observable performance decrease compared to the unfiltered CS-D model. The reduction in model performance when using strain data filtering indicates that DOFS strain data contains vital information for the investigated acoustic events in a wide range of spatial frequencies.

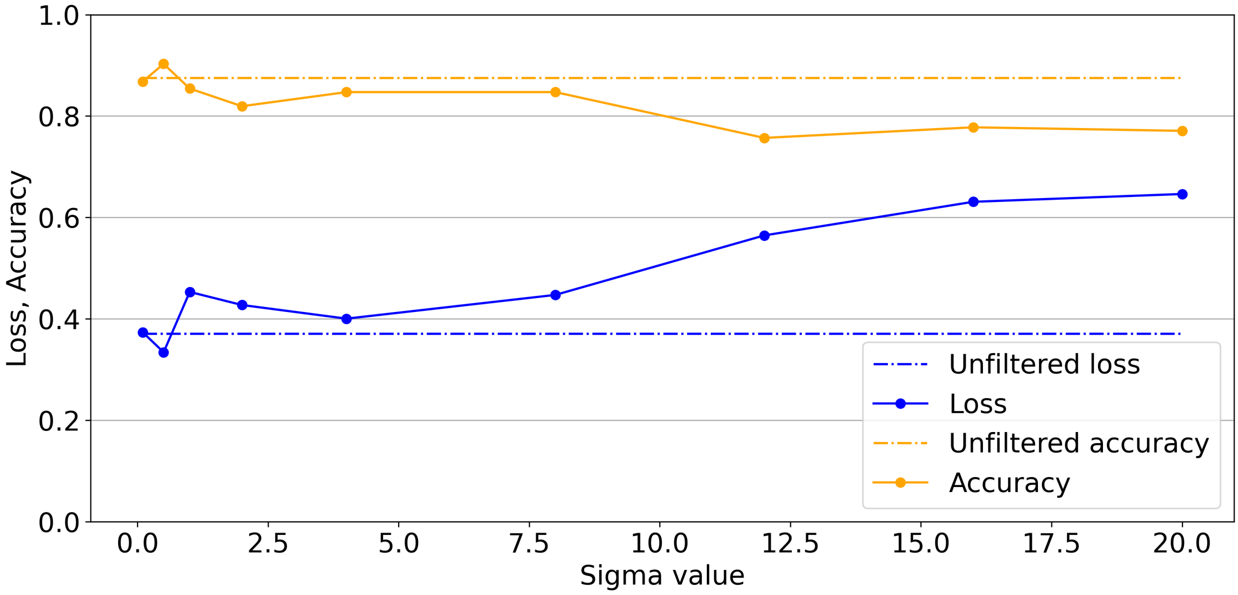


Figure 3.11.: Final validation loss and accuracy at last epoch for model CS-D using strain data points run through a Gaussian filter with different sigma values prior to TCN training and analysis.

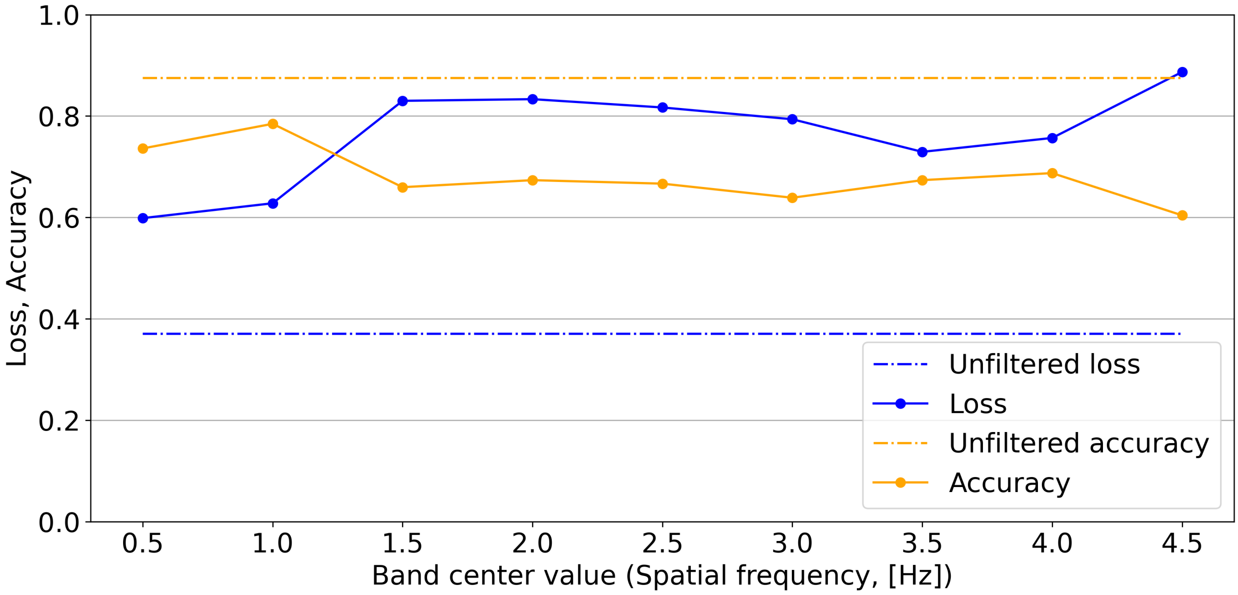


Figure 3.12.: Final validation loss and accuracy at last epoch for model CS-D trained using strain data points filtered for different spatial frequency bands. For each plotted point, the model was trained using a spatial frequency band from 0.5 $\frac{1}{m}$ below to 0.5 $\frac{1}{m}$ above the central frequency on the x-axis.

3.3. Comparisons of Reference Measurements

As the reference measurements for both DOFS were all taken in the real-life noisy environment of the harbour, the effect each specific reference trace had on strain calculation was investigated. For this, a random measurement was selected for each of the two DOFS, and strains were then calculated with the different reference measurements using the procedure described in Section 2.1.3. Comparison matrices of the resulting plots can be seen in Appendix C. For both the helical and straight DOFS in Figure C.1 and C.2, no reference results in equal strain plots. For both the Helical and Straight DOFS, the references at positions $r2s3$ and $r1s3$ resulted in the most similar strain plots, but there are still differences in the position and magnitude of their peaks due to the difference in the noise affecting the two reference measurements.

This calculation exemplifies how reference measurements for acoustic field experiments taken in noisy real-life conditions affect the final strain measurements. However, to be able to accurately extract specific patterns in data, a reference measurement taken in very quiet and stable conditions is highly desirable. This was unfortunately not achieved in current set of experiments, and considerable reference-to-reference varability exists.

Chapter 4.

Discussion of the Results

4.1. Model Performance

4.1.1. Categorical Models

The loss and accuracy curves and confusion matrices presented in Section 3.1.1 indicate that analysing DOFS strain data with categorical TCN models is a viable method for the localisation of subsea acoustic sources. The results of repeated stochastic runs of models CS-D in Figure 3.9 confirm that these results are not a one-off. There is no large difference between the full models and those predicting only one parameter, contradicting the recent findings of [37] where decoupling of prediction tasks with TCN improved the overall model performances. The findings of the sample size investigation shown in Figure 3.8a indicate that the performance plateau is reached at around 60 samples. This can further explain the lack of difference in performance between the full and single-parameter TCN models, despite the full models having only a third of the available number of samples per category to train on compared to the single-parameter models, with 100 vs 300 samples per category, respectively.

Use of Helical vs Straight DOFS Data

Somewhat surprisingly, it was observed that categorical models trained using the straight DOFS data are equally or higher performing than those trained using the helical DOFS data. This is despite the straight DOFS with its inherent linear vertical shape being assumed to be agnostic towards sector angle and work much better for distance predictions. As the helical DOFS is three-dimensional, it was assumed to allow for the prediction of direction in addition to the distance similar to a traditional full azimuth array. But while the straight DOFS in theory should act as a two-dimensional sensor, it is in practice also three-dimensional as the slack applied in the fibre implemented to avoid any pre-straining gives it a three-dimensional curved shape. This, combined with it likely not being positioned exactly normal to the direction of the acoustic waves, can perhaps explain why models using the straight DOFS data are comparably able to predict the direction of the acoustic source as those utilizing helical DOFS data.

In the preceding project thesis [1] it was found that a free jacketed DOFS allowed for TCN models to perform better frequency classification than when using DOFS attached to a structure. As the helical DOFS for this experiment was attached to a tubular mesh structure in order to maintain the helical shape, the mesh presence likely also has a detrimental effect on the performance of CH-F and CH-D, similar to the effect found in the project thesis.

The mesh will unavoidably create resistance and interference for acoustic waves passing through towards the straight DOFS inside. This resistance could act as a filter on acoustic waves in the water and thus altering the periodic waves from the acoustic source. If this filtering affects the higher frequencies, it could have a performance-increasing effect as it filters out noise similar to the Gaussian filter with the lowest σ-values as shown in Figure 3.11.

The helical DOFS also allows for more data points along the tubular structure compared to the straight fibre. This is because it is a physically longer fibre which should give the models longer data sequences to base their predictions on. This will however inherently also raise the total amount of noise in the measurements, potentially further decreasing the performance of models trained with helical DOFS data. While time constraints meant that this could not be investigated further in this thesis, training models using data points of varying lengths could be performed to investigate how sequence length affects model performance.

By analysing the confusion matrices for models CH-F and CS-F in Figure 3.4a and Figure 3.4b respectively, off-diagonal cells are more populated for CH-F compared to that of CS-F, reflecting the slightly lower performance of the CH-F model. It can further be noted that CS-F has a higher local confusion between the sectors at the furthest ($3\ m$) distance. Given that this seemingly is a very localised effect, it could indicate that the upper range limit of the method is reached.

Possible Performance Improvements

The investigation of the effect of sample size on model performance shows a clear plateau being reached at approximately 60 training samples. It is therefore unlikely that increasing the dataset further without also adding new positions would result in a meaningful increase in model performance. Further model tuning is therefore likely needed to increase model performance past 90%. It is however hard to know whether higher performance is feasible without performing tests utilising the datasets specific to the problem to be solved. Performance improvements could also require more extensive architectural changes, such as the addition of a Support Vector Machine (SVM) as a final layer as suggested by Shi et al. [32], or the use of raw OBR data to avoid information loss in pre-processing completely.

4.1.2. Regression Models

The consistent loss values and loss curves for the regression models, along with the resulting curve of the repeated stochastic study in Figure 3.10, show that the TCN regression model architecture used is stable. But as the loss values do not have any practical interpretation other than as a metric to compare similar models, the box plots were used to analyse the performance of each regression model further. As shown by the boxplots in Figure 3.7 and 3.5, there is a trend showing an increase in the mean values of predictions for increasing expected value for both predicted parameters, although they are all skewed towards the middle values, 2 m and 90° respectively. However, as the box and whiskers for each plot show, individual predictions are spread over a large range. There is also no clear pattern in how these values are spread related to the parameters, models or dataset used. This means that any individual prediction is not reliable, and as the median values are all skewed toward the middle, taking an average of several predictions for a single event would likely not give correct values either. As a result, TCN regression models using DOFS strain data for acoustic source localization are considered unusable in their current form, despite there being indications in the mean value trends for each parameter that a regression TCN model could work.

