Doctoral theses at NTNU, 2022:163

Peihua Han

Data-driven Methods for Decision Support

in Smart Ship Operations

Doctoral thesis

NTNU
Norwegian University of Science and Technology
Thesis for the Degree of
Philosophiae Doctor
Faculty of Engineering
Department of Ocean Operations and Civil Engineering

NTNU
Norwegian University of
Science and Technology

Peihua Han

Data-driven Methods for Decision Support

in Smart Ship Operations

Thesis for the Degree of Philosophiae Doctor

Ålesund, May 2022

Norwegian University of Science and Technology
Faculty of Engineering
Department of Ocean Operations and Civil Engineering

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the Degree of Philosophiae Doctor

Faculty of Engineering
Department of Ocean Operations and Civil Engineering

ISBN 978-82-326-6033-9 (printed ver.)
ISBN 978-82-326-6301-9 (electronic ver.)
ISSN 1503-8181 (printed ver.)
ISSN 2703-8084 (online ver.)

IMT-report 2022:163

Doctoral theses at NTNU, 2022:163

Printed by NTNU Grafisk senter

Abstract

Vessels operating on the surface of the ocean today are now increasingly equipped with sensors. This includes GPS, MRU, IMU that monitor the vessel's motion behavior, and power, RPM, temperature sensors that monitor the status of components such as engines and thrusters, and anemometers that provide information about the surrounding environment. These sensor measurements are obtained in real-time and historical data is saved in cloud storage. The increased digital capabilities motivate the industry to increase the automation of the vessel by developing decision support systems, digital twin, or autonomous ships, which might potentially lead to safer and more efficient ship operations.

How to use the massive data on ships to gain better insight into ship operations has always been a key issue, and the data-driven approach is a promising solution. Data-driven methods, or machine learning methods, have been used broadly across a range of industries concerned with data-intensive issues. As the ship is gradually turned into a colossal sensor hub, the massive volume of data can be used with supervised learning to generate models to support efficient ship operations, or unsupervised learning to provide key insights about ships.

To provide information to the human ship operator or autonomous ship operating system, two aspects can be identified: (1) a better understanding of the current status, such as component's status, environmental conditions, or operating conditions. (2) a better forecast on what will happen if a specific action is taken, which can be referred to as what-if analysis. In such a context, many elements can be involved (localization, trajectory prediction, etc.). In this dissertation, two important applications are highlighted: fault diagnostics and prognostics of components, and the estimation of the sea state.

Fault diagnostics and prognostics aim to detect and isolate faults on components or systems, and then predict how the fault will progress and how long it will be until complete failure. Through these actions, recommendations for maintenance can be provided. In other words, an ideal maintenance schedule can be devised and failure can be eliminated. In this way, the vessel can operate safely and efficiently. The sea state information is of key importance for ship operations, such as motion control, pipeline laying, and path planning. Wave radar may be an ideal solution for obtaining information about surrounding waves, but most ships today do not equip with one. However, it is also possible to estimate the sea state from the vessel motion responses (especially motions that are not affected by the controller: roll, pitch, heave, etc.). Thus another focus of this dissertation is to develop sea state estimation models that use ship motion responses as inputs. Since both fault diagnostics and prognostics and sea state estimation can be achieved with machine learning, the main objective is to develop data-driven models for these two applications. Three case studies are conducted to validate the de-

veloped data-driven models for these two applications, where the first two concerns fault diagnostics and prognostics (use thruster and engine as an example, respectively), the last one concerns sea state estimation. Experiments are carried out with data collected in simulations, in the laboratory, and on the vessel RV Gunnerus operating in the real world. The results demonstrate the advantages of developing data-driven models to support ship operations. Additionally, data-driven models can outperform traditional models in certain scenarios.

Acknowledgment

The research presented in this thesis was conducted at the Norwegian University of Science and Technology in Ålesund within the Department of Ocean Operations and Civil Engineering (IHB). Financial support was provided by the Knowledge-Building Project for Industry "Digital Twins for Vessel Life Cycle Service" from the Research Council of Norway under Project 280703.

My supervisors throughout this Ph.D. project have been Prof. Houxiang Zhang, Prof. Guoyuan Li and Dr. Stian Skjong. The guidance and support I received during the last three years are highly appreciated. In particular, I would like to thank my main supervisor, Prof. Houxiang Zhang, for motivating me and shaping me into an independent researcher. Thanks for your constant advice when I am confused. Also, I would like to thank Prof. Guoyuan Li for your help and discussion on various aspects of my research. I would like to thank Dr. Stian Skjong, who provides valuable feedback and suggestions on my research.

I would like to thank my previous officemate Dr. André Listou Ellefsen for leading me into the field of fault diagnostics and prognostics. It is always a pleasure to work with you. Thanks for involving me in the development of software for DIPAI AS. I wish you make a successful business.

Thanks to Dr. Shuai Yuan, who shares the house with me and help me get used to life in Norway. Thanks to Tongtong Wang and Baiheng Wu, who pursue Ph.D. with me. Thanks for being my comrades and inviting me for dinner. Thanks to Dr. Xu Cheng for taking me fishing and bringing me the rice cooker from China.

A big thanks to all my colleagues at the Intelligent Systems Lab (formerly Mechatronics Lab) at NTNU Ålesund. Thanks to Dr. Thiago Gabriel Monteiro, Dr. Lars Ivar Hatledal, Dr. Robert Skulstad, Dr. Luman Zhao, Dr. Pierre Major, Finn Tore Holmeset, Chunlin Wang, Motoyasu Kanazawa, William Schmidt, Mingda Zhu, and Zizheng Liu.

Finally, I give special thanks to my family for their patience and support.

Contents

List of Publications

This thesis is based on research resulting in five journal papers and one conference paper. They are all enclosed in the appendix section. In the following list of publications, the papers are listed chronologically, but in the main body of the text a more thematic presentation order is prioritized over the chronological one.

I P. Han, G. Li, R. Skulstad, S. Skjong and H. Zhang, "A Deep Learning Approach to Detect and Isolate Thruster Failures for Dynamically Positioned Vessels Using Motion Data", *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2020.

II P. Han, G. Li, S. Skjong, B. Wu and H. Zhang, "Data-driven Sea State Estimation for Vessels Using Multi-Domain Features from Motion Responses", *In 2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2120-2126, May 2021.

III P. Han, G. Li, X. Cheng, S. Skjong and H. Zhang, "An Uncertainty-aware Hybrid Approach for Sea State Estimation Using Ship Motion Responses", *IEEE Transactions on Industrial Informatics*, vol. 18, issue 2, pp. 819–900, 2021.

IV P. Han, A. L. Ellefsen, G. Li, F. T. Holmeset and H. Zhang, "Fault Detection With LSTM-Based Variational Autoencoder for Maritime Components", *IEEE Sensors Journal*, vol. 21, issue 19, pp. 21903–21912, 2021.

V P. Han, A. L. Ellefsen, G. Li, V. Æsøy and H. Zhang, "Fault Prognostics Using LSTM Networks: Application to Marine Diesel Engine", *IEEE Sensors Journal*, vol. 21, issue 22, pp. 25986–25994, 2021.

VI P. Han, G. Li, S. Skjong and H. Zhang, "Directional Wave Spectrum Estimation with Ship Motion Responses using Adversarial Networks", *Marine Structures*, vol. 83, p. 103159, 2022.

The following papers will not be discussed in this thesis. They may, however, be considered relevant due to co-authorship and similar topics:

i A. L. Ellefsen, P. Han, X. Cheng, F. T. Holmeset, V. Æsøy and H. Zhang, " Online Fault Detection in Autonomous Ferries: Using fault-type in-dependent spectral anomaly detection", in *IEEE Transactions on Instrumentation and Measurement*, vol. 69, issue 10, pp. 8216–8225, 2020.

ii X. Cheng, P. Han, G. Li, S. chen and H. Zhang, "A Novel Channel and Temporal-wise Attention in Convolutional Networks for Multivariate Time Series Classification", in *IEEE Access*, vol. 8, pp. 212247–212257, 2020. (equal contributions)

iii X. Cheng, G. Li, P. Han, S. Robert, S. chen and H. Zhang, "Data-Driven Modeling for Transferable Sea State Estimation Between Marine Systems", in *IEEE Transactions on Intelligent Transportation Systems*, 2021.

iv C. Wang, G. Li, P. Han, O. Osen and H. Zhang, "Impacts of COVID-19 on ship behaviours in port area: An AIS data-based pattern recognition approach", in *IEEE Transactions on Intelligent Transportation Systems*, accepted, 2022.

List of Abbreviations

TCOM	Technology Centre for Offshore and Marine
DSS	Decision Support System
ML	Machine Learning
CNN	Convolutional Neural Networks
LSTM	Long Short-Term Memory
AE	Autoencoder
VAE	Variational Autoencoder
GAN	Generative Adversarial Network
kNN	k Nearest Neighbor
SVR	Support Vector Regression
RBF	Radial Basis Function
GBDT	Gradient Boost Decision Tree
GP	Gaussian Process
RNN	Recurrent Neural Networks
DP	Dynamic Positioning
FDI	Failure Detection and Isolation
CONV	Convolutional Operation
BN	Batch Normalization
ReLu	Rectified Linear Unit
GAP	Global Average Pooling
FFT	Fast Fourier Transform
PSD	Power Spectral Density
LR	Logistic Regression
SVM	Support Vector Machine

RF	Random Forest
MLP	Multi-Layered Perceptron
FCN	Fully Convolutional Neural Network
PHM	Prognostics and Health Management
MSE	Mean Square Error
TTD	Time to Detect
DSF	Detection Stability Factor
MRN	Multi-Regime Normalization
RUL	Remaining Useful Life
PwL	Piece-wise Linear
FNN	Feed-forward Neural Network
RMSE	Root Mean Square Error
DOF	Degree of Freedom
mRMR	Minimum-Redundancy Maximum-Relevance
MAE	Mean Absolute Error
EN	ElasticNet Regularization
WBA	Wave Buoy Analogy

Nomenclature

ϵ, C	Two hyperparameters in the SVR
$\kappa(\cdot, \cdot)$	Kernel function
γ	A hyperparameters in the RBF kernel
$\Omega(\cdot)$	A regularization term for decision tree
$\mathcal{N}(\mu, \sigma^2)$	Normal distribution with mean μ and variance σ^2
$m(\cdot)$	Mean function in GP
I	Identity matrix
α, l	Two hyperparameters in the rational quadratic kernel
$q_\phi(z\|x)$	The encoder in VAE which is parametrized by ϕ
$p_\theta(x\|z)$	The decoder in VAE which is parametrized by θ
$p_\theta(z)$	The prior distribution of the latent variables in VAE
D_{KL}	The Kullback-Leibler divergence
h_t	Hidden state vector at time t in LSTM
c_t	Cell state vector at time t in LSTM
i_t, f_t, g_t, o_t	Tthe input, forget, cell and output gates at time t in LSTM
σ	Sigmoid function
d_{ft}	The fault time step in marine diesel engine
H_s	Significant wave height
D_m, θ_m	Mean wave direction
T_m	Mean wave period
T_p	Peak wave period
σ_s	Directional spreading parameter
$E(\omega, \theta)$	Directional wave spectrum, where ω is angular frequency and θ is the directional angle

ω_m	Model angular frequency of the double-peak wave spectrum
Γ	Gamma function
$S_{ij}(\omega)$	Cross-spectra value of motion i and j at frequency ω
$\Phi_i(\omega,\theta), \overline{\Phi_j(\omega,\theta)}$	The complex-value transfer function and its complex conjugate
$l(\cdot,\cdot), loss, \mathcal{L}$	Loss function
$\hat{p}_i, p_i$	Predicted and true probability
γ	Tunable parameter in the focal loss function
P_{macro}	Evaluation metric for multi-class classification: Macro-Precision
R_{macro}	Evaluation metric for multi-class classification: Macro-Recall
$F1_{macro}$	Evaluation metric for multi-class classification: Macro-F1
$f_{mRMR}(\cdot)$	Minimum-redundancy maximum-relevance feature selection function
$I(\cdot,\cdot)$	Mutual information
$P(y\|ML,WBA)$	Predictive distribution from the combination of ML and WBA method
$P(y\|ML), P(y\|WBA)$	Predictive distribution from ML and WBA method, respectively
μ_{ML}, σ^2_{ML}	Mean and variance from ML method
μ_{WBA}, σ^2_{WBA}	Mean and variance from WBA method
$\mathcal{L}_{mse}$	Mean square loss
$\mathcal{L}_{bce}$	Binary classification loss
λ	A hyperparameter to balance $\mathcal{L}_{mse}$ and $\mathcal{L}_{bce}$

List of Figures

List of Tables

1 Introduction

The maritime industry is undergoing a massive shift as vessels are being monitored due to the advance in sensor and digital technology. The vessels are transformed into sensor hubs, whose systems and components are generating data and connecting to the Internet. It opens up the opportunity to enhance the performance and operation of a vessel by enabling increased automation and decision support, which can be termed as smart ship operation. This dissertation mainly focuses on how to make use of data-driven modeling to provide decision support for vessels, hence enabling the ship to operate in a safer and more efficient manner.