Possible Performance Improvements

Similar to the categorical models, the sample size investigation shows that increasing the sample size would likely not give any meaningful increase in performance, as performance improvements stagnated past 60 samples. A larger change to the model architecture is therefore likely required to achieve practically useful predictions. As regression models predict continuous values, it is further suspected that the data set must be populated with measurements from more acoustic source positions in order for the models to develop a sufficient mapping between input and output values.

4.2. Effects of Filtering on Model Performance

The decreased model performance for both increasing sigma values in the Gaussian filter and for higher spatial frequency band filtering, as shown in Figure 3.11 and Figure 3.12 respectively, indicates that the information the models utilise for predictions resides in the strain-position patterns with lower frequencies. Unlike the original assumption that any filtering would lead to an information loss, the small performance increase for $\sigma = 0.1$ show that some prefiltering of the data can be advantageous. The low filter value only removes the highest frequency spikes as shown in Figure 2.5 and indicates that these generally can be considered outliers (i.e. actual noise) with low information value. Similarly, the larger drop in performance past $\sigma = 8.0$ indicates that the Gaussian filter then started removing frequencies containing information that significantly contributes to the prediction ability of the model, resulting in much lower performance. A similar effect was observed for band filtering of bands with centre values 1.5 $\frac{1}{m}$ and higher, suggesting that $sigma = 10.0$ for a Gaussian filter filters out similar frequencies to the band filter having a high-pass cut-off of 1.0. The consistently lower performance of the TCN model with band filter is likely due to it filtering out both high and low frequencies, unlike the Gaussian filter, which due to its function as a low-pass filter only filters out higher frequencies. This further demonstrates that there is no one spatial frequency range that contains all information needed, but that the models rather utilise patterns from a broader spectrum of frequencies when calculating predictions for a certain data point.

4.3. Potential Sources of Error

All measurements for one source position were taken before moving the acoustic source to the next position. This means that any local effect could create a specific effect in selected elements in measurement data, helping the model make correct predictions by learning to detect this temporally transient pattern (e.g. due to passing boats) instead of permanent ones related to distance and direction of the acoustic source. Given the sample size needed, combined with the fact that moving the acoustic source between positions and taking reference measurements was a manual process, avoiding these patterns emerging would have unfeasibly extended the duration of the data collection process for this project. It is therefore likely that such alien patterns exist for some subsets of the measurement data. Having less control over experimental conditions is a common challenge for measurements in real-life test setups. At the same time, chaos from real-life events often makes field experiments more valuable compared to well-controlled laboratory conditions where such effects are impossible to duplicate. It is hoped and assumed that patterns related to the distance and direction of the acoustic waves still are the predominant features of the measurement set obtained for each position of the acoustic source.

As the data collection spanned over several hours per session, having only one reference measurement for the entire session was considered to increase the risk of local, periodic acoustic events affecting the data. Having to take one reference measurement for each measurement sample would however unfeasibly increased the time needed for collecting a data set of the suspected needed size. The use of one reference measurement per position was therefore chosen as a trade-off to avoid the effects of temporally transient acoustic events while also streamlining the data collection process as much as possible to maximise the data set size available for training. It is important to emphasize that the sample size needed was unknown at the time of planning and execution, along with the time for data collection being very limited.

4.3.1. Impact of Noise in Reference Measurements

As discussed earlier, the reference measurement defines the stationary state of the optical fibre. However, the reference measurements of all DOFS states were taken in the noisy real-life environment of the harbour. This means that any noise in the reference introduces patterns in the extracted data that would be unique to that reference and consequently affect each buzzer position strain data that uses this reference. As the vastly different strain plots in reference comparisons in Appendix C reveal, this is indeed the case. These unique patterns introduced by noisy reference data mean that there is a risk of the models learning to recognize these noise effects instead of the patterns related to the acoustic source and its position.

> *As an analogy, imagine tasking a system with sorting post-it notes based on what is written on them. Instead, it sorts them by the colour of the note, as it turns out that the text has been written on notes of different colours, with a unique colour for each group. While the outcome might be correct, the system has possibly never had to find a pattern in what is written on the notes. This also means that if the colours were to change, the system would not be able to sort them correctly anymore, despite all previous results indicating that it is sorting them correctly.*

This could explain the large performance difference between the categorical and regression models, as it is likely easier for the models to categorise different unique strain patterns affected by noisy references than to predict continuous values from groups of patterns that include many different noise references. The early convergence of models, with no significant improvements or indications of overfitting as the number of epochs increases further, could be a sign of this being the case. The models would here be 'stuck' at these noise-affected patterns functioning as a local performance maximum, making them unable to look for the global maximum related to the patterns created by the acoustic source and its position. Low confusion of the categorical models does however suggest that these noise patterns, if detected, are not fully dominating. But whether the confusion is due to increased noise in individual measurements or if the models are detecting patterns created by the acoustic source is however unclear and would require further tests with new data.

The issue of reference measurements should be investigated further, e.g. by using one common reference measurement or by using a running reference, meaning the previous measurement is used as a reference for calculating strains in the next measurement. With the tools and software available for this master's thesis, doing this was considered to unreasonably impact the other investigations performed within the time frame given.

Chapter 5.

Conclusions

This investigation was undertaken to explore the practical viability of using Temporal Convolutional Network-based (TCN) machine learning analysis on Distributed Optical Fibre Sensor (DOFS) strain data in order to locate acoustic sources in a real-life subsea environment. The prototype task was to predict the distance and direction of a nearby placed acoustic buzzer.

The results show that categorical TCN models are able to categorise DOFS strain data with respect to the distance to and direction of the acoustic source in an underwater environment, with the models reaching ca. $70-80\%$ validation accuracies for both helical and straight DOFS. The regression models also converge, but the resulting predictions have too high a variance to provide practically usable predictions at their current state. Unlike previous works, no significant difference in performance was found between the models predicting both or only one positional parameter for either the categorical or regression models.

Models using data generated from helically positioned DOFS measurements were found to have slightly lower performance compared to the models using straight DOFS. It is suspected that the tubular metal mesh structure used to achieve the helical shape of the DOFS influences the measurements, similar to the findings in the preceding project thesis.

Testing of models trained with different levels of pre-filtered data revealed that TCN models perform best with unfiltered data and data filtered using Gaussian filters with sigma-values up to $\sigma = 0.1$, backing up the hypothesis of filtering leading to information loss, while Deep Learning-models are able to filter out this noise on their own with minimal information loss. Using a band filter further revealed that there is no smaller spatial frequency range containing all information needed for the model to make predictions, but that TCN models rather rely on a larger range of spatial frequencies for the extraction of relevant data and patterns in order to make predictions.

This was the first experiment outside a controlled lab environment, revealing significant problems related to the use of reference measurements. It was found that using several reference measurements taken in noisy environments introduces patterns in the extracted strains, greatly affecting the ability of the model to extract relevant data and consequently also lowering their final performance.

Chapter 6.

Future Work

The results presented have revealed a need to establish new methods for collecting reference measurements used for strain extraction. For this, further investigation into how noisy reference measurements affect the final strain measurements is likely needed.

As an alternative, an investigation into how TCN performs when the raw OBR measurements are used instead of calculating strains is of interest. Not only would this likely omit the problem of good reference measurements altogether by not having to extract strains, but the calculation time needed in preprocessing would also decrease, in addition to avoiding the inherent information loss of the strain extraction process.

References

[1] J. B. Bugge, *Estabilishing hidden patterns in DOFS data by TCN based machine learning*, Unpublished. Available upon request, 2022.

[2] 'Nord Stream blast 'blew away 50 metres of pipe',' en-GB, *BBC News*, [Online]. Available: `https://www.bbc.com/news/world-europe-63297085` (visited on 11/05/2023).

[3] M. Landrø, L. Bouffaut, H. J. Kriesell *et al.*, 'Sensing whales, storms, ships and earthquakes using an Arctic fibre optic cable,' en, *Scientific Reports*, vol. 12, no. 1, p. 19 226, Nov. 2022, Number: 1 Publisher: Nature Publishing Group, ISSN: 2045-2322. DOI: `10.1038/s41598-022-23606-x`. [Online]. Available: `https://www.nature.com/articles/s41598-022-23606-x` (visited on 23/05/2023).

[4] Y. Yang, H. Zhang, Y. Li and Y. Li, 'Pipeline Safety Early Warning by Multifeature-Fusion CNN and LightGBM Analysis of Signals From Distributed Optical Fiber Sensors,' *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–13, Jan. 2021, MAG ID: 3199110137. DOI: `10.1109/tim.2021.3092518`.

[5] E. Saeter, K. Lasn, F. Nony and A. T. Echtermeyer, 'Embedded optical fibres for monitoring pressurization and impact of filament wound cylinders,' en, *Composite Structures*, vol. 210, pp. 608–617, Feb. 2019, ISSN: 0263-8223. DOI: `10.1016/j.compstruct.2018.11.051`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S0263822318325662` (visited on 13/12/2022).

[6] *Subsea internet cables could help detect earthquakes - BBC News*. [Online]. Available: `https://www.bbc.com/news/technology-61506705` (visited on 11/05/2023).

[7] K. Peters, 'Polymer optical fiber sensors—a review,' en, *Smart Materials and Structures*, vol. 20, no. 1, p. 013 002, Dec. 2010, ISSN: 0964-1726. DOI: `10.1088/0964-1726/20/1/013002`. [Online]. Available: `https://dx.doi.org/10.1088/0964-1726/20/1/013002` (visited on 11/12/2022).

[8] B. Culshaw, 'Optical fiber sensor technologies: Opportunities and-perhaps-pitfalls,' *Journal of Lightwave Technology*, vol. 22, no. 1, pp. 39–50, Jan. 2004, Conference Name: Journal of Lightwave Technology, ISSN: 1558-2213. DOI: `10.1109/JLT.2003.822139`.

[9] A. Othonos, 'Fiber Bragg gratings,' *Review of Scientific Instruments*, vol. 68, no. 12, pp. 4309–4341, Dec. 1997, ISSN: 0034-6748. DOI: `10.1063/1.1148392`. [Online]. Available: `https://doi.org/10.1063/1.1148392` (visited on 16/05/2023).

[10] *ChatGPT*. [Online]. Available: `https://chat.openai.com` (visited on 11/05/2023).

[11] *Bard*, en. [Online]. Available: `https://bard.google.com` (visited on 11/05/2023).

[12] M. Mohri, A. Rostamizadeh and A. Talwalkar, *Foundations of Machine Learning, Second Edition* (Adaptive Computation and Machine Learning), English. Cambridge, Massachusetts: The MIT Press, 2018, vol. Second edition, ISBN: 978-0-262-03940-6.