1.1 Background and motivation

The systems and components of a vessel are now increasingly equipped with sensors and connected to the internet. The development of digital technology enables the industry to monitor ships in real-time, while accessing historical data in cloud storage. These advances motivate the industry, as well as academia, to develop decision support systems, digital twin [1], automation systems to realize smart ship operation, and even autonomous vessels [2]. DNV GL has launched a new chapter called digital features and a smart vessel notation to its world leading-rule for ship classification[1]. SINTEF Ocean and Technology Centre for Offshore and Marine (TCOM) in Singapore have developed a roadmap for smart and autonomous sea transport systems with an emphasis on smart and autonomous shipping[2]. In such a context, smart represents the digital capabilities to enhance vessel performance due to the continuous increase in automation of ship processes and decision support systems.

Fig. 1.1 presents an illustration of smart ship. The vessel can transfer the data to a remote control center through a satellite data link. The maritime performance and monitoring can be integrated into a decision support system for the entire life cycle of the vessel. A decision support system (DSS) is a computerised information system which contains domain-specific knowledge and analytical decision models to assist decision maker by providing information and various alternatives [3]. The decision support system for the smart ship can be installed onboard for the crew or placed in the remote control center. The system intends to allow the operators to better understand the vessels. For instance, the operating status of the components such as engine can be monitored and its fault development trend can be tracked to support optimal mainte-

[1]New DNV GL rules drive smart ship operation and management, https://www.dnv.com/news/new-dnv-gl-rules-drive-smart-ship-operation-and-management-188497, Date accessed 12-November-2021

[2]A roadmap for smart and autonomous sea transport systems, https://www.sintef.no/en/latest-news/2020/a-roadmap-for-smart-and-autonomous-sea-transport-systems, Date accessed 12-November-2021

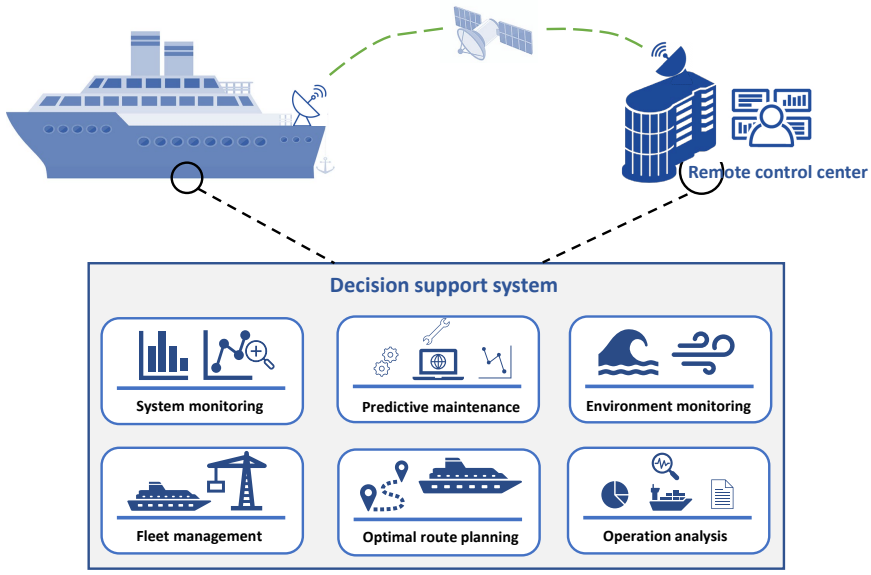


Figure 1.1: An illustration of smart ship and decision support system.

nance [4]. The environmental information such as wind, wave, and current can be used for route planning with low fuel consumption and carbon emissions. A DSS architecture involves three fundamental components: the database, the model, the user interface [5]. In this dissertation, only the model part is heavily discussed with the focus on using data-driven methods to develop models for better understanding of the vessel's status.

Data-driven method is becoming increasingly popular in recent years. It is an empirical modelling method and based on analysing the data about a system, in particular finding connections between the system's input and output variables without explicit knowledge of the physical behaviour of the system [6]. It focuses on computational intelligence and machine learning (ML). The term machine learning is often used interchangeably. For many applications, it can be far easier to train a system by showing the desired input-output behavior using ML rather than to program it manually [7]. ML has also been used broadly across a range of industries concerned with data-intensive issues. As the ship is gradually turned into a colossal sensor hub, the massive volume of data can be used with ML to provide critical insights and sophisticated models that enhance decision-making for safe and efficient ship operations.

1.2 Research questions

The focus of this dissertation is concerned with data-driven method for decision supports of vessels. This prompts the first question of this dissertation:

- **What are the fundamental principles to build a data-driven model?**

In order to answer this question it is necessary to investigate the principles behind

machine learning algorithm and how machine learning model is applied on other domains. ML models, especially for supervised learning, require a large amount of high-quality labeled data covering the region of interest. Since these kind of methods rely purely on data, there is a limit of what you can do with the data. The algorithms are the way to approach this limit. In some cases, a simple algorithm with good data outperforms an advanced algorithm with bad data. Furthermore, it is necessary to establish what data-driven methods can and cannot do, thus the following research question is raised:

- **What are the limitations come with data-driven methods?**

There are always assumptions involved in the ML models. One particular assumption for most ML algorithms is that the test data, which represents the performance when the model is deployed into the real world, should be sampled independently and identically from the same distribution as the training data [8]. This is not always the case in the maritime domain. Maritime operations usually involve highly complex and uncertain environments and the system might change over time, where the model trained with historical experience could fail.

Other limitations include that the learned model could be black-box and not explainable, the model can easily fit spurious correlation, and performance cannot be guaranteed at the "long-tail" data [9, 10, 11]. This leads to the next research question:

- **How to assess the accuracy and uncertainty of data-driven models?**

The ML model is usually evaluated in a train-test-spilt manner with a problem specific evaluation metric. The results from the test data provide a quantitative measurement of how this model will perform in the real world. However, as mentioned earlier, there might be out-of-distribution data when the model is deployed for maritime operations. Ideally, an ML model should be able to provide not only the predictions but also how much confidence it has in the predictions. There is extensive research in the literature on the predictive uncertainty of data-driven models [12, 13, 14].

Building upon the previous research, more practical research questions related to developing data-driven models for decision support of vessels emerged.

- **How might the data-driven method be used to provide decision supports for vessels?**

The decision support for a vessel usually involves two aspects: (1) a better understanding of the current status, such as component's status, environmental conditions, or operating conditions. (2) a better forecast on what will happen if a specific action is taken, which can be referred as what-if analysis. These two aspects are not necessarily separate since a better understanding of the current status might also lead to a better forecast. By exploiting the historical data of ship operation, it is possible to build a data-driven model for recognize the state of the ship and predict the future through pattern recognition, time series prediction, etc.

In terms of decision support, there are many elements that may be beneficial to ships, as shown in Fig. 1.1. This dissertation is limited to the fault diagnostics and prognostics of component and the estimation of wave information. This leads to the following two research questions:

- **How to perform fault diagnostics and prognostics on vessel's component such as engine using data-driven methods?**
- **How to estimate the sea state information when the vessel is in operation using data-driven methods?**

To address these two questions it is imperative to investigate how to perform maintenance and how to obtain wave information, in the vessel today. Maintenance operations on vessel today follow a reactive maintenance or preventive maintenance approach[3]. Reactive maintenance is post-failure repair while preventive maintenance involves predefined maintenance intervals. Sensor data is underutilized and no diagnostics or prognostics is involved. For sea state estimation, most vessels are not equipped with a wave radar, and thus no wave information is available. Although the above two have not yet been implemented, they constitute an important issue for decision support in smart ship operations. Recently, there have been some attempts to solve these problems with data-driven methods [15, 16], which is also the focus of this dissertation.

In recent years, machine learning has developed rapidly, especially in the field of deep learning. The deep learning methods are representation-learning methods can model very complex function without manual features, and has produced very promising results in many fields. It leads to the final research question:

- **Can recent advanced machine learning techniques benefit the model development for the above applications?**

1.3 Scope of work

1.3.1 Research objectives

In seeking to answer the above research questions, this dissertation seeks to obtain the following research objectives (ROs):

√ **RO1: Assess the status the internal components of the ship, the main goal is to develop fault diagnosis and prognostics models using data-driven method.**

The monitoring of the internal components of a ship is a key part to enable a smart ship. The components in a vessel today are usually inspected at static time intervals, which are purely based on the experience of the manufacturer or shipowner. The faults can only be be discovered when the scheduled inspection is performed. By incorporating fault diagnostics and prognostics, reliability can be improved and redundancy can be reduced. The development of fault detection and isolation model for thruster is covered in paper I, while the fault detection and prognostics model for engine is covered in paper IV and V. However, faulty data might not be easy to collect, which leads to the next research objective:

[3]Beyond condition monitoring in the maritime industry, https://www.dnv.com/Publications/beyond-condition-monitoring-in-the-maritime-industry-12403, Date accessed 12-November-2021

√ **RO2: Develop a fault detection method for maritime components based on semi-supervised learning.**

As mentioned previously, the lack of labeled fault data makes it difficult to use supervised learning for fault detection. Instead, a large amount of normal operation data can be collected easily. This necessitates the use of semi-supervised procedure. This approach builds a model for the classes that correspond to normal behavior by using only normal data, and then uses the model to identify anomalies in the test set. Paper IV covers the development and utilization of a semi-supervised approach for fault detection of diesel engine.

√ **RO3: Make the environmental conditions that the ship is currently operating in more accessible using data-driven method.**

Environmental conditions, especially sea waves, play an important role in the safe and efficient operation of ships. The motion responses of a ship reflect the sea state conditions, and therefore, a ship can be considered as a large wave buoy. From this perspective, a vessel is essentially equipped with an environmental condition estimation system. The goal is to create a data-driven model to estimate the sea state based on the ship motion responses. Papers II, III, and VI explores different data-driven models to build a wave estimation model using motion responses. In addition to the accuracy, the trust to the model and the generalization of the model to unseen data is also important since it is almost impossible to collect enough real-world data. This leads to the next research objective:

√ **RO4: Propose techniques to improve the reliability of the data-driven models in maritime operations.**

The data-driven model are usually black-box and cannot be generalized to unseen data. A certain degree of interpretability, e.g., how each feature contributes to the prediction, can provide posthoc validation to examine if the model actually learns some useful rules. It could be beneficial for the user to trust the model, which is of key importance in the maritime industry. Due to the complexity and uncertainty of maritime operations, the model may inevitably encounter out-of-distribution data, which must be carefully considered. The above two issues are discussed in papers II and III.

1.3.2 Interconnection between the research objectives

The interconnection between the research objectives and the papers published are shown in Fig. 1.2. In order to satisfy RO1, two case studies are presented, one of which takes the propeller as an example and the other takes the engine as an example. Paper I regards the problem of thruster fault isolation as a supervised classification problem, and uses convolutional neural networks (CNN) to classify faulty thrusters. For diesel engine, paper IV and V presents a fault detection model and fault prognostics model, respectively. Furthermore, paper IV proposes a fault detection model based on variational autoencoder (VAE) and long short-term memory (LSTM) for maritime components. The proposed model uses only normal data for training, as suggested by RO2.

For RO3, an ensemble model to estimate the wave parameters is proposed in paper II. The model is subsequently developed in paper III to include a model-based method to compensate for the unseen data. Papers II and III also tries to accommodate the issues of interpretability and out-of-distribution data as suggested by RO4. To obtain more detail information of the ocean wave, a model based on generative adversarial network (GAN) is proposed in paper VI to estimate the directional wave spectrum. All of the

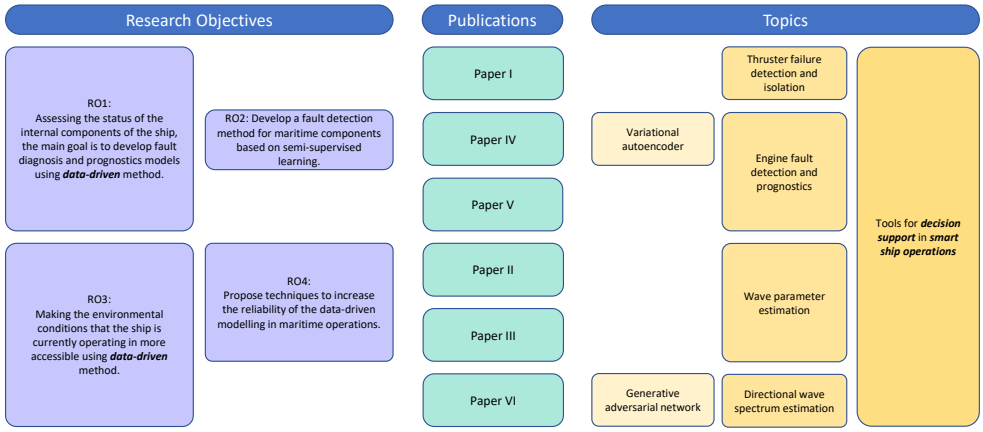


Figure 1.2: Interconnection of published paper in the thesis.

1.4 Contributions of the dissertation

The major contributions of this dissertation are as follows, which is related the research objective above:

- Present the fundamental framework to use data-driven methods for decision support to enhance vessel performance. The effectiveness of the methods is shown through three case studies. It is close related to RO1 and RO3.
- To deal with the difficulty to collect fault data, a data-driven fault detection method which uses only normal operation data for training is proposed. It is related to RO1 and RO2.
- Propose a hybrid method to compensate for the out-of-distribution predictions of data-driven method in sea state estimation. Additionally, method to estimate the detailed 2D directional wave spectrum is proposed. It is related to RO3 and RO4.

1.5 Structure of the dissertation

This introductory chapter presented the background for the dissertation research, establishing its main goals and defining the scope of work. The rest of this dissertation unfolds as follows. Chapter 2 introduces the foundation of applying data-driven method to support smart ship operation and the experimental platforms that are used to develop

and test the model. Chapter 3 presents the first case study, which focuses on thruster failure detection and isolation when the vessel is in dynamical positioning operation. This chapter is based on paper I. Chapter 4 relates to papers IV and V, and discuss fault detection as well as fault prognostics for maritime diesel engine. Chapter 5 presents sea state estimation using ship motion responses. This chapter is based on papers II, III, and VI. Chapter 6 concludes the dissertation, summarizes the contributions, and indicates the directions for future works.

2

Data-driven methods to support smart ship operations

This chapter introduces data-driven methods as a fundamental technology to support smart ship operations. Fig. 2.1 presents a schematic illustration of applying data-driven methods for decision support of vessel. The sensor data that monitor each part of the vessel are stored in a database. The data-driven approaches are then used to explore the data and build models from these data. The real-time data is fed into the model to provide decision supports such as sea state estimation, fuel consumption prediction, etc.

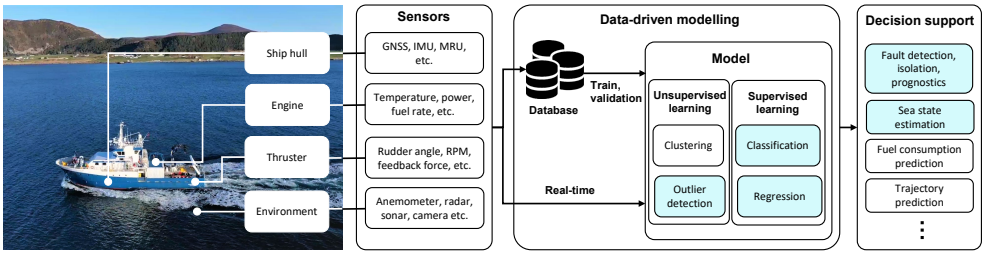


Figure 2.1: Schematic illustration of data-driven methods to support smart ship operation. The contents in blue background are the focus in this dissertation.

2.1 Fundamental of data-driven methods

Data-driven method is the general term for using data to build models to make accurate predictions. The term machine learning is often used interchangeably, which refers to the computational methods that learn through experience. Learning problem refers to improving a certain performance of a certain task through a certain type of training experience. The experience is expressed in the format of data and therefore the methods are data-driven.

Depending on whether labels (or corresponding target values) exist in the training set, the problem can be divided into two paradigms: supervised learning and unsupervised learning. Another paradigm called reinforcement learning [17], in which the learning algorithm is interacting with its environment to find the action that maximize a reward, will not be discussed in this dissertation.

2.1.1 Supervised learning

Supervised learning methods are the most widely used machine learning methods [18]. Supervised learning usually focus on function approximation problem. Given the training data set in the form of a collection of (x, y) pairs, the task is to find a function f so that a prediction $y^* = f(x^*)$ can be made to a query x^* based on the training data. The input x could be a vector or more complex objects, such as images and sensor signals. The form of the function f depends on the learning algorithms, and therefore many forms of f exists. Sometimes f is explicitly expressed as a parameterized function and the parameters are determined by the training data through a optimization process, while in other cases it is implicit and the form of f as well as its parameters are determined simultaneously by a search process with tunable hyperparameters. Despite the difference in the learning algorithms, supervised learning is usually used for the following tasks:

- **Classification:** this is the problem of assigning a label to the input x. For example, the fault isolation of marine diesel engines can be regarded as a classification problem, and labels such as air filter fault, turbo fault or bearing fault can be assigned according to the state of the engine. According to the type of label, there are binary classification (where y uses one of the two categorical values), multi-class classification (where y uses one of the K labels), and multi-label classification (where y is composed of several K labels are assigned at the same time) [7].
- **Regression:** this is the problem of predicting a value for input x, where the target y is a continuous value. For instance, fuel consumption prediction and trajectory prediction consists of predicting the value of fuel consumption and the location of the vessel, respectively.
- **Ranking:** the task is to provide a order on a set for input x. For instance, the recommendation system will return a list of preferred items in order for the user.

2.1.2 Unsupervised learning

Unsupervised learning generally involves the analysis of unlabeled data. Unlike supervised learning, unsupervised learning receives a dataset in the form of x and the pattern in that dataset is discovered based on some assumptions about the structural properties of the data. Common tasks in unsupervised learning usually involve:

- **Clustering:** clustering is the problem of finding a partition of data without any labels to indicate the possible partitions. The data in the same group is usually called a cluster. Examples of the use of clustering methods include the analysis of the ship's trajectory pattern [19] and the partition of the operating conditions of the diesel engines onboard [15].
- **Dimensional reduction or manifold learning:** the task is to transform an initial representation of objects into a lower-dimensional representation. The data can be assumed to lie on a low-dimensional manifold and the manifold is identified from data. A common example is to perform feature reduction for supervised learning.

- **Novelty and outlier detection:** both of these methods are used for anomaly detection, where the objective is to detect abnormal or unusual observations in the dataset. These methods can be used for data-driven fault detection.

2.2 Recent applications of machine learning for ship operations

This subsection will briefly introduce the state-of-the-art applications of machine learning for decision supports in ship operations.

- **Trajectory prediction:** Trajectory prediction is often formulated as an autoregressive time series forecasting problem, and time series forecasting models are often used. Long short-term memory (LSTM) networks are the de facto choice for many researchers. Tang et al. [20] used an LSTM that was trained on AIS data collected at a port to predict ship location 20 minutes later. Skulstad et al. [21] used an LSTM to predict the position of a vessel during a dynamic positioning operation when the GPS signal is lost..
- **Prediction of power or fuel consumption:** The power or fuel consumption prediction is usually formulated as a regression problem. Inputs to such models usually involve ship speed, draft, environmental conditions, etc. Models employed range from simple linear regression models to complex deep neural networks [22]. The models obtained from data-driven methods can then be easily used for route planning [23, 24].
- **Condition monitoring of machinery system:** Condition monitoring is close related to fault diagnostics and prognostics. Fault diagnostics is formulated as a classification or regression problem by taking the time series sensor measurement as input. Tan et al. [25] compared several multi-label classification algorithms to classify different fault types of the electric propulsion system. Xu et al. [26] fused three different machine learning models for wear diagnosis of marine diesel engine. Fault prognostics is about to predict the remaining useful life for predictive maintenance. To the best of my knowledge, it is not well investigated for vessels since run-to-failure data is hard to collect. Nonetheless, the problem can be formulated as a regression problem and artificial neural network [27] or recurrent neural network [28] can be used.
- **Ocean wave estimation and forecasting:** Waves are generated by complex interaction of wind with the ocean and is stochastic. Wave estimation is performed by using the time series ship motion data to estimate the current wave condition. Cheng et al. [29] divided the sea state conditions into several predefined categories and used neural network to perform the classification. Wave forecasting using machine learning is often referred to as time series forecasting. Neural networks, random forests, and recurrent neural networks are typical choice for such applications [30, 31].

From a broader view, the above applications aim to provide essential or extra information for the ship operators. The contributions of this dissertation mainly lie in the last two parts.

2.3 Machine learning models used in this dissertation

In the literature, a variety of machine learning algorithms have been developed to cover the various data and problem types that are manifested in different machine learning problems [32]. A machine learning model can be viewed as a search process in a large number of possible function spaces to find a function that optimizes performance metric through the training experience. Different machine learning models vary greatly, but they are mainly reflected in two aspects. One is the way they represent the function, e.g., mathematical function, decision tree, and neural network. The other is the optimization algorithms to search through the space of functions, e.g., gradient descent in neural networks, the spilt rule in decision tree. Here, only the machine learning models used in this dissertation to support smart ship operations are introduced.