[13] C. Lea, M. D. Flynn, R. Vidal, A. Reiter and G. D. Hager, 'Temporal Convolutional Networks for Action Segmentation and Detection,' 2017, pp. 156–165. [Online]. Available: `https://openaccess.thecvf.com/content_cvpr_2017/html/Lea_Temporal_Convolutional_Networks_CVPR_2017_paper.html` (visited on 17/12/2022).

[14] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning* (Adaptive Computation and Machine Learning), English. Cambridge, Massachusetts: The MIT Press, 2016, ISBN: 978-0-262-03561-3.

[15] H. Wu, Y. Qian, W. Zhang and C. Tang, 'Feature extraction and identification in distributed optical-fiber vibration sensing system for oil pipeline safety monitoring,' *Photonic Sensors*, vol. 7, no. 4, pp. 305–310, Sep. 2017, MAG ID: 2754863095. DOI: `10.1007/s13320-017-0360-1`.

[16] J. Tejedor, J. Macias-Guarasa, H. F. Martins, S. Martin-Lopez and M. Gonzalez-Herraez, 'A Multi-Position Approach in a Smart Fiber-Optic Surveillance System for Pipeline Integrity Threat Detection,' *Electronics*, vol. 10, no. 6, p. 712, 2021, MAG ID: 3138486465. DOI: `10.3390/electronics10060712`.

[17] H.-H. Zhu, W. Liu, T. Wang, J.-W. Su and B. Shi, 'Distributed Acoustic Sensing for Monitoring Linear Infrastructures: Current Status and Trends,' *Sensors*, vol. 22, no. 19, pp. 7550–7550, Oct. 2022, MAG ID: 4303981137. DOI: `10.3390/s22197550`.

[18] D. F. Kandamali, X. Cao, M. Tian, Z. Jin, H. Dong and K. Yu, 'Machine learning methods for identification and classification of events in phase-OTDR systems: A review.,' *Applied Optics*, vol. 61, no. 11, pp. 2975–2997, Apr. 2022, MAG ID: 4226022847. DOI: `10.1364/ao.444811`.

[19] J. Li, Y. Wang, P. Wang *et al.*, 'Pattern Recognition for Distributed Optical Fiber Vibration Sensing: A Review,' *IEEE Sensors Journal*, vol. 21, no. 10, pp. 11 983–11 998, May 2021, ISSN: 1558-1748. DOI: `10.1109/JSEN.2021.3066037`.

[20] Z. Peng, J. Jian, H. Wen *et al.*, 'Distributed fiber sensor and machine learning data analytics for pipeline protection against extrinsic intrusions and intrinsic corrosions,' EN, *Optics Express*, vol. 28, no. 19, pp. 27 277–27 292, Sep. 2020, Publisher: Optica Publishing Group, ISSN: 1094-4087. DOI: `10.1364/OE.397509`. [Online]. Available: `https://opg.optica.org/oe/abstract.cfm?uri=oe-28-19-27277` (visited on 24/04/2023).

[21] C. Huynh, C. Hibert, C. Jestin *et al.*, 'Real-Time Classification of Anthropogenic Seismic Sources from Distributed Acoustic Sensing Data: Application for Pipeline Monitoring,' *Seismological Research Letters*, Jul. 2022, MAG ID: 4288096413. DOI: `10.1785/0220220078`.

[22] L. Zhongqi, J. Zhang, J. Zhang, M. Wang, Y. Zhong and F. Peng, 'Fiber distributed acoustic sensing using convolutional long short-term memory network: A field test

on high-speed railway intrusion detection.,' *Optics Express*, vol. 28, no. 3, pp. 2925–2938, Feb. 2020, MAG ID: 3001715844. DOI: 10.1364/oe.28.002925.

[23] M. Aktas, T. Akgun, M. U. Demircin and D. Buyukaydin, 'Deep learning based multi-threat classification for phase-OTDR fiber optic distributed acoustic sensing applications,' vol. 10208, pp. 75–92, Apr. 2017, MAG ID: 2609937040. DOI: 10.1117/12.2262108.

[24] Y. Bai, J. Xing, F. Xie, S. Liu and J. Li, 'Detection and identification of external intrusion signals from 33km optical fiber sensing system based on deep learning,' en, *Optical Fiber Technology*, vol. 53, p. 102 060, Dec. 2019, ISSN: 1068-5200. DOI: 10.1016/j.yofte.2019.102060. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S106852001930077X (visited on 24/04/2023).

[25] X. Chen, C. Xu and C. Xu, 'Disturbance pattern recognition based on an ALSTM in a long-distance phase-OTDR sensing system,' *Microwave and Optical Technology Letters*, vol. 62, no. 1, pp. 168–175, Jan. 2020, MAG ID: 2971632157. DOI: 10.1002/mop.32025.

[26] M. M. Khan, P. Jaiswal, M. Mishra and R. K. Sonkar, 'Strategic-cum-Domestic vehicular movement detection through Deep Learning approach using designed Fiber-Optic distributed vibration sensor,' Apr. 2022, MAG ID: 4285816527. DOI: 10.1109/i2ct54291.2022.9824045.

[27] Y. Li, K. Li, X. Liu *et al.*, 'A hybrid machine learning framework for joint SOC and SOH estimation of lithium-ion batteries assisted with fiber sensor measurements,' en, *Applied Energy*, vol. 325, p. 119 787, Nov. 2022, ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2022.119787. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261922010662 (visited on 24/04/2023).

[28] S. Dhanalakshmi, P. Nandini, S. Rakshit *et al.*, 'Fiber Bragg grating sensor-based temperature monitoring of solar photovoltaic panels using machine learning algorithms,' en, *Optical Fiber Technology*, vol. 69, p. 102 831, Mar. 2022, ISSN: 1068-5200. DOI: 10.1016/j.yofte.2022.102831. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1068520022000141 (visited on 24/04/2023).

[29] Y. Zhuang, Q. Yang, T. Han *et al.*, 'Fiber optic sensor embedded smart helmet for real-time impact sensing and analysis through machine learning,' en, *Journal of Neuroscience Methods*, vol. 351, p. 109 073, Mar. 2021, ISSN: 0165-0270. DOI: 10.1016/j.jneumeth.2021.109073. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S016502702100008X (visited on 24/04/2023).

[30] L. Shiloh, A. Eyal and R. Giryes, 'Deep Learning Approach for Processing Fiber-Optic DAS Seismic Data,' EN, in *26th International Conference on Optical Fiber Sensors (2018), paper ThE22*, Optica Publishing Group, Sep. 2018, ThE22. DOI: 10.1364/OFS.2018.ThE22. [Online]. Available: https://opg.optica.org/abstract.cfm?uri=OFS-2018-ThE22 (visited on 24/04/2023).

[31] Y. Shi, Y. Li, Y. Zhang, Z. Zhuang and T. Jiang, 'An Easy Access Method for Event Recognition of phase-OTDR Sensing System Based on Transfer Learning,' *Journal of Lightwave Technology*, vol. 39, no. 13, pp. 4548–4555, Jul. 2021, Conference

Name: Journal of Lightwave Technology, ISSN: 1558-2213. DOI: 10.1109/JLT.2021.3070583.

[32] Y. Shi, Y. Wang, L. Wang, L. Zhao and Z. Fan, 'Multi-event classification for phase-OTDR distributed optical fiber sensing system using deep learning and support vector machine,' en, *Optik*, vol. 221, p. 165 373, Nov. 2020, ISSN: 0030-4026. DOI: 10.1016/j.ijleo.2020.165373. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0030402620312092 (visited on 24/04/2023).

[33] M. Bublin, 'Event Detection for Distributed Acoustic Sensing: Combining Knowledge-Based, Classical Machine Learning, and Deep Learning Approaches,' *Sensors*, vol. 21, no. 22, p. 7527, Nov. 2021, MAG ID: 3214640732. DOI: 10.3390/s21227527.

[34] S. Liehr, C. Borchardt, S. Münzenberger and S. Muenzenberger, 'Long-distance fiber optic vibration sensing using convolutional neural networks as real-time denoisers.,' *Optics Express*, vol. 28, no. 26, pp. 39 311–39 325, 2020, MAG ID: 3098545480. DOI: 10.1364/oe.402789.

[35] D. Desai and N. Mehendale, 'A Review on Sound Source Localization Systems,' en, *Archives of Computational Methods in Engineering*, vol. 29, no. 7, pp. 4631–4642, Nov. 2022, ISSN: 1886-1784. DOI: 10.1007/s11831-022-09747-2. [Online]. Available: https://doi.org/10.1007/s11831-022-09747-2 (visited on 13/06/2023).

[36] K. Guirguis, C. Schorn, A. Guntoro, S. Abdulatif and B. Yang, 'SELD-TCN: Sound Event Localization & Detection via Temporal Convolutional Networks,' in *2020 28th European Signal Processing Conference (EUSIPCO)*, ISSN: 2076-1465, Jan. 2021, pp. 16–20. DOI: 10.23919/Eusipco47968.2020.9287716.

[37] S. Song, C. Zhang and X. You, 'Decoupling Temporal Convolutional Networks Model in Sound Event Detection and Localization,' en-US, *Journal of Internet Technology*, vol. 24, no. 1, pp. 89–99, Jan. 2023, Number: 1, ISSN: 2079-4029. [Online]. Available: https://jit.ndhu.edu.tw/article/view/2842 (visited on 25/01/2023).

[38] L. Bouffaut, K. Taweesintananon, H. J. Kriesell *et al.*, 'Eavesdropping at the Speed of Light: Distributed Acoustic Sensing of Baleen Whales in the Arctic,' *Frontiers in Marine Science*, vol. 9, 2022, ISSN: 2296-7745. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fmars.2022.901348 (visited on 23/01/2023).

[39] D. Rivet, B. de Cacqueray, A. Sladen, A. Roques and G. Calbris, 'Preliminary assessment of ship detection and trajectory evaluation using distributed acoustic sensing on an optical fiber telecom cable,' *The Journal of the Acoustical Society of America*, vol. 149, no. 4, pp. 2615–2627, Apr. 2021, ISSN: 0001-4966. DOI: 10.1121/10.0004129. [Online]. Available: https://doi.org/10.1121/10.0004129 (visited on 16/06/2023).

[40] *Smb-e1550h_datasheet*. [Online]. Available: https://fiber-optic-catalog.ofsoptics.com/documents/pdf/BF06158-02-Clearlite-POLY-1550-17-Optical-Fiber-web.pdf (visited on 14/05/2023).

[41] Lunainc, *OBR - Overview and Applications.pdf*. [Online]. Available: https://lunainc.com/sites/default/files/assets/files/resource-library/OBR%20-%20Overview%20and%20Applications.pdf (visited on 08/10/2022).

[42] L. Thévenaz, *Advanced Fiber Optics: Concepts and Technology*, en. EPFL Press, 2011, ISBN: 978-2-940222-43-8.

[43] Lunainc, *LUNA-Data-Sheet-OBR-4600-V2.pdf*. [Online]. Available: https://lunainc.com/sites/default/files/assets/files/resource-library/LUNA-Data-Sheet-OBR-4600-V2.pdf (visited on 13/12/2022).