2.3.1 k nearest neighbor

K nearest neighbor (kNN) is the most fundamental and simple supervised learning methods that has been used extensively in the machine learning literature. It was developed in 1951 in an unpublished report from the need to perform discriminant analysis when parametric estimates of probability densities are difficult to determine. Later it was formal introduced and some of the properties such as the error rates were presented [33]. Since then, a long line of research has been carried out, including distance weighted approaches [34], fuzzy learning methods [35], and soft computing methods [36]. Moreover, improvements by distance function, neighborhood size, and class probability estimation are widely investigated. Today, kNN is often used as a benchmark for more complex learning algorithm and sometimes it still fares better than many more powerful classifiers. A recent review of kNN can be found in [37].

kNN is a lazy learner that it stores the training data set $((x_1, y_1), ..., (x_n, y_n))$ and predicts a testing point based on a fixed number k of its closest neighbors in the feature space of the training data set. It is originally used for classification but it can be easily modified for regression. For a novel test point x, kNN regression computes the mean of the function values of its k-nearest neighbors. A distance-based version kNN is used here, where the k-nearest neighbors are weighted by the inverse of their distance:

$$f_{kNN}(x) = \frac{1}{k} \sum_{i \in N_k(x)} \frac{\frac{1}{d(x,x_i)}}{\sum_{i \in N_k(x)} \frac{1}{d(x,x_i)}} y_i \tag{2.1}$$

where set $N_k(x)$ containing the indices of the k-nearest neighbors of x. $d(x, x_i) = \sqrt{|x - x_i|^2}$ is Euclidean distance.

2.3.2 Support vector machine

Support vector machine (SVM) is first introduced by Boser, Guyon, and Vapnik in 1992 [38]. It is theoretically motivated from the statistically learning theory since the 1960 [39]. Support vector machine constructs hyperplanes to seperate data in a high dimensional space with maximum margin since it assumes that the maximum-margin hyperplanes are most robust. To create nonlinear classifier, kernel trick is applied. Later, soft-margin was proposed [40] and the soft-margin SVM is an example for empirical risk minimization. It soon became popular since its success in handwritten digit recogni-

tion. Extensions for multi-class classification [41], clustering [42], regression [43] were proposed. Today it is less popular but it still remains as a fundamental algorithm for kernel learning.

In this dissertation, the variant for regression that called support vector regression (SVR) is used. Due to the cost function that ignores any training data close to the model prediction, it depends only on a subset of the training data. The basic idea of SVR is to fit a function $f(x) = \langle w, x\rangle + b$ onto a training data set. The weights vector w, b can be obtained by solving the optimization problem:

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}||x||^2 + C\sum_{i=1}^{n}(\xi_i - \xi_i^*) \\ \text{s.t.} \quad & -\epsilon - \xi_i^* \leq \langle w, x_i\rangle + b - y_i \leq \epsilon + \xi_i \\ & \xi_i, \xi_i^* \geq 0 \end{aligned} \tag{2.2}$$

where ξ and ξ^* are slack variables representing the deviation from a predefined gap with hyperparameter ϵ. The hyperparameter C denotes the strength of the regularization which is inversely proportional to C. Solving this problem requires the application of the Lagrangian multiplier technique, which by itself leads to a dual optimization problem:

$$\begin{aligned} \min_{\alpha,\beta} \quad & \frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i - \beta_i)(\alpha_j - \beta_j)\kappa(x_i, x_j) \\ & + \epsilon\sum_{i=1}^{n}(\alpha_i + \beta_i) - \sum_{i=1}^{n} y_i(\alpha_i + \beta_i) \\ \text{s.t.} \quad & 0 \leq \alpha_i, \beta_i \leq C \end{aligned} \tag{2.3}$$

where $\kappa(x_i, x_j)$ is a kernel function which is used to account for nonlinearities. For the kernel function, radial basis function (RBF) kernel is used, which is expressed as $\kappa(x_i, x_j) = exp(-\gamma||x_i - x_j||^2)$. γ is a hyperparameter in the RBF kernel. The details of SVR and the solving process for its dual optimization problem can be refer to [44].

2.3.3 Gradient boost decision tree

Gradient boosting is a machine learning technique that generates a prediction model in the form of an ensemble of weak prediction model such as decision trees. The idea of gradient boosting originated from the boosting algorithm Adaboost [45]. Later it was formulated as gradient descent with a special loss function to increase the performance [46] and this regression technique is named as gradient boost machine. Apart from theoretical research, scalable system such as XGBoost [47] and LightGBM [48] were developed for ease of use. In recent years, it has gained a lot of attention as the algorithm of choice for many machine learning competition winning teams.

Gradient boost decision tree (GBDT) is an ensemble model using gradient boost technique with decision trees as base learners. The prediction of the GBDT is the sum of M trees:

$$\hat{y}_i = \sum_{t=1}^{M} f_t(x_i) \tag{2.4}$$

In t iteration, a tree model $f_t(\cdot)$ is generated by minimizing the following function:

$$\begin{aligned} Obj^{(t)} &= \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \\ &\simeq \sum_{i=1}^{n} [g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t) + constant \end{aligned} \tag{2.5}$$

where $l(\cdot, \cdot)$ denotes the loss function, mean square error is usually used for regression problem. $\Omega(\cdot)$ is a regularization term for decision tree. The objective function can be approximated by second-order Taylor expansion, where $g_i = \partial_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$ and $h_i = \partial^2_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$.

2.3.4 Gaussian process regression

A Gaussian Process (GP) is a probability distribution over functions and inference taking place directly in the space of functions. GP regression originated in geostatistics in 1967 and is known as kriging [49]. It was introduced to the machine learning community from the inspiration to construct GP from neural networks [12]. Since GP is a nonparametric and interpretable Bayesian model, it was soon applied to learn forward or inverse dynamics of robotic systems. Later modifications were mainly to improve its scalability and handle sparse data [50]. The advantage of GP is that it provides a well-calibrated uncertainty of the prediction.

We assume either exact or independent normally distributed measurement errors, i.e. the evaluation of $y(x)$ at point x satisfies:

$$y(x)|f(x) \sim \mathcal{N}(\mu(x), \sigma^2(x)) \tag{2.6}$$

where σ^2 is a known function describing the variance of the measurement errors and $\mu(x)$ is the mean.

GP is characterized by a mean function $m(x)$ and a covariance kernel function $\kappa(x, x')$. Given the training set at n points with input as $x_{1:n} \triangleq \{x_1, x_2, \cdots, x_n\}$ and target as $y_{1:n} \triangleq \{y_1, y_2, \cdots, y_n\}$, the posterior can be obtained by combining these observed values with prior:

$$\begin{aligned} \mu(x) &= m(x) + \kappa(x, x_{1:n})[\kappa(x_{1:n}, x_{1:n}) + \sigma_n^2 I]^{-1}(y_{1:n} - m(x_{1:n})) \\ \sigma^2(x) &= \kappa(x, x) - \kappa(x, x_{1:n})[\kappa(x_{1:n}, x_{1:n}) + \sigma_n^2 I]^{-1}\kappa(x_{1:n}, x) \end{aligned} \tag{2.7}$$

where σ_n^2 is a additive noise level. The $\mu(x)$ can be viewed as the prediction of the function value, while the σ^2 is a measure of uncertainty of the prediction. Usually a constant mean function $m(x) = 0$ is used and the rational quadratic kernel is used:

$$\kappa(x, x') = \left(1 + \frac{(x - x')^2}{2\alpha l^2}\right)^{-\alpha} \tag{2.8}$$

where α and l are parameters of the kernel. These parameters are obtained by maximizing the log marginal likelihood.

2.3.5 Artificial neural network

An artificial neural network (ANN) is based on a collection of connected units called artificial neurons. The neurons are connected and a neuron receives a signal then processes it. Neurons have weights that adjusts as learning proceeds. The first ANN was invented in 1958 called the perceptron [51]. Later in 1965 a network with many layers was published. The general method of automatic differentiation [52] was proposed in 1970s and was soon used for practical training of artificial neural networks by back-propagation. Only recently did GPUs make backpropagation for deep neural networks efficient, and it is now the foundation of deep learning [53]. Three types of ANN architectures are introduced as follows.

Variational autoencoder

The variational autoencoder (VAE) [54] is the artificial neural network architecture introduced in 2013 rooted in the variational inference methodology [55]. The model is a generative model that encode the data of interest into a low-dimensional latent distribution, which enables deep unsupervised representation learning. It has a strong impact on the machine learning community and has been applied widely to signal processing problems such as image and speech synthesis. There are many extensions, e.g., to model sequential data, modeling sequential data [56], forcing disentangle representations [57], and forcing deterministic constrained representations of data [58].

VAE shares similar architectural affinity with the autoencoder (AE). The VAE replaces an AE's latent representation z of given data x with stochastic variables, as shown in Fig. 2.2. The encoder $q_\phi(z|x)$ approximates the true posterior and the decoder $p_\theta(x|z)$ represents the likelihood of the complex process of data generation that results in the data x from z. The encoder and decoder are modeled in the structure of the neural network which is parametrized by ϕ and θ, respectively. The VAE optimizes the parameters, ϕ and θ, by maximizing the lower bound of the log-likelihood.

$$\begin{aligned} \mathcal{L}_{vae} &= -D_{KL}\big(q_\phi(z|x)||p_\theta(z)\big) + E_{q_\phi(z|x)}[\log p_\theta(x|z)] \\ &\leq \log p(x) \end{aligned} \tag{2.9}$$

where D_{KL} is the Kullback-Leibler (KL) divergence. Minimizing D_{KL} between the approximated posterior $q_\phi(z|x)$ and the prior $p_\theta(z)$ of the latent variable regularizes the latent space. The common choice of the prior distribution $p_\theta(z)$ is a standard Gaussian distribution $\mathcal{N}(0, 1)$ [54].

Convolutional neural network

A convolutional neural network (CNN) is a class of artificial neural network, based on the shared-weight architecture of the convolution kernels or filters that slide along input features and provide translation equivariant responses known as feature maps. The

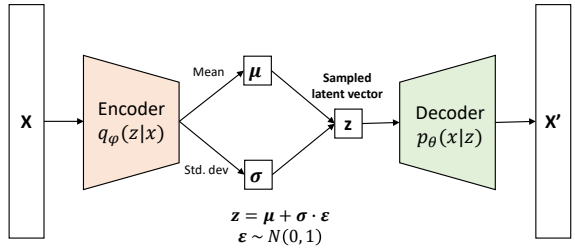


Figure 2.2: A simple illustration of a VAE.

first architecture of CNN was proposed in the 1980s and was called neocognition [59]. The two basic layers in CNN: the convolutional layer, and the downsampling layer were introduced. Inspired by the above work, the fist modern CNN was proposed in 1990s and back-propagation is used to learn the convolutional kernel weights [60]. Around 2012 a CNN called AlexNet [61] achieved state-of-the-art performance for image classification in the ImageNet challenge. Soon, CNNs rapidly gained popularity and found applications in image and video recognition, language processing, and time series modeling.

CNN is widely used for image when a 2D convolutional kernel is used. In this dissertation, the 1D convolutional kernel is considered since we focus on time series data. Suppose that f is a convolutional filter with kernel size s and T is a multi-variate time series with channel number m, the discrete 1D convolutional operation is defined as:

$$z[i] = \sum_{k=1}^{m} \sum_{j=1}^{s} f_k[j] * T_k[i + j - 1] + b \tag{2.10}$$

where i denotes the i_{th} element of result and b is bias. The convolutional filter with size s will move along the time axis with stride length r and repeat the operation as shown in (2.10). Depending on the filter, the convolution is capable of extracting insightful information from the original time series. The weight in the convolutional filter f and bias b can be learned through the back-propagation algorithm.