[44] S. Wang, K. Lasn, C. W. Elverum, D. Wan and A. Echtermeyer, 'Novel in-situ residual strain measurements in additive manufacturing specimens by using the Optical Backscatter Reflectometry,' eng, *Additive Manufacturing*, 2020, Publisher: Elsevier, ISSN: 2214-8604. DOI: 10.1016/j.addma.2020.101040. [Online]. Available: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2636181 (visited on 05/09/2022).

[45] S. Wang, E. Sæter and K. Lasn, 'Comparison of DOFS Attachment Methods for Time-Dependent Strain Sensing,' en, *Sensors*, vol. 21, no. 20, p. 6879, Jan. 2021, Publisher: Multidisciplinary Digital Publishing Institute, ISSN: 1424-8220. DOI: 10.3390/s21206879. [Online]. Available: https://www.mdpi.com/1424-8220/21/20/6879 (visited on 05/09/2022).

[46] S. Wang and K. Lasn, 'Integration of optical fibre sensors by material extrusion 3-D printing – The effect of bottom interlayer thickness,' en, *Materials & Design*, vol. 221, p. 110 914, Sep. 2022, ISSN: 0264-1275. DOI: 10.1016/j.matdes.2022.110914. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0264127522005366 (visited on 23/05/2023).

[47] S. Wang, 'Integration and performance of distributed optical fibre sensors in thermoplastics and thermoplastic composites,' eng, Doctoral thesis, NTNU, 2023. [Online]. Available: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/3063356 (visited on 23/05/2023).

[48] V. Usenco, 'ANN Analysis of DOFS Measurements Compromised by Mechanical and Acoustic Vibrations,' eng, Accepted: 2022-10-07T17:32:07Z, M.S. thesis, NTNU, 2022. [Online]. Available: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/3024724 (visited on 13/12/2022).

[49] A. v. d. Oord, S. Dieleman, H. Zen *et al.*, *WaveNet: A Generative Model for Raw Audio*, arXiv:1609.03499 [cs], Sep. 2016. DOI: 10.48550/arXiv.1609.03499. [Online]. Available: http://arxiv.org/abs/1609.03499 (visited on 15/02/2023).

[50] S. Bai, J. Z. Kolter and V. Koltun, *An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling*, arXiv:1803.01271 [cs], Apr. 2018. DOI: 10.48550/arXiv.1803.01271. [Online]. Available: http://arxiv.org/abs/1803.01271 (visited on 26/05/2023).

[51] P. Remy, *Temporal Convolutional Networks for Keras*. [Online]. Available: https://github.com/philipperemy/keras-tcn (visited on 11/12/2022).

[52] *Idun – High Performance Computing Group*, en-US. [Online]. Available: https://www.hpc.ntnu.no/idun/ (visited on 22/05/2023).

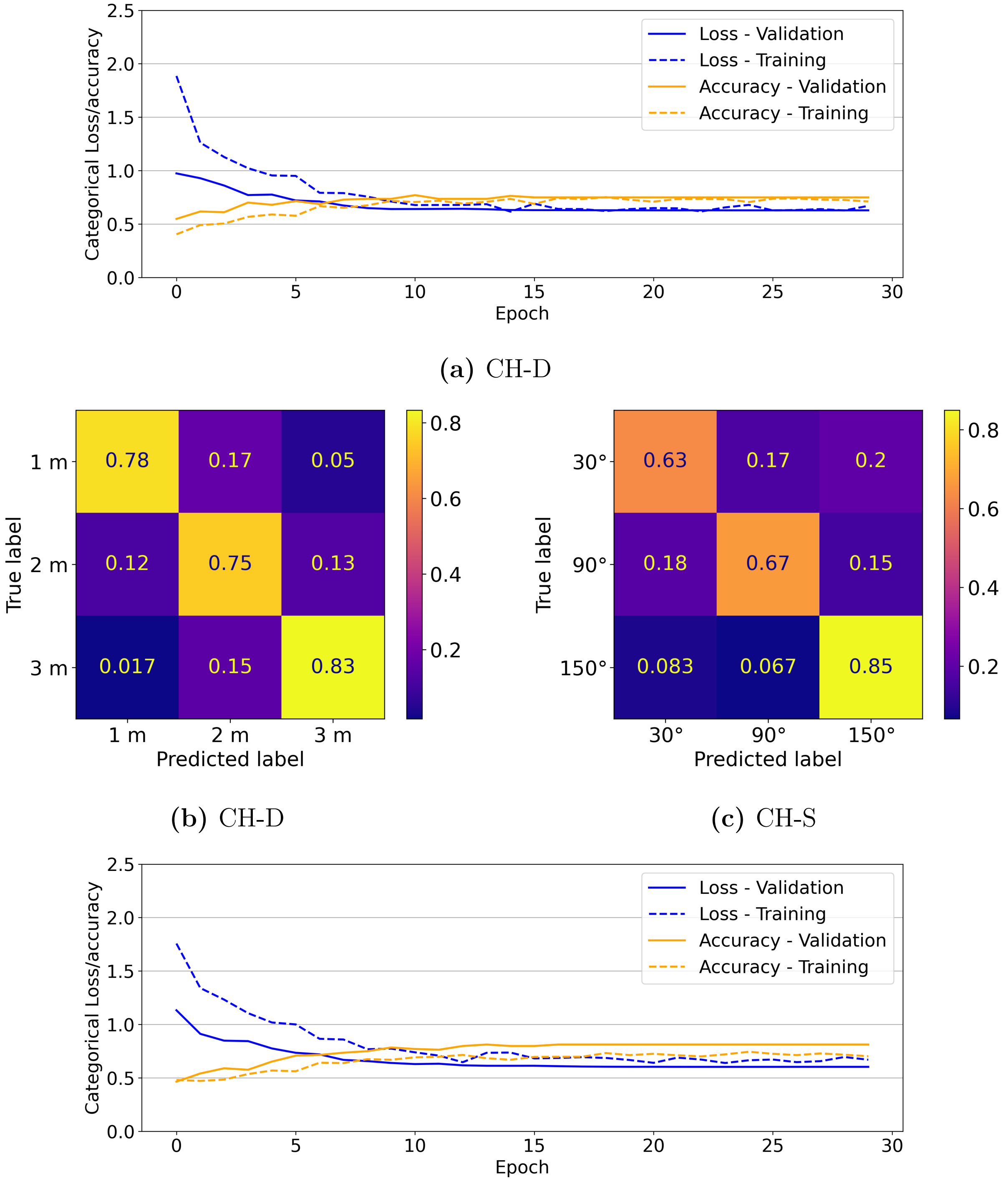
[53] *GitHub*, Link to repository available upon request. [Online]. Available: https://github.com/ (visited on 24/05/2023).

[54] *Norgeskart*. [Online]. Available: https://norgeskart.no/#!?project=norgeskart&layers=1002&zoom=12&lat=7042035.83&lon=269513.82&markerLat=7042035.8279539505&markerLon=269513.82149244414&p=searchOptionsPanel&sok=Nedre%20Ila (visited on 27/05/2023).

[55] *Trondhjems Seilforening – Stiftet 1895*, nb-NO, May 2023. [Online]. Available: https://www.trondhjems-seilforening.no/ (visited on 27/05/2023).

Appendix A.