Long short-term memory

Long short-term memory (LSTM) is a type of recurrent neural networks (RNN), which is also a class of artificial neural networks. The RNN was brought up in 1986 to use the internal state to process variable length of inputs for sequential modelling. In 1997, LSTM was introduced to accommodate long-term gradient vanishing or exploding in RNNs when trained with back-propagation [62]. Then LSTM started to outperform traditional models in certain speech applications. Today, LSTM has been widely applied in language and speech modelling such as machine translation and soeecg recognition.

The LSTM introduces a memory cell that regulates the information flow in and out of the cell. As shown in Fig. 2.3, the memory cell consists of three non-linear gating units that protect and regulate the cell state. The introduction of these gating units enable easy information flow along the entire chain, therefore, the gradient vanish problem can

–

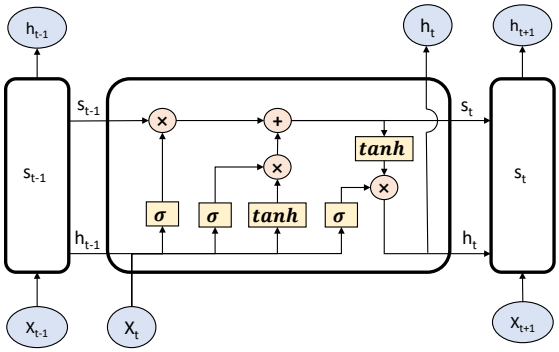


Figure 2.3: Schematic illustration of a LSTM cell.

be eliminated and it is able to learn long term dependencies. For each element in the input sequence, the LSTM computes the following function:

$$
\begin{aligned}
i_t &= \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}) \\
f_t &= \sigma(W_{if}x_i + b_{if} + W_{hf}h_{t-1} + b_{hf}) \\
g_t &= \tanh(W_{ig}x_i + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \\
o_t &= \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \\
c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\
h_t &= o_t \odot \tanh(c_t)
\end{aligned}
\tag{2.11}
$$

Where h_t is the hidden state at time t, c_t is the cell state at time t, x_t is the input at time t, h_{t-1} and c_{t-1} is the hidden state and cell state at time $t-1$, respectively. i_t, f_t, g_t, o_t are the input, forget, cell and output gates, respectively. σ is the sigmoid function, where $\sigma(x) = 1/(1+e^{-x})$. $\odot$ is the Hadamard product. W and b are the weights and bias in the LSTM cell.

2.4 Experimental platforms and data collection

This dissertation focus on two aspects for decision supports of vessels: fault diagnostics and prognostics, and sea state estimation. For fault diagnostics and prognostics, two components are used as case studies: thruster and engine. In order to verify the effectiveness of the data-driven approach, both simulation and real-world data were used. This section introduces the experimental platform and data collection procedure.

2.4.1 Thruster failure data

The thruster failure data is collected from OSC simulator[1]. The simulator features a simulated environment in which a user may manipulate the wind, waves, and ocean current to mimic environmental conditions. A multi-purpose offshore vessel is selected. This offshore vessel is equipped with 4 tunnel thrusters and 2 main thrusters as presented

[1]OSC AS, https://osc.no/, Date accessed 3-January-2022

in Fig. 2.4. Three different typical sea states are simulated as shown in Table 2.1. The wind, wave, and current will come from the same direction. The direction of the environmental disturbances (α in Fig. 2.4) is incremented at an interval of 60 degrees from 0 to 360 degrees, relative to the vessel frame.

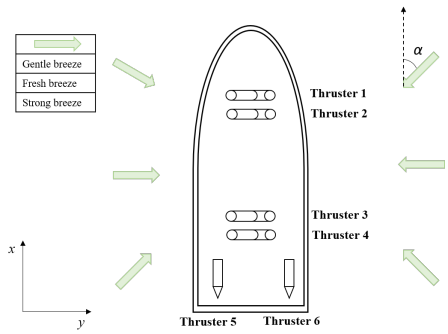


Figure 2.4: Thruster configuration.

Table 2.1: Descriptions of sea states.

Beaufort scale	Wind velocity (m/s)	Wave height (m)	Current velocity (m/s)
Gentle breeze	4	1	0.2
Fresh breeze	8	2	0.2
Strong breeze	12	3	0.2

The dynamic positioning (DP) operation is simulated and the desired position is set to (0, 0). Thrusters are randomly disabled in various environmental conditions to simulate thruster failure. The resulting dataset is shown in Fig. 2.5. 'Normal' denotes no thruster failures and 'Thruster 1' represents failure in thruster 1. In total, around 43 hours was simulated whereof 58% without thruster failure. The dataset is relatively unbalanced. Three control signals including the surge, sway, and yaw forces together with the 6 degrees of freedom motion data of the vessel were extracted. The data was extracted at a sampling rate of $10\,Hz$.

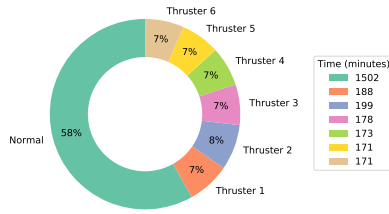


Figure 2.5: Thruster failure distribution of the dataset.

In order to provide a more realistic scenario, noise was added to the measured position and velocity. The position measurements are subjected to a combination of

white noise, a bias and a Gauss-Markov process as presented in [63]. The angular and linear velocity received only a constant bias and white noise [64].

2.4.2 Engine data

Fault data collection

The data for engine fault detection is collected from a diesel engine operated on Norwegian University of Science and Technology's research vessel Gunnerus. Gunnerus is equipped with the latest technology for a variety of research activities within biology, technology, geology, archaeology, oceanography and fisheries research. The diesel electric system of Gunnerus is used to generate electric power which is supplied to the power grid for operating the vessel. The data from an entire month of November 2019 are collected. During these periods, the vessel has been sent out for several purposes such as sea trial, maneuvering courses, etc. No specific fault for the engine was found in this period. The time interval when the vessel is in operation is filtered out. A total of 10 days of vessel operating time were obtained, averaging approximately 6 hours per day. Table 2.2 lists the sensor measurement related to the diesel engine from the logging system. The sensor data is collected at a sampling rate of 1 Hz.

Table 2.2: Descriptions of 9 sensors included in the logging system.

Index	Sensor	Unit
1	Boost Pressure	bar
2	Engine Speed	RPM
3	Engine Exhaust Gas Temperature 1	°C
4	Engine Exhaust Gas Temperature 2	°C
5	Fuel Rate	liter/min
6	Lube Oil Pressure	bar
7	Lube oil Temperature	°C
8	Engine Power	kW
9	Cooling Water Temperature	°C

On 21th, November 2019, a fault on this diesel engine was introduced. The air filter clogging fault was simulated using a cloth winding tape, as shown in Fig. 2.6. The left subgraph in Fig. 2.6 shows the diesel engine onboard and the right subgraph presents that the air filter is clogged with the tape. The outer surface of the air filter of the diesel engine is wrapped with a cloth winding tape. In this way, the heat dissipation and exhaust capacity of the air filter are reduced. Note that the fault introduced can be categorized as an abrupt fault.

The 10 days of collected data is divided into the following three parts: (1) a test set containing 2 days of data: the day when the fault was introduced and a random normal operation day; (2) a validation set containing data for 1 of the remaining 8 days; (3) a training set containing the data for the remaining 7 days. The training set and validation set are used in the training phase of the model. The test set is used to show the efficacy of the model.

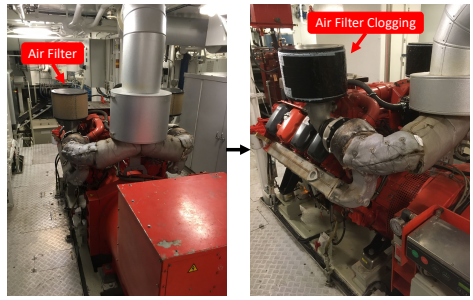


Figure 2.6: Diesel engine operated in the NTNU's research vessel. The air filter is manually clogged for a period of time.

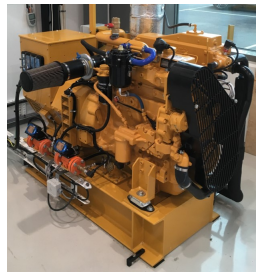


Figure 2.7: The marine diesel engine included in the hybrid power lab.

Run-to-failure data collection

A hybrid power lab, which is designed to research ship autonomy, is used to collect the run-to-failure data sets for fault prognostics. The lab is established by the Department of Ocean Operations and Civil Engineering at the Norwegian University of Science and Technology in Aalesund. The lab includes a marine diesel engine, a marine battery system, and a marine automation system to control the process. To simulate load alterations in the system, the produced power is supplied back to the power grid. The diesel engine is shown in Fig. 2.7.

Two engine load profiles have been used during the data collection process. The profiles aim to replicate two different environmental conditions the autonomous ferry may encounter during a ferry crossing on the west coast of Norway. First, the engine runs in normal condition. Then, two typical and independent fault-types associated with the marines diesel engine were provoked gradually in both profiles to obtain degradation data. The first fault-type is clogging of the air filter, while the second fault-type is a malfunction of the turbocharger. The data collection process was terminated when the engine reached operational failure. That is, a time after the fault-types were 100% provoked and the engine loses its operational ability.

Profile 2 is subjected to different engine loads, and hence, it will be used as the test set. However, the degradation in the test set should end sometime before failure in order to verify that the model is able to predict the RUL. Accordingly, a random

Table 2.3: The four original run-to-failure data sets collected from the marine diesel engine.

Data set	Profile	Usage	d_{ft}	Last RUL target	Time steps
Air filter	1	Train/val	1,660	0	2,346
Turbo	1	Train/val	1,347	0	2,346
Air filter	2	Test	1,483	106	2,240
Turbo	2	Test	1,347	490	1,856

interval of time steps is removed in both the air filter fault and the turbo fault in profile 2. Table 2.3 summarizes the data sets used. All data sets include 47 input features, e.g., engine load, engine speed, flow, pressure, and temperature measurements.

2.4.3 Wave estimation using ship motion

Data collection for wave parameters estimation

The ship motion data is collected through log files created by a data acquisition system onboard the RV Gunnerus. For all measurements in the data set, a sampling rate of 1 Hz was observed. To ensure the ship is in a nearly stationary condition, the maneuvering data that the vessel is cruising with a constant speed and constant heading is obtained. The cruising speed of the vessel is about 10 knots. Three sensor measurements related to the vessel motion were obtained: *sway velocity*, *roll*, *pitch*, and *heave*. These measurements are responsible for estimating the sea state. Two additional variables *longitude* and *latitude* are obtained, which is for matching the target sea state into the motion responses.

The sea state information is collected from the weather forecast system provided by the Norwegian Meteorological Institute. Since the vessel is only operating in the west coastal region of Norway, the coastal data is used. The coastal wave data is obtained by a numerical wave model which is run on an 800-meter grid with ECMWF and AROME atmospheric force. Three sea state characteristics are considered: Significant wave height H_s, mean wave direction D_m and mean wave period T_m.

Fig. 2.8 shows the contour plot of the significant wave height in the coastal region of middle Norway on a specific day. The two sea state characteristics are then matched to the ship motion data through position information. Specifically, the *longitude* and *latitude* corresponded to the ship motion data are used to query the nearest sea state information. The process is done by utilizing a ball tree with the Haversine distance.