Results of all standard Categorical Models



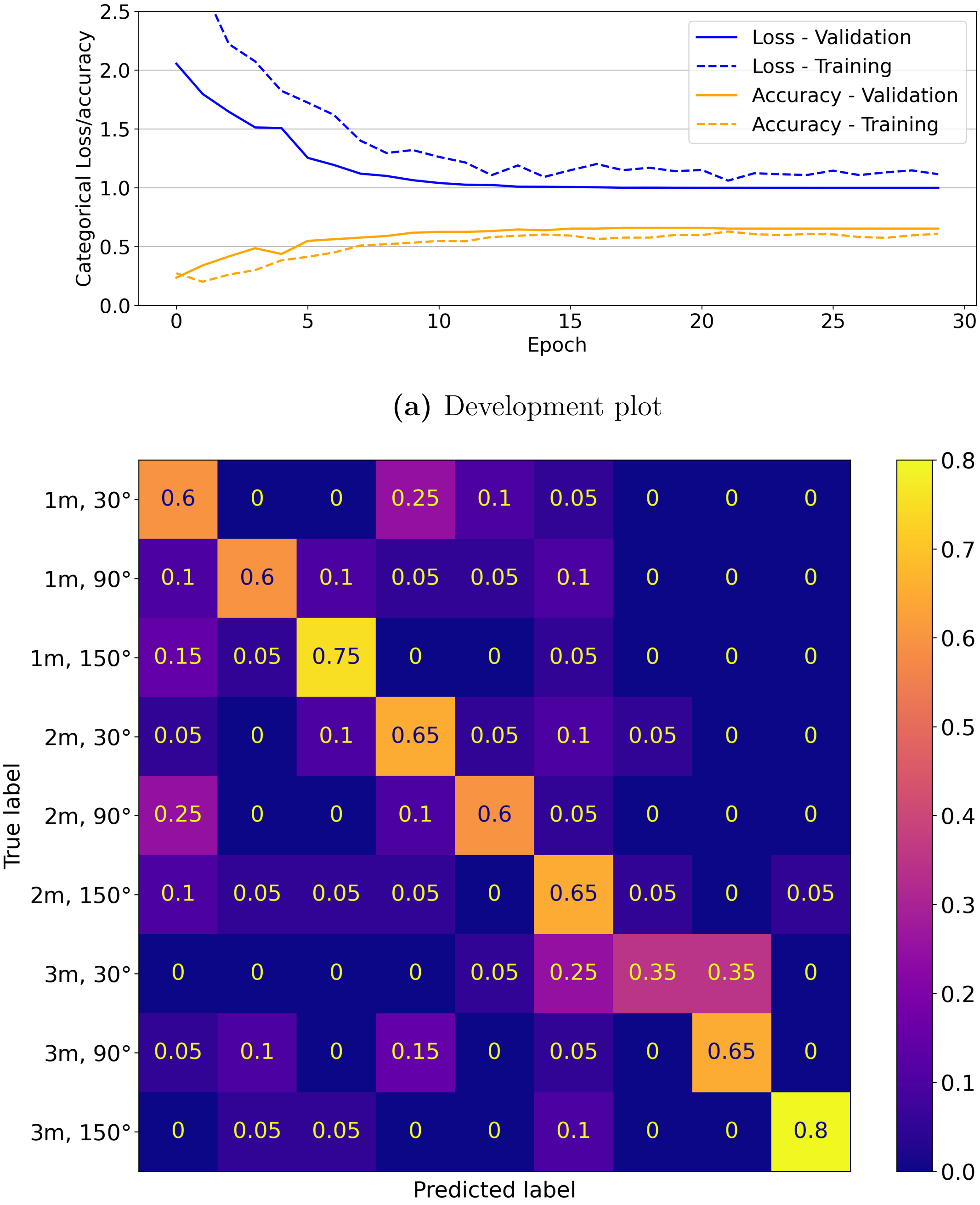
(a) CH-D

(b) CH-D

(c) CH-S

(d) CH-S

Figure A.1.: Single-parameter models trained with helical DOFS dataset

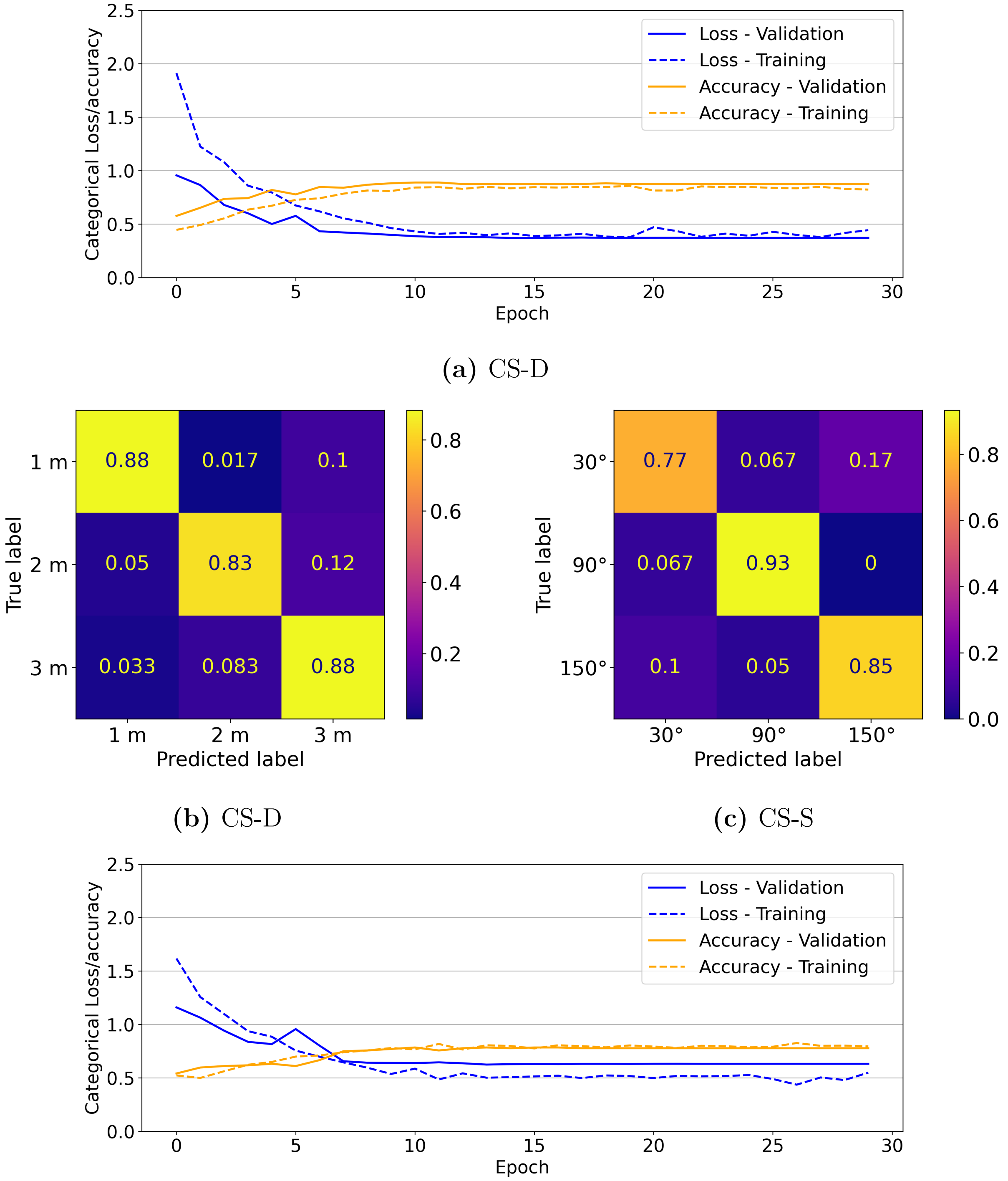


(a) Development plot

1m, 30°	0.6	0	0	0.25	0.1	0.05	0	0	0
1m, 90°	0.1	0.6	0.1	0.05	0.05	0.1	0	0	0
1m, 150°	0.15	0.05	0.75	0	0	0.05	0	0	0
2m, 30°	0.05	0	0.1	0.65	0.05	0.1	0.05	0	0
2m, 90°	0.25	0	0	0.1	0.6	0.05	0	0	0
2m, 150°	0.1	0.05	0.05	0.05	0	0.65	0.05	0	0.05
3m, 30°	0	0	0	0	0.05	0.25	0.35	0.35	0
3m, 90°	0.05	0.1	0	0.15	0	0.05	0	0.65	0
3m, 150°	0	0.05	0.05	0	0	0.1	0	0	0.8

(b) Confusion matrix

Figure A.2.: CH-F



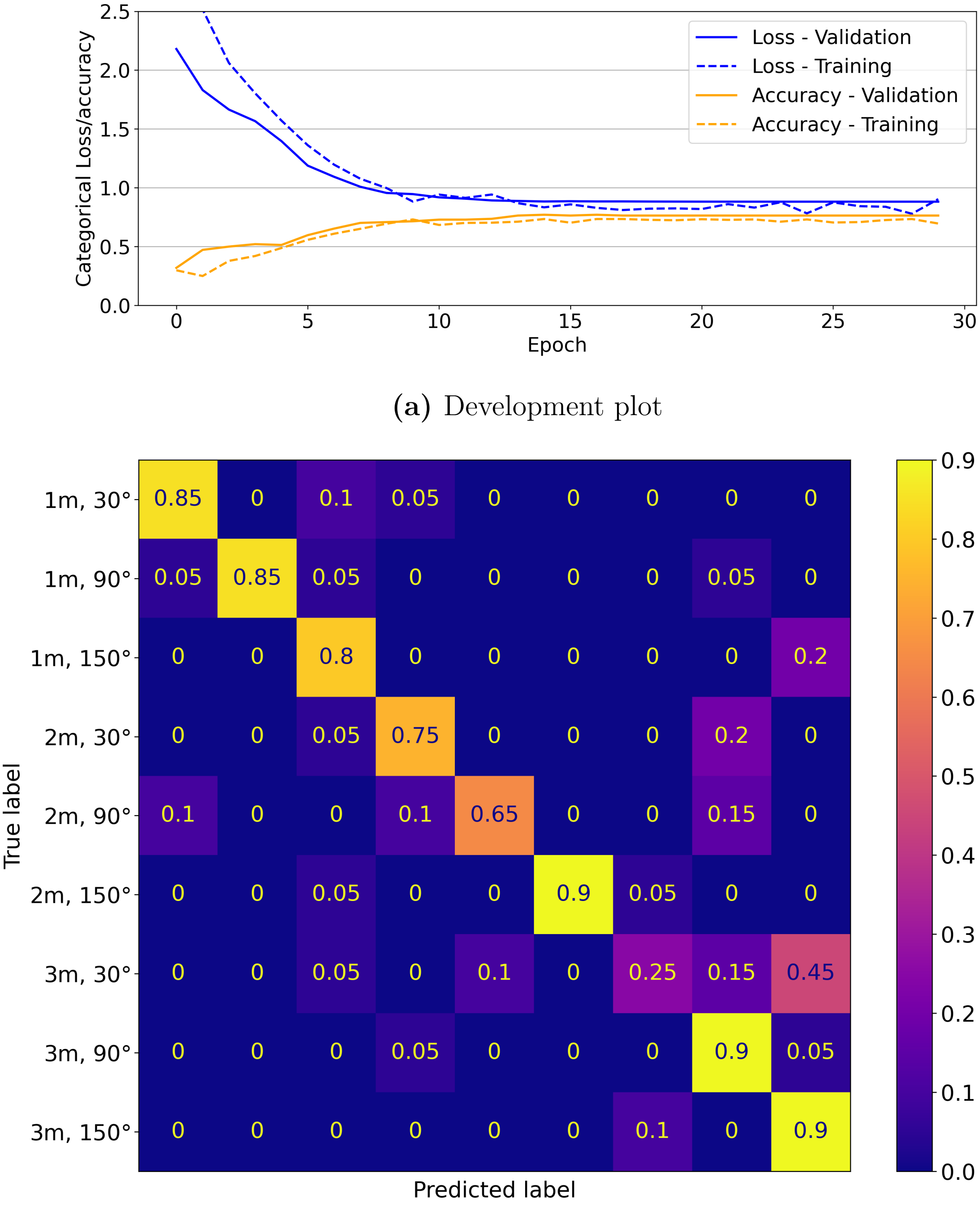
(a) CS-D

(b) CS-D

(c) CS-S

(d) CS-S

Figure A.3.: Single-parameter models trained with straight DOFS dataset



1m, 30°	0.85	0	0.1	0.05	0	0	0	0	0
1m, 90°	0.05	0.85	0.05	0	0	0	0	0.05	0
1m, 150°	0	0	0.8	0	0	0	0	0	0.2
2m, 30°	0	0	0.05	0.75	0	0	0	0.2	0
2m, 90°	0.1	0	0	0.1	0.65	0	0	0.15	0
2m, 150°	0	0	0.05	0	0	0.9	0.05	0	0
3m, 30°	0	0	0.05	0	0.1	0	0.25	0.15	0.45
3m, 90°	0	0	0	0.05	0	0	0	0.9	0.05
3m, 150°	0	0	0	0	0	0	0.1	0	0.9