Simulation data for wave spectrum estimation

For wave spectrum estimation, the wave spectrum-ship motion pairs are essential. These pairs are generated from simulation. In the simulations, a double-peak wave spectrum [65] is adopted since it covers a wide range of possible spectrum shapes and it models both the wind waves and the swell waves. The directional wave spectrum is given by:

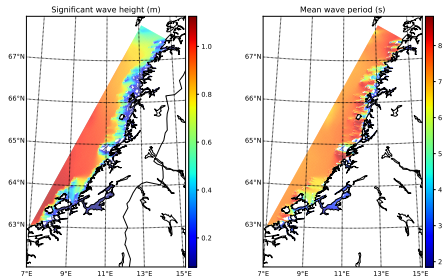


Figure 2.8: Sea State information in the middle Norway at 12:00, 13th, June, 2018 reported by the Norwegian Meteorological Institute.

$$E_g(\omega,\theta) = \frac{1}{4}\sum_{i=1}^{2} \frac{\left(((4\lambda_i+1)/4)\omega_{m,i}^4\right)^{\lambda_i}}{\Gamma(\lambda_i)} \frac{H_{s,i}^2}{\omega^{4\lambda_i+1}} A(s_i) \times \cos^{2s_i}\left(\frac{\theta-\theta_{m,i}}{2}\right) \exp\left[-\frac{4\lambda_i+1}{4}\left(\frac{\omega_{m,i}}{\omega}\right)^4\right] \tag{2.12}$$

where H_s is the significant wave height, θ_m is the mean wave direction and ω_m is the model angular frequency. s and λ are two shape parameters. Γ demotes the Gamma function. The function $A(s)$ is defined as:

$$A(s) = \frac{2^{2s-1}\Gamma^2(s+1)}{\pi\Gamma(2s+1)} \tag{2.13}$$

Note that the above wave spectrum model $E_g(\omega,\theta)$ is only used to generate the simulation data for this study and will not be used in our estimation network model. NTNU's research vessel R/V Gunnerus is used as the example vessel. The complex-valued response amplitude operators (RAOs) of the vessel are obtained from ShipX [66]. The ship motion cross-spectra is then calculated as:

$$S_{ij}(\omega) = \int_{-\pi}^{\pi} \Phi_i(\omega,\theta)\overline{\Phi_j(\omega,\theta)}E_g(\omega,\theta)d\theta \tag{2.14}$$

where $\Phi(\omega,\theta)$ is the complex-value transfer function and $\overline{\Phi(\omega,\theta)}$ is its complex conjugate.

Three corresponding ship motions, *sway velocity*, *pitch*, *heave*, are used. This results in 9 power spectra (6 real part and 3 imaginary part). The used wave spectrum consists of 10 parameters $[H_{s,1}, \omega_{m,1}, \theta_{m,1}, s_1, \lambda_1, H_{s,2}, \omega_{m,2}, \theta_{m,2}, s_2, \lambda_2]$. These parameters are sampled randomly to generate 1000 different wave spectrum and its corresponding ship motion.

2.5 Chapter summary

This chapter introduces the fundamentals of data-driven methods. The machine learning models and data sources that are used in this dissertation are also briefly introduced. It is worth noting that the data collected in these three case studies are time-series data and nearly stationary: 1) the ship motion data in dynamical position operation

and moving straight with constant speed is stable unless the wave conditions change suddenly; 2) the measurement from the engine is also stable and only depends on the operating conditions. The data also comes from the same data acquisition system and therefore the nature of data is quite similar.

Table. 2.4 shows how models and data sources are contained in different chapters. Chapter 3 presents thruster failure detection and isolation, which uses data in Section 2.3.1. Chapter 4 illustrate engine fault detection and prognostics, covering the data in Sections 2.3.2. Chapter 5 demonstrates case studies on sea state estimation, the data in Sections 2.3.3 are used. These three case studies form the main part of this dissertation, with the main goal to implement data-driven approaches to support vessel operations.

Table 2.4: Interconnection of the model and data source contained in different Chapters.

Data \ Model	kNN	SVR	GBDT	GP	CNN	VAE	LSTM
Section 2.3.1					Chapter 3		
Section 2.3.2						Chapter 4	
Section 2.3.3	Chapter 5						

3 Case study: Thruster failure detection and isolation

This chapter presents research results from paper I. Thrusters are the main propulsion units used to position a modern vessel. To mitigate the effects of thruster failures in dynamic positioning (DP) operation, vessels today with DP classes 2 and 3 [67] have been equipped with redundant thrusters. Once a thruster failure is correctly located, a warning can be sent to a crewmember or a high-level controller and the over-actuated vessel can still maintain its position or perform certain tasks if proper reallocation of the desired thrust is initiated.

If the thruster failure causes insufficient driving power of the vessels to compensate environmental effects, drift-offs will occur. In such a case, ship motion is correlated to the thruster failure event. Therefore the motion data can be used inversely to pinpoint the failed thruster.

In this chapter, the thruster failure detection and isolation (FDI) is treated as a time series classification problem. This chapter is not concerned with the failure modes in a thruster (such as failed gears or seals) but rather intends to locate which thruster has failed in a broader range. A convolutional neural network (CNN) is developed for thruster FDI of a dynamically positioned offshore vessel. The control signals and logged ship motions are used as input to the network. The output is the estimated thruster condition of the vessel.

3.1 Methodology

Fig. 3.1 shows a detailed diagram of the proposed method. The model is trained with historical data which also contains scenarios where thrusters are failing. Then it is deployed and provides predictions as new data comes in. Finally, the fault detection and isolation results are confirmed by feeding the network's predictions into a fault detection module.

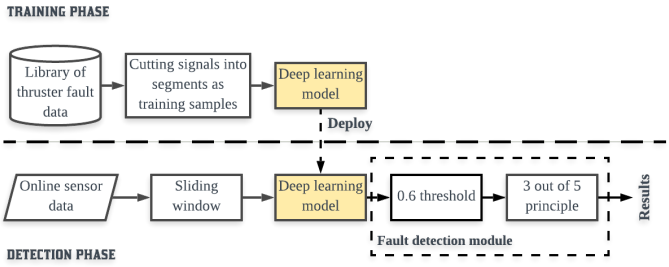


Figure 3.1: Block diagram of the proposed method for thruster FDI.

Unlike the offline training step, a fault detection module is introduced on the online detection step. A fault is first isolated when its corresponding probability exceeds 0.6. Then a 3/5 principle [68] is applied. As five new data points come in, the sliding window will move forward five times and provide five successive predictions. Only when more than three predictions indicate the same faulty thruster can we confirm the fault identification. Otherwise, it will be considered to be in normal condition. The fault predictor is expected to ensure robustness and eliminate accidental errors.

3.1.1 Network architecture

The detail network architecture is shown in Fig. 3.2. The sensor data is used as input to the model while the conditions of the thrusters are given as outputs. The convolutional layer includes three operations:

$$\begin{aligned} \mathbf{s} &= Conv(\mathbf{x}) \\ \mathbf{s} &= BN(\mathbf{s}) \\ \mathbf{s} &= ReLU(\mathbf{s}) \end{aligned} \tag{3.1}$$

where $\mathbf{x}$ is the input; $Conv$ represents the convolutional operation and it contains the learnable weight; BN denotes the batch normalization [69] layer which helps to accelerate the training process; $ReLu$ [70] is the activation function.

Max pooling layers are used after the first two convolutional layers. Global average pooling (GAP) is applied after the final convolutional layer instead of a fully connected layer, which largely reduces the number of weights. Finally, a softmax layer is employed tc
k

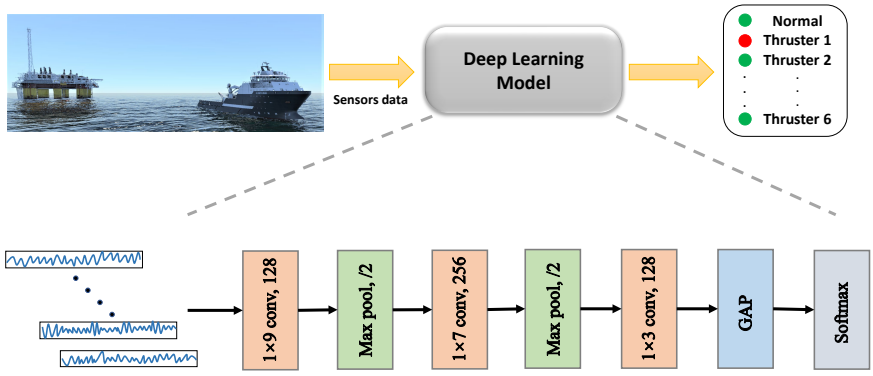


Figure 3.2: The proposed network architecture for thruster FDI.

3.1.2 Focal loss function

To train the network, the focal loss function [11] is used here instead of the cross-entropy loss function to address the problem of the imbalanced dataset. The focal loss function can be expressed as follows:

$$loss = \sum_{i=1}^{n} -p_i(1-\hat{p}_i)^{\gamma}\log(\hat{p}_i) \tag{3.2}$$

where $\hat{p}_i$ is the predicted probability value for class i and p_i is the true probability for that class, n denotes the class number, and γ is a tunable parameter. When $\gamma = 0$, this function degrades to the cross-entropy loss function. The term $(1-\hat{p}_i)^{\gamma}$ diminishes the loss assigned to well-classified examples and increases the loss for mis-classified examples. By modulating the cross-entropy loss towards the hard examples, the focal loss is addressed for class imbalance.

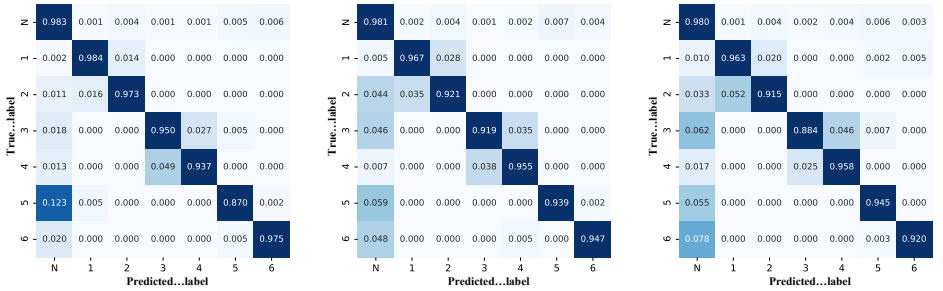
3.2 Experimental results

The collected data is split 70%-30% for training and testing. Each input measurement in the training set is normalized with z-score normalization and the corresponding normalization statistics are applied to the test set.

Since this problem is transformed into a multi-class classification problem, three widely used metrics for multi-class classification are used in this chapter to qualify the performance of the model: Macro-Precision (P_{macro}), Macro-Recall (R_{macro}), and Macro-F1 ($F1_{macro}$).

3.2.1 Evaluation on different environmental conditions

Fig. 3.3 shows the normalized confusion matrix under different sea state levels, where N denotes the label 'normal' and 1 is the label 'thruster 1 failure', etc. It is apparent that it is easy to confuse thruster 1 and 2 but they have an extremely small probability of being classified as other thruster failures. The phenomenon is much more obvious for thruster 3 and 4. The reason might be that they are located close to each other and thus they provide similar functionality for the vessel.



(a) (b) (c)

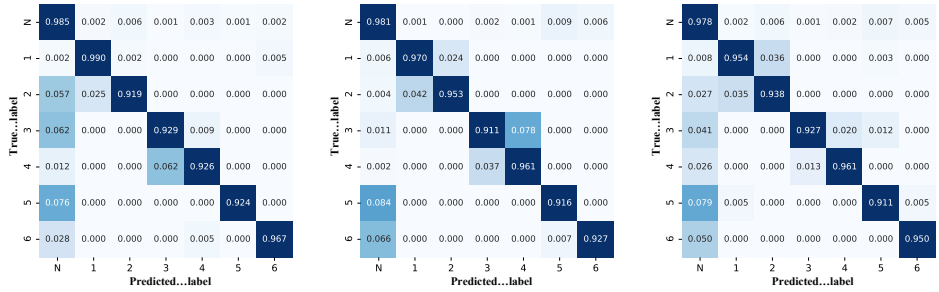
Figure 3.3: Confusion matrix under different sea states: (a) Gentle breeze, (b) Fresh breeze, and (c) Strong breeze.