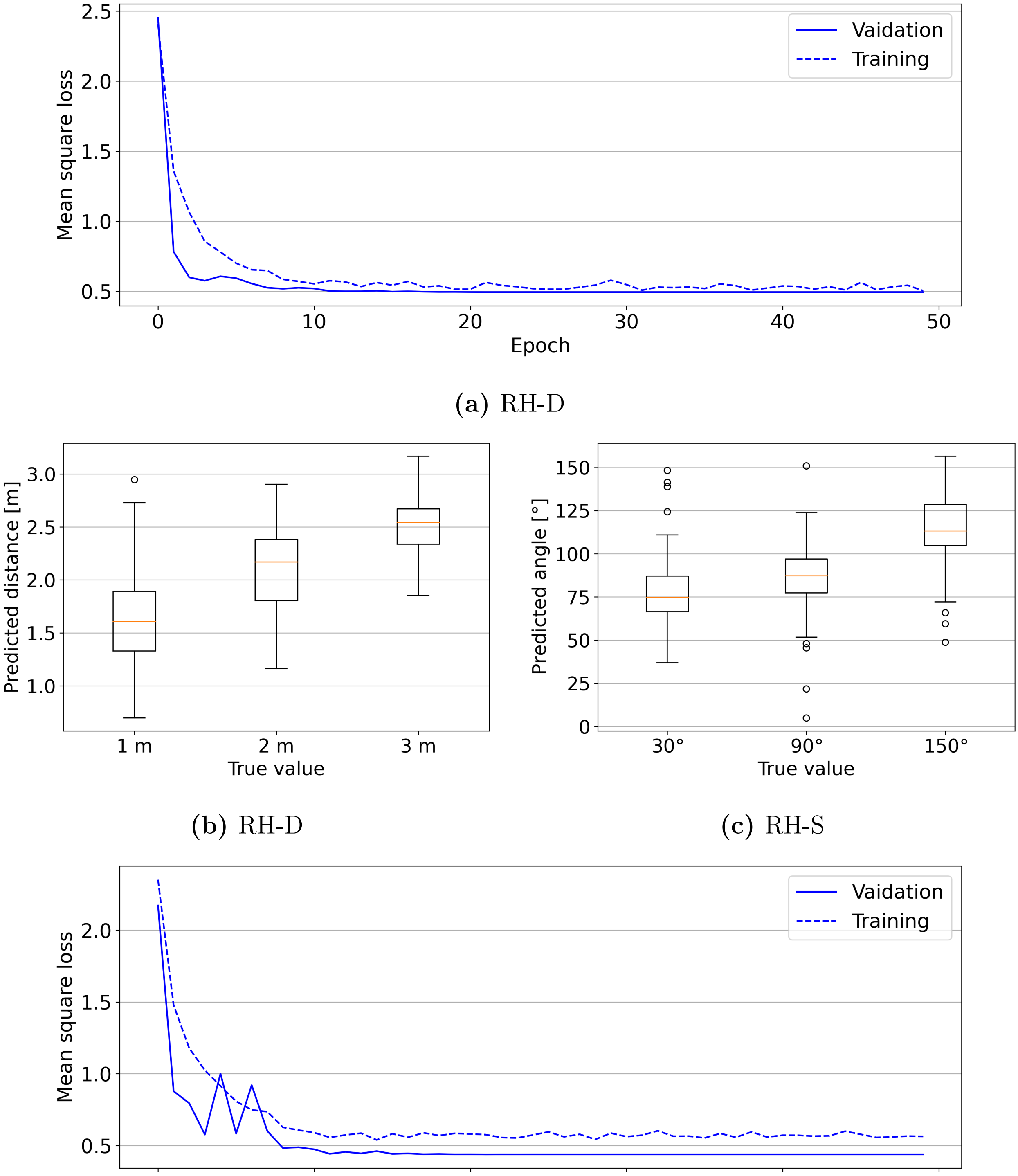
(a) Development plot

(b) Confusion matrix

Figure A.4.: CS-F

Appendix B.

Results of all standard Regression Models



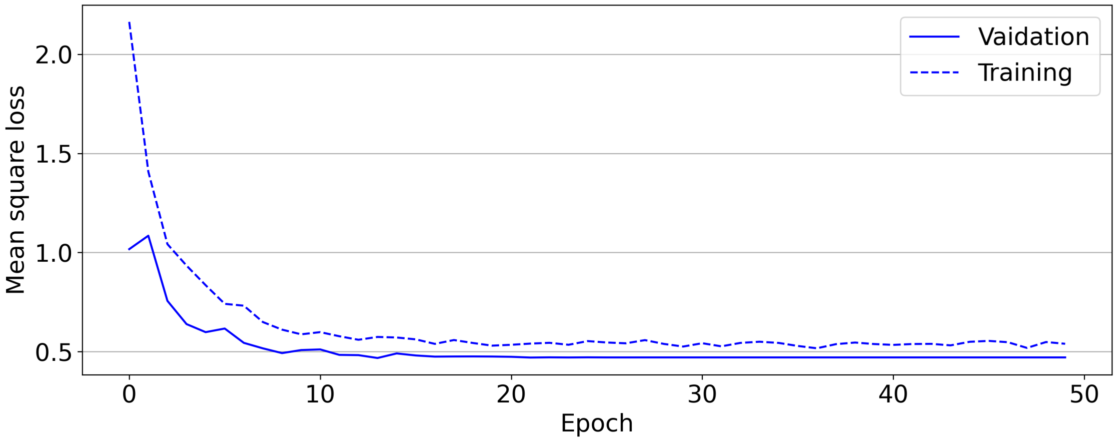
(a) RH-D

(b) RH-D

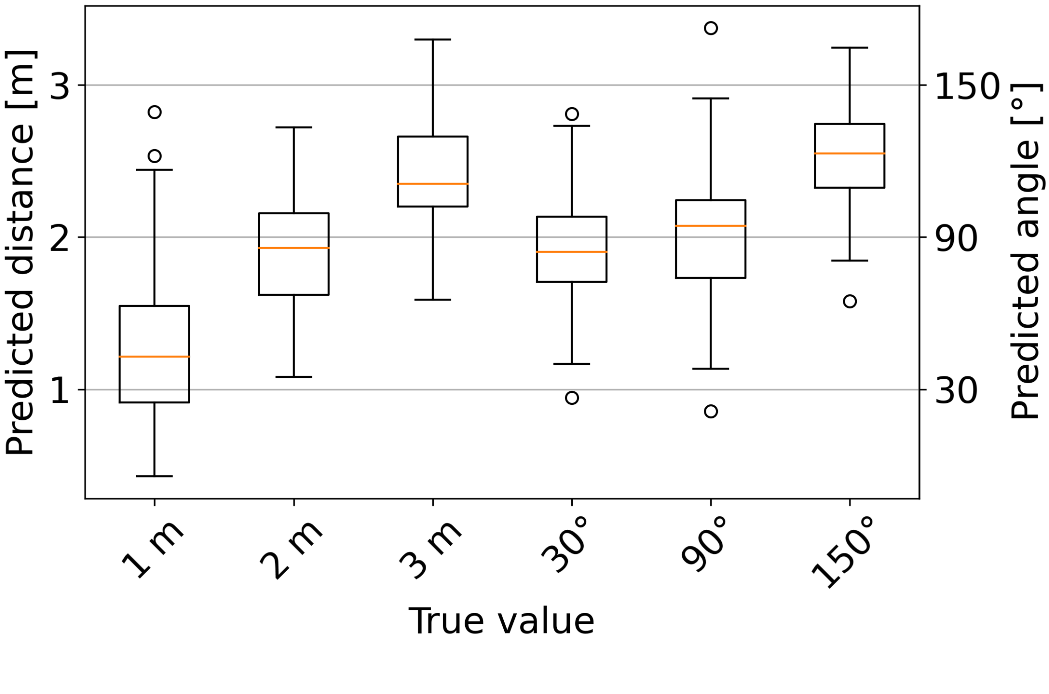
(c) RH-S

(d) RH-S

Figure B.1.: Single-parameter models trained with helical DOFS dataset

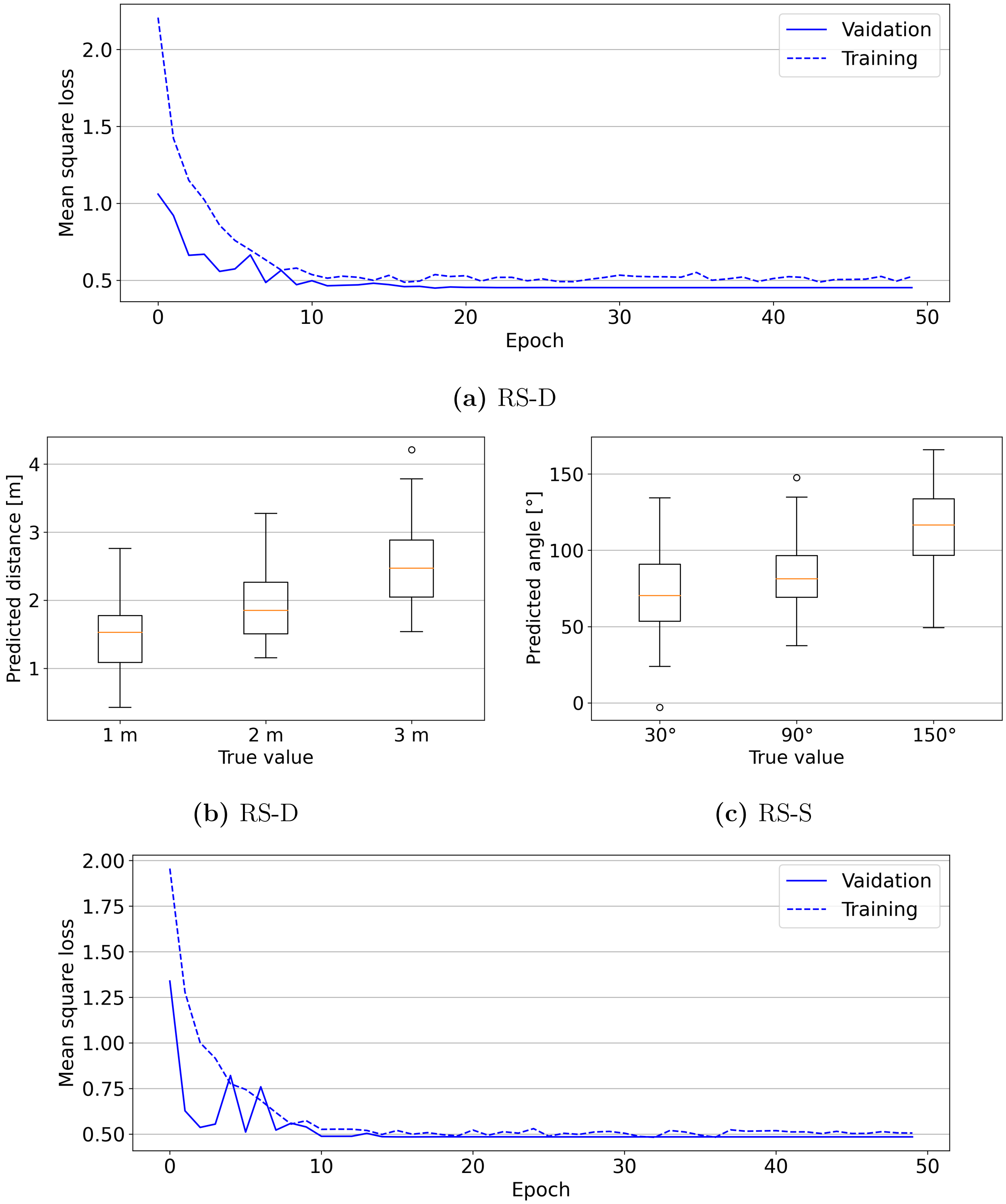


(a) Development plot



(b) Confusion matrix

Figure B.2.: RH-F



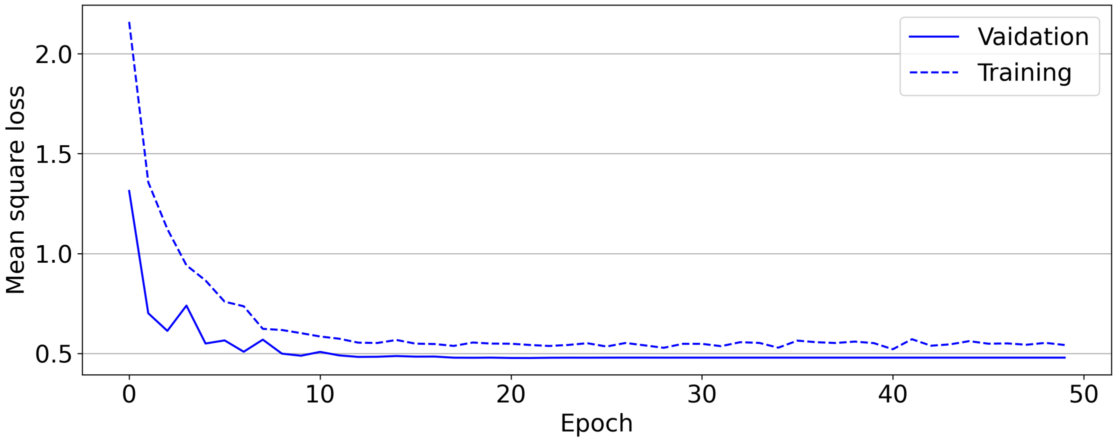
(a) RS-D

(b) RS-D

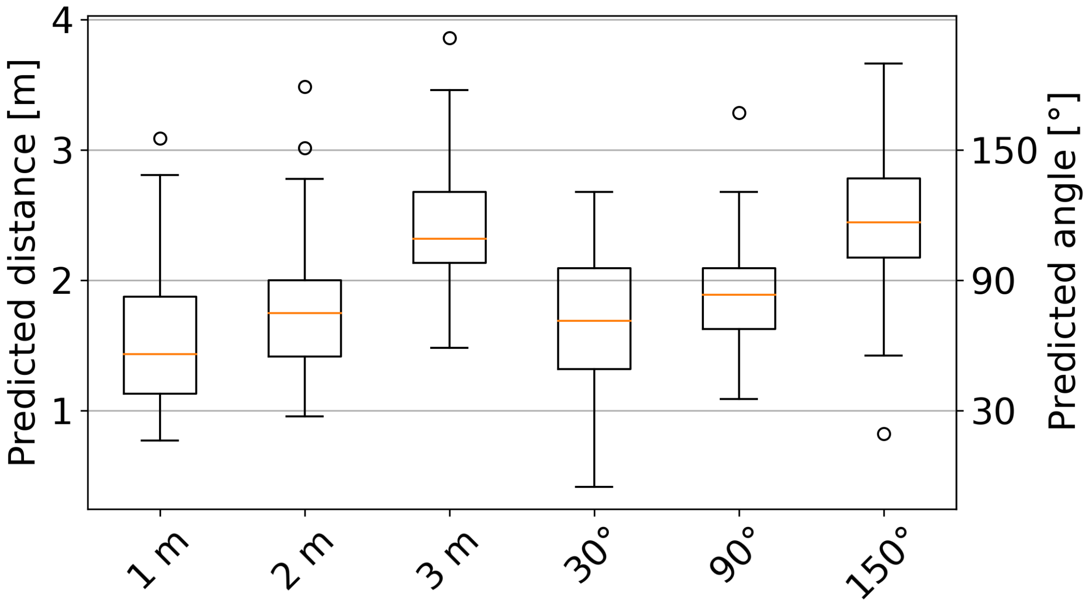
(c) RS-S

(d) RS-S

Figure B.3.: Single-parameter models trained with straight DOFS dataset



(a) Development plot



(b) Confusion matrix

Figure B.4.: RS-F

Appendix C.

Reference Measurement Comparisons

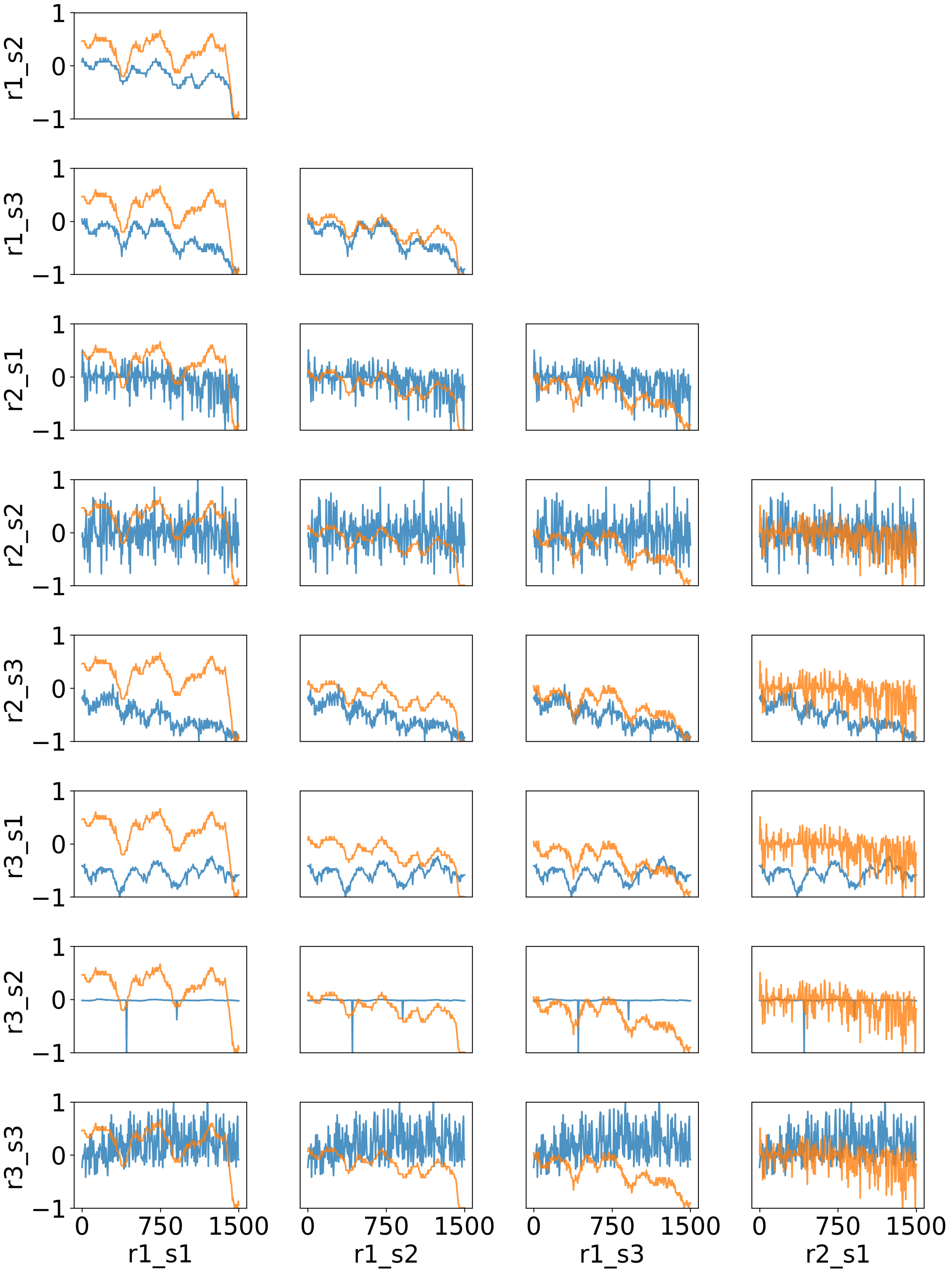
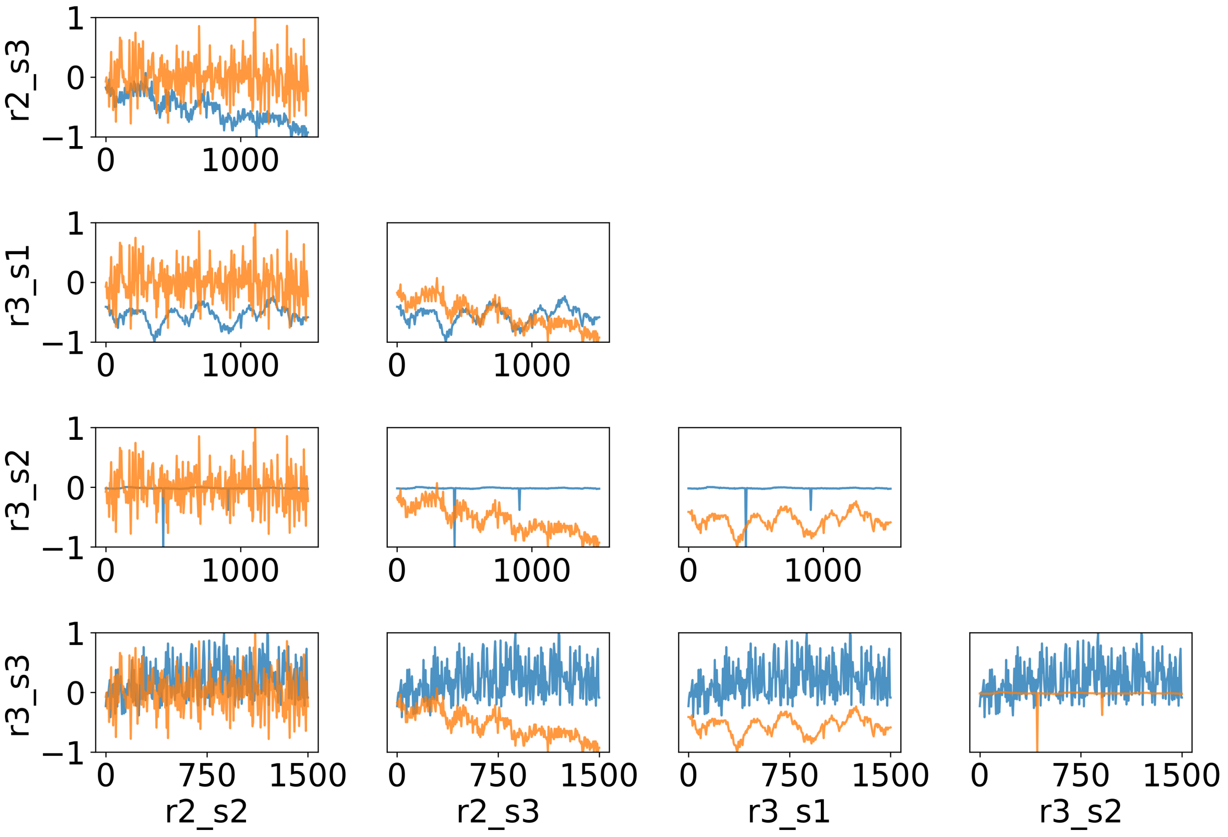


Figure C.1.: Helical DOFS references. For each reference, strains were extracted using the same measurement file, showing the effect of noisy references on the resulting strain values.



Helical DOFS references (cont.)