In Fig. 2.4, the six directions of environmental disturbances are further merged into three groups since they are axis-symmetric. Direction 1 is angle of 30° and 330°, direction 2 is angle of 90° and 270°, and direction 3 is 150° and 210°. Fig. 3.4 shows the confusion matrix for the different directions and Table 3.2 summarizes the results. For direction

Table 3.1: Evaluation of thruster FDI on different sea states.

Window size	P_{macro}	R_{macro}	$F1_{macro}$
Gentle breeze	0.96	0.95	0.96
Fresh breeze	0.95	0.95	0.95
Strong breeze	0.95	0.94	0.94

2, thrusters 5 and 6 have a high probability of being mis-classified as normal, but the other thrusters do not. For directions 1 and 3, thrusters 1, 2, 3, and 4 have a higher probability of being mis-classified as normal than for direction 2. The reason might be that tunnel thrusters play more significant parts when environmental disturbances come from direction 2.



(a) (b) (c)

Figure 3.4: Confusion matrix under different direction of environmental disturbances: (a) Direction 1, (b) Direction 2, and (c) Direction 3.

Table 3.2: Evaluation of thruster FDI on different direction of environmental disturbances.

Window size	P_{macro}	R_{macro}	$F1_{macro}$
Direction 1: $\alpha = 30/330°$	0.96	0.95	0.95
Direction 2: $\alpha = 90/270°$	0.95	0.94	0.94
Direction 3: $\alpha = 150/210°$	0.95	0.95	0.95

3.2.2 Baseline comparison

Two different types of methods are compared here, namely the feature-based method and the end-to-end method (deep learning model). The feature-based method requires data pre-processing while the end-to-end method can work with raw sensor data directly.

For the feature-based method, features are extracted manually from the data. For each sensor, six time domain features and eight frequency features from fast Fourier transform (FFT) and power spectral density (PSD) are extracted. This results in 126 features in total. Down-sampling is performed to reduce the data with label "normal" to almost the same amount as the faulty data. Three different algorithms are used to train a classifier: Logistic regression (LR), Support vector machine (SVM), and Random forest (RF). For the end-to-end method, three models are compared: Multilayer

perceptron (MLP), Long short term memory (LSTM), and Fully convolutional neural network (FCN) [71].

Table 3.3 presents the performance for different methods. End-to-end methods outperform the feature-based method. Even though it is expected to achieve better results in feature-based methods with a more sophisticated feature extraction process or down-sampling approach, it will be highly time-consuming and laborious. The advantage of end-to-end methods is that they can achieve competitive results without a careful feature extraction process. Moreover, LSTM, FCN and the proposed neural network clearly outperform MLP in this case. The proposed network has a slight advantage over FCN and LSTM. The comparison also shows that the focal loss can improve the performance of the network.

Table 3.3: Comparison of different methods in terms of thruster FDI.

Types	Methods	P_{macro}	R_{macro}	$F1_{macro}$
Feature-based	LR	0.32	0.43	0.35
	SVM	0.56	0.68	0.60
	RF	0.71	0.83	0.75
End-to-end	MLP	0.70	0.81	0.72
	LSTM	0.91	0.91	0.91
	FCN [71]	0.93	0.92	0.93
	Proposed NN (w/o focal loss)	0.93	0.92	0.92
	Proposed NN (focal loss)	**0.95**	**0.94**	**0.95**

3.2.3 Online detection

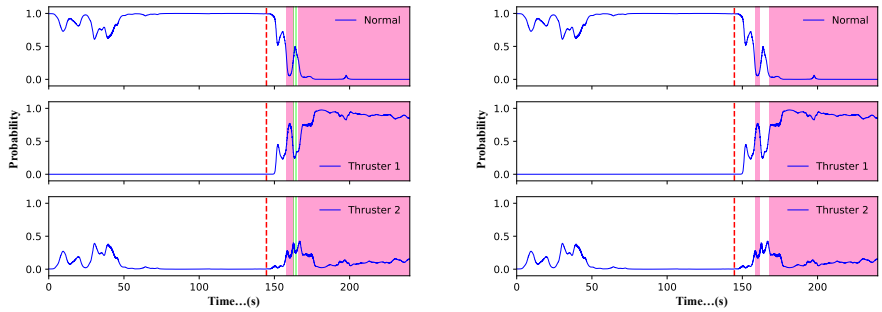
Fig. 3.5, Fig. 3.6 and Fig. 3.7 present the detection and isolation of thruster fault under gentle, fresh and strong breezes, respectively in the online detection setting. The red dashed line indicates the ground truth of the moment when thruster 1 has failed. The background in red represents that the thruster 1 failure is detected while the background in green denotes that the thruster 2 failure is detected. Without the predictor, the result is simply the highest probability of the predicted class. Only the probabilities of normal condition, the thruster 1 failure and the thruster 2 failure are presented since the probabilities of thrusters 3, 4, 5, 6 failure are relatively low in these cases. Table 3.4 summarizes the detection time and corresponding delays for the three cases. The fault can be detected under fresh and strong breezes but has a 9s delay in the gentle breeze. It should be noted that it does not represent the accuracy but a possible prediction delay under different environmental conditions.

Table 3.4: Summary of the online prediction cases.

	Ground truth time (s)	Failure detection time (s)	Detection delay (s)
Gentle breeze	144.7	153.6	8.9
Fresh breeze	144.9	145.5	0.6
Strong breeze	159.2	160.2	1.0

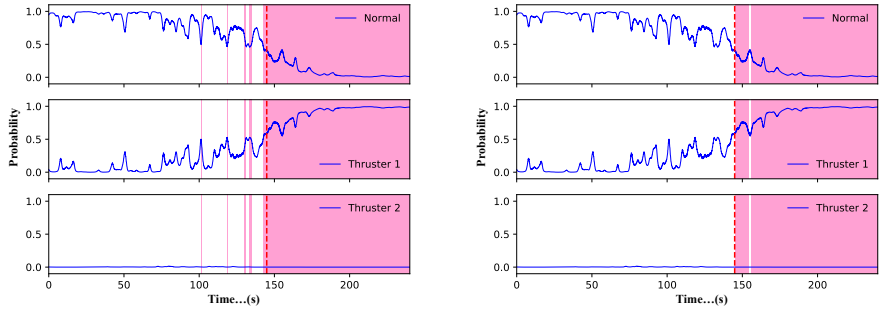
3.3 Chapter summary

This chapter uses thruster failure detection and isolation as an example. A deep CNN is proposed to detect and isolate potential thruster failures for DP vessels based on the



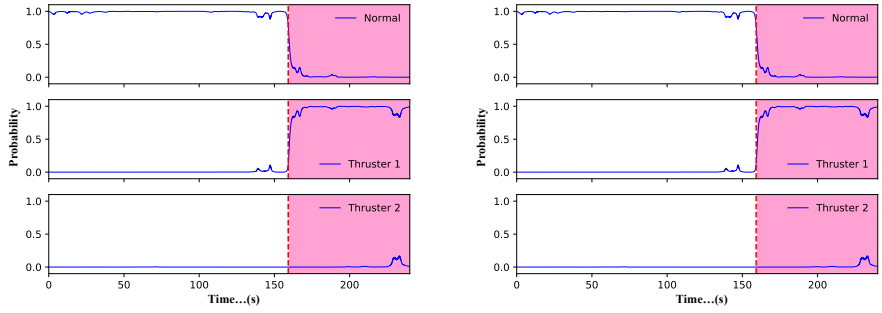
(a) (b)

Figure 3.5: FDI under gentle breeze: (a) w/o fault predictor, and (b) w fault predictor.



(a) (b)

Figure 3.6: FDI under fresh breeze: (a) w/o fault predictor, and (b) w fault predictor.



(a) (b)

Figure 3.7: FDI under strong breeze: (a) w/o fault predictor, and (b) w fault predictor.

control signals and corresponded ship motion. The problem is transformed into a time series classification problem and the model is trained with historical data set that contains

normal and fault data. Results from the simulation cases show that the proposed model is able to distinguish thruster failure under various environmental conditions with up to 95% accuracy. An online detection scheme is also presented to improve the robustness.

Findings suggest that when sufficient normal and fault data are available, the data-driven approach provides good performance for detecting and isolating thruster failure in DP ships without any vessel-dependent model.

4 Case study: Engine fault detection and prognostics

This chapter presents research results from papers IV and V. Maintenance is the key to ensuring the safe and efficient operation of marine vessels. Currently, reactive maintenance and preventive maintenance are two main approaches used onboard [72]. These approaches are either cost-intensive or labor-intensive. Recently, attention has shifted to prognostics and health management (PHM), which has the greatest promise for managing maintenance operations to achieve zero-downtime performance [73]. PHM systems aim to perform fault detection, fault isolation, fault identification, and remaining useful life prediction using available sensor measurements. In this way, an ideal maintenance schedule can be developed by continuously monitoring the status of the components and the evolution of their failures, which will considerably enhance operational availability and reliability as well as system safety. This chapter uses the diesel engine as an example, with the focus on developing fault detection and prognostics models for PHM.

4.1 Fault detection using LSTM-VAE

Fault detection or anomaly detection is the fundamental part of any PHM system. It focuses on identifying when the current execution differs from typical successful experiences. Model-based and data-driven methods are two paradigms depending on whether a physical model is used. In the data-driven methods, the semi-supervised anomaly detection method uses only normal data for training [74]. Therefore, it is widely applicable for maritime components since recording anomalous data is costly or even dangerous in comparison to normal data [75].

In this section, the LSTM-VAE for anomaly detection for maritime components is introduced. The structure of our proposed LSTM-VAE is different from the seq2seq model in [76] but similar to [77]. This structure allows us to consider long term dependencies of time series data and perform online predictions naturally.

4.1.1 Long-short term memory based variational autoencoder

The schematic illustration of the LSTM-VAE is shown in Fig. 4.1. LSTM-VAE is a combination of VAE and LSTM. Specifically, the LSTM as in eq.(2.11) is used to model the encoder $q_\phi(z|x)$ and decoder $p_\theta(x|z)$ in eq.(2.9).