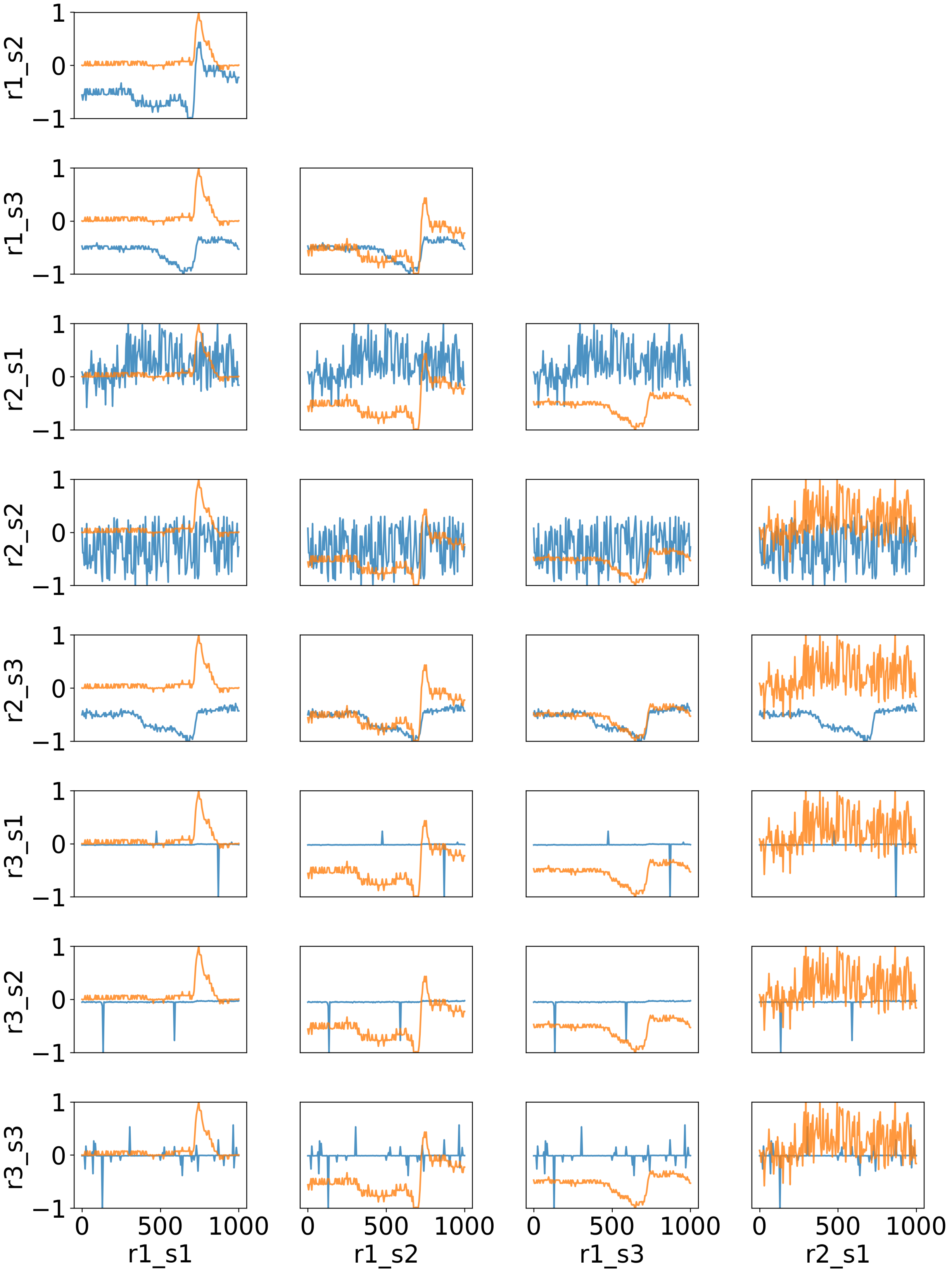
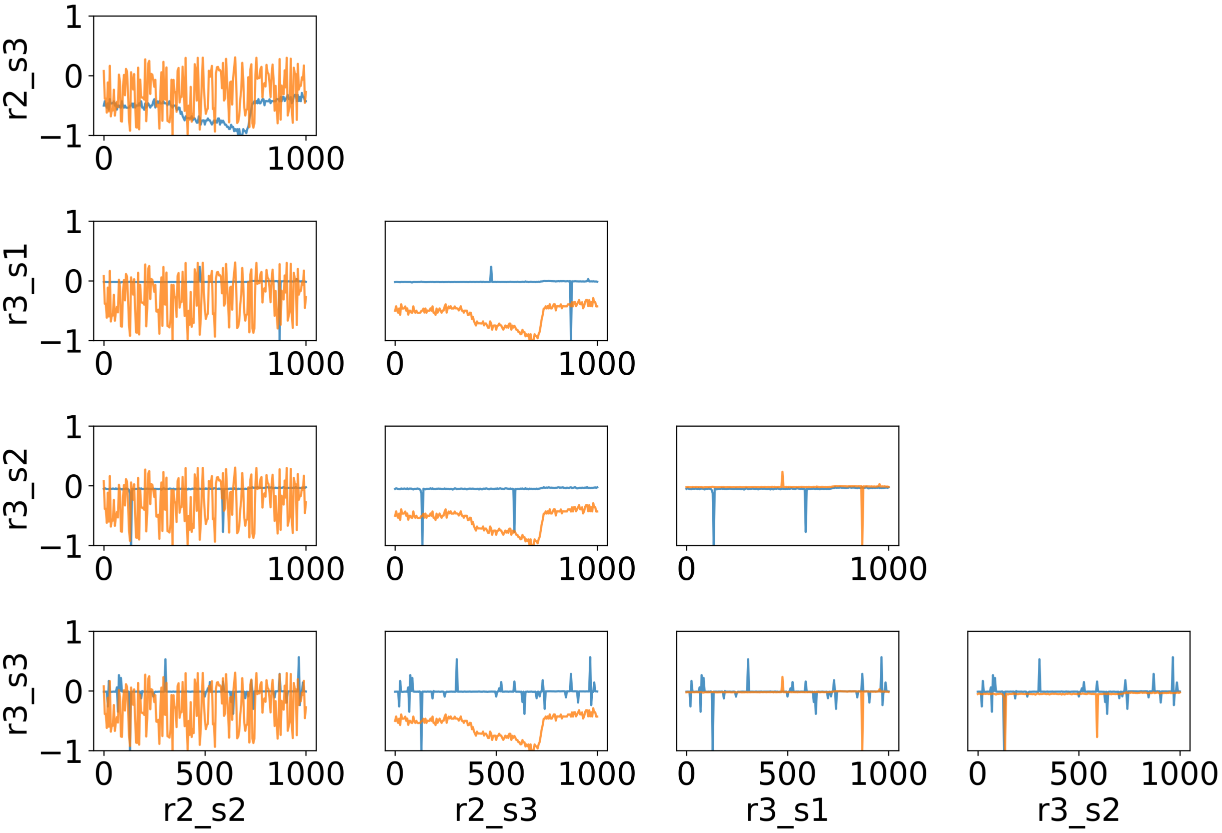


Figure C.2.: Straight DOFS references. For each reference, strains were extracted using the same measurement file, showing the effect of noisy references on the resulting strain values.



Straight DOFS references (cont.)

Appendix D.

Risk Assessment for Field Work

RISIKOKARTLEGGING

Enhet / Institutt:	**Konsekvensområder:**	**Gjenstand/ område:**
Institutt for Maskinteknikk og Produksjon (MTP)	**Menneske, materiell, ytre miljø og omdømme**	**Masterprosjekt - feltarbeid**

Ansvarlig linjeleder:	**Ansvarlig gjennomfører:**	**Eksisternde risikoreduserede tiltak**
Fornavn Etternavn	**Jens Buvik Bugge, Kaspar Lasn**	*Apparaturkort* *Vernebriller* *Online HMS-kurs* *Egen risikovurdering* *HMS Romomvisning*
Signatur:	**Signatur:**	
Dato for vurdering	**Deltakere på risikovurdering:**	
20.04.2023		
Dato for revisjon	**Deltakere:**	
	Jens Buvik Bugge, Kaspar Lasn	

Farer å vurdere	Aktuelt nr/ja
Støy:	nr
Støv:	nr
Klemfare:	nr
fall:	Ja
Kutt:	nr
Snubling:	ja
Deler med høy hastiget:	nr
Roterende deler:	nr
Vibrasjon:	nr
Brann:	nr

Farer å vurdere	Aktuelt nr/ja
Forbrenninger:	nr
Forgiftning:	nr
Stråling:	nr
Arbeid i høden:	nr
Høyt / lavt trykk:	nr
Høy /lav temperatur:	Ja
Elektrisk sjokk:	nr
Fallende gjenstander:	nr
Eksportkontroll - relevant?	
nr	

	Situasjon	Mulig uønsket hendelse	Tiltak	Vurdering av sannsynlighet (S)	Vurdering av konsekvens (K) (0-5)				Risikoverdi (S x X)			
Nr.				(1-5)	Menneske	Økonomi/ materiell	Ytre miljø	Omdømme	Menneske	Økonomi/ materiell	Ytre Miljø	Omdømme
1	*Håndtering av oppsett*	*Fall i sjøen*	*Flytevest, tørre tepper og skift på land, vite hvor redningsutstyr og nærmeste stige befinner seg (oppstartsbrief). Bruk av båtshake som tillater håndtering fra trygt område, samt holde arbeidsområdet ryddig for å unngå snubling. Minst en ekstra person i nærheten når oppsett settes ut og tas opp pga høyere risiko*	*2*	*3*			*1*	*6*	*NR*	*NR*	*2*
2	*Montering og demontering av oppsett*	*Kuttskader*	*Fingre unna, minimere bruk av strips for å redusere antall kutt. Førstehjelpskit tilgjengelig*	*2*	*1*			*1*	*2*	*NR*	*NR*	*2*
3	*Andre personer på brygge I*	*Forstyrre målinger*	*Merke område med skilt etc, nye målinger ved hendelse*	*3*		*1*			*NR*	*3*	*NR*	*NR*
4	Andre personer på brygge II	*Skade på optisk fiber*	*Fiber merkes, henges opp og beskyttes med skumrør på ekstra utsatte steder. Ekstra fiberlengde klar i reserve*	*2*		*1*			*NR*	*2*	*NR*	*NR*
5	Måleutstyr (laster, optisk switch og pc) står på kaia/nær vannkanten	*Utstyret faller i vannet*	*Laser og pc holdes i bil, med signalkabel ut til oppsett*	*1*		*4*		*1*	*NR*	*4*	*NR*	*1*
6	*Vann på utstyr*	*Kortslutning/vannskade*	*Alt utstyr som ikke er vanntett holdes i bil eller i tette plastkasser for å beskytte mot skøysprøyt. Reserver av spesielt utsatte deler*	*2*		*2*			*NR*	*4*	*NR*	*NR*
7	*Arbeid på oppsett, verktøy faller i sjøen*	*Tap av verktøy*	*Holde arbeid i vannkanten til et minimum. Sikre løst verktøy med tau til belte,evt til faste objekter. Ekstra sett av kritisk verktøy*	*2*		*1*			*NR*	*2*	*NR*	*NR*
8	*Kunstige lydkilder i sjøen*	*Forstyrrelse av marint liv*	*Svært lokal lyd og lavt volum*	*3*			*1*		*NR*	*NR*	*3*	*NR*

NTNU

Norwegian University of
Science and Technology