The VAE assumes that data streams are i.i.d. in time. To introduce temporal dependency for this model, we replace the feed-forward network in a VAE to LSTM. Fig. 4.1 shows the LSTM-VAE structure that is unrolled in time. Given a multivariate input x_t at time t, the encoder LSTM output the hidden state h_t utilizing x_t, h_{t-1}, c_{t-1}. Then h_t is feed into two linear modules to estimate the mean μ_t and log-variance $\log \sigma_z$ of the posterior $p(z_t|x_t)$. A random sample z_t from $p(z_t|x_t)$ feeds into the decoder LSTM

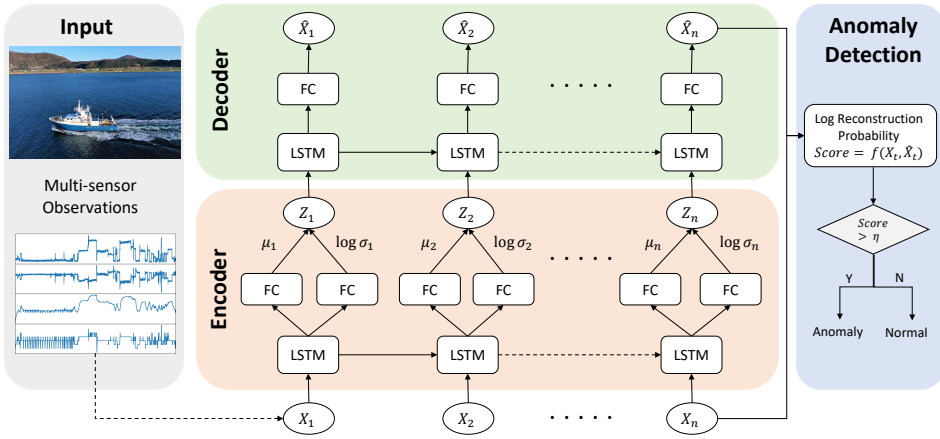


Figure 4.1: Illustration of the LSTM-VAE anomaly detector unrolled in time. Note that FC is fully connected layer. The FC in encoder have Relu activation while FC in decoder have identity activation. LSTM uses tanh activation. η is the fault detection threshold.

and then a final linear module outputs the reconstructed input $\hat{x}_t$. The parameters ϕ for encoder and θ for decoder can be obtained by minimizing the loss function as follows:

$$Loss = \sum_{t=1}^{T} [D_{KL}\big(q_\phi(z_t|x_t)||p_\theta(z_t)\big) + MSE(x_t, \hat{x}_t)] \tag{4.1}$$

where MSE denotes mean square error, T is the length of the sequences. A standard normal distribution $\mathcal{N}(0, 1)$ is used as the prior $p_\theta(z_t)$ of the latent space. Note that eq.(4.1) is only variation of eq.(2.9) since a multivariate Gaussian distribution can be assumed for continuous data and therefore maximizing the log-likelihood in eq.(2.9) equals minimizing the MSE in eq.(4.1).

4.1.2 Online anomaly detection with reconstruction probability

In autoencoders, reconstruction error is usually used as the anomaly score. Since VAE is stochastic in nature, the variability of the latent space can be taken into account. We use the reconstruction probability as the anomaly score for the proposed LSTM-VAE. The reconstruction probability is the Monte Carlo estimate of the log-likelihood $E_{q_\phi(z|x)}[\log p_\theta(x|z)]$ in (2.9), which can be calculated by a number of samples drawn from the latent variable distribution. Therefore the variability of the latent variable space can be taken into account, which extends its expressive power since normal data and anomaly data might share the same mean value but have different variability [78].

However, the Monte Carlo estimate requires sampling from the latent space and then forward the samples to the decoder to calculate the reconstruction probability. We implemented it in a different way by making use of a batch prediction, i.e., replicate the input by the number of samples and then perform the forward pass through the whole network. Algorithm 1 shows the pseudo-code for the online detection process

using reconstruction probability.

Algorithm 1 Online anomaly detection algorithm in terms of reconstruction probability

Input: $x_t \in R^D$, s_{t-1}, n
Output: $p_\theta(x_t|\hat{x}_t)$, s_t
$\phi, \theta \leftarrow$ *load the trained LSTM-VAE model, the* ϕ, θ *is obtained using the loss function in eq.*(4.1)
$x_t \leftarrow$ *get current multi-sensor data*
$s_{t-1} \leftarrow$ *get the state of LSTM from previous time step*
$x_t \leftarrow Normalize(x_t)$
$\mathbf{x_t} \leftarrow Batch(x_t, n)$
$\hat{\mathbf{x}}_\mathbf{t}, s_t \leftarrow f_{\phi,\theta}(\mathbf{x}_t, s_{t-1})$, refer to eq.(2.11)
$\mu, \sigma \leftarrow Statistics(\hat{\mathbf{x}}_\mathbf{t})$
$p_\theta(x|\hat{x}) = p(x|\mu, \sigma)$
return $\log p_\theta(x|\hat{x})$, s_t

4.1.3 Experimental results

Two performance evaluation metrics, *time to detect* and *detection stability factor* [79], are used in this paper to evaluate the performance of proposed method. *Time to detect* (TTD) is defined as the period of time from the beginning of a fault injection to the moment of the first detection signal occurs. *Detection stability factor* (DSF) is the level of stability of the detection signal measured as a percentage of the sum of duration of fault detection signals to the total time elapsed after fault injection.

To evaluate the performance of the proposed method, we implemented 6 baseline methods: (1) iForest: An isolation forest based detector with standard normalization. (2) iForest (MRN): An isolation forest based detector with multi-regime normalization (MRN). (3) AE: An autoencoder based detector with standard normalization. (4) AE (MRN): An autoencoder based detector with multi-regime normalization. (5) VAE: A variational autoencoder based detector with standard normalization. (6) VAE (MRN): A variational autoencoder based detector with multi-regime normalization.

Qualitative results

Fig. 4.2 shows the qualitative comparison of our proposed LSTM-VAE with the baselines method. The left subgraphs are from the day where a fault is introduced while the right subgraphs are from the normal operation test day. The period where the air filter clogging fault is introduced is marked with red background.

From the left first three subgraphs, the fault can be only detected when the multi-regime normalization is used. The right four subgraphs show the anomaly score for one normal operation day. The scores are therefore relatively low. The LSTM-VAE applies directly to the standard normalized data, which makes the method easy to scale to a complex system. Generally, lots of sensors are equipped in a maritime system and complex operation conditions are involved. Performing multi-regime normalization is unrealistic in most scenarios. From the left fourth and fifth subgraphs in Fig. 4.2, it is shown that the reconstruction probability provides a more noticeable change than reconstruction error.

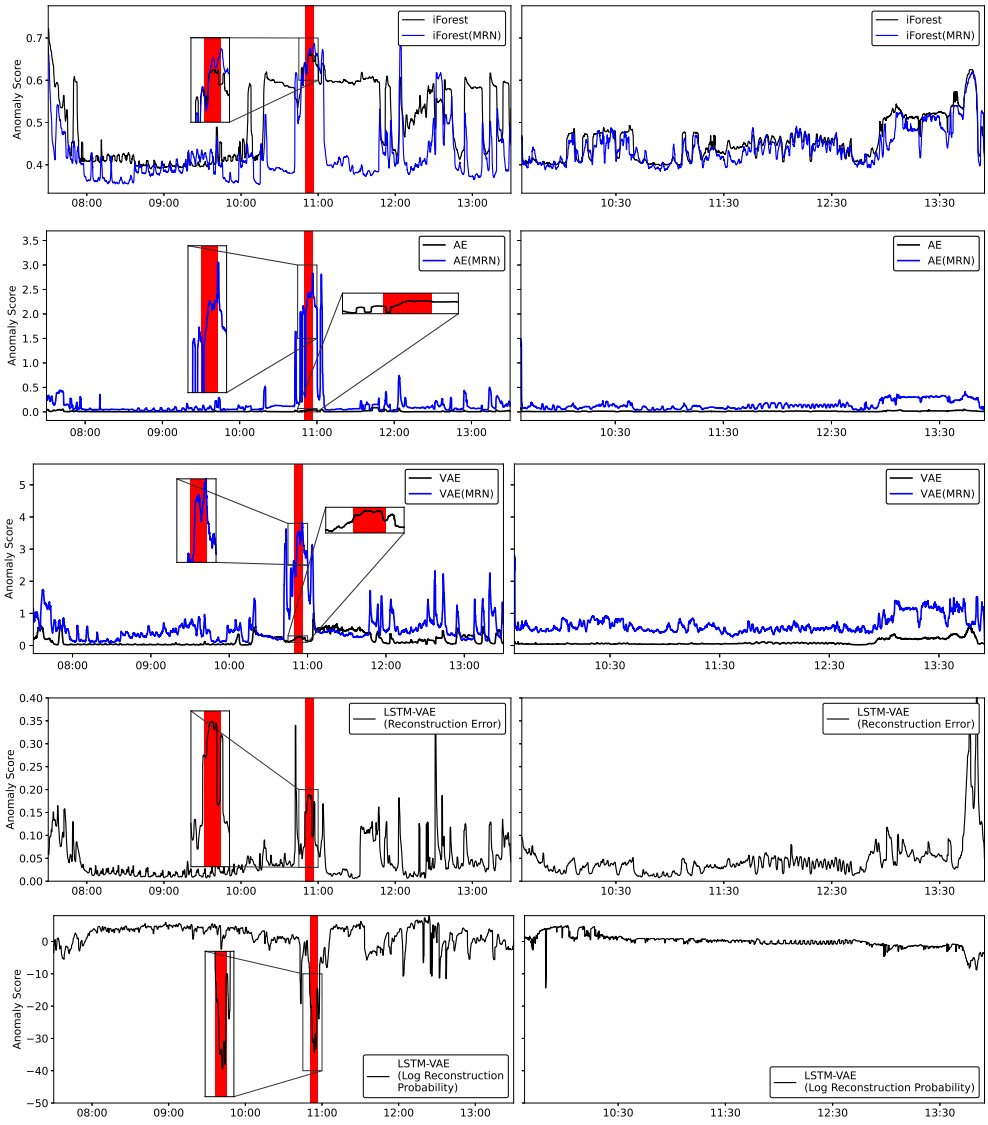


Figure 4.2: Visualization of the anomaly scores over time in the test set. The left five sub graphs are the anomaly score from the day where a fault is introduced. The right five sub graphs show the anomaly score from the normal operation test day. The red background on the graphs represents the ground truth of the fault.

Table 4.1: Comparison of different methods for engine fault detection.

Method	ω	TTD ↓	DSF ↑
iForest (MRN)	39	163	0.553
	59	175	0.564
	79	178	0.565
	99	182	0.565
AE (MRN)	39	77	0.755
	59	78	0.781
	79	78	0.781
	99	79	0.781
VAE (MRN)	39	76	0.757
	59	80	0.749
	79	83	0.783
	99	84	0.785
LSTM-VAE (Reconstruction Error)	39	75	0.782
	59	77	0.782
	79	80	0.784
	99	83	0.782
LSTM-VAE (Log Reconstruction Probability)	39	**60**	0.774
	59	65	0.786
	79	66	**0.791**
	99	72	**0.791**

Quantitative results

Table 4.1 summarizes the performance of different methods in terms of TTD and DSF. Only the results of the Iforest, AE, and VAE with the multi-regime normalization is shown since these models with standard normalization fail to detect the fault. It is shown that the AE, VAE, and LSTM-VAE performs better than the iForest in our case. The AE, VAE and LSTM-VAE shows a similar performance with TTD around 80 seconds and DSF around 0.78. For the LSTM-VAE, it is shown that using log reconstruction probability as anomaly score provides lower TTD as well as higher DSF than using reconstruction error. With log reconstruction probability, the LSTM-VAE can achieve TTD as 60 seconds and DSF as 0.791.

4.2 Fault prognostics using LSTM networks

Fault prognostics is the key action of a PHM system since the prognostics algorithm aims to estimate the available time before an anomalous component suffers an operational failure. Such estimations are normally referred to as the remaining useful life (RUL) and used to devise an ideal maintenance schedule.

In this section, the feasibility of using data-driven fault prognostics for marine diesel engine is investigated. First, run-to-failure data of engine is collected. Then the RUL targets can be constructed and then aligned with the sensor measurements. Finally a RUL predictor can be built by utilizing machine learning regression techniques.

4.2.1 RUL targets construction

The piece-wise linear (PwL) degradation model is widely used to label the RUL targets. In the original PwL model, all engines utilize the same initial RUL values. Since the engines might follow different degradation pattern, an improved PwL model as presented in [80] is used.

Fig. 4.3 depicts the improved PwL model. An anomaly detector is used to detect the time step where the fault is initially provoked. Then the RUL targets are constructed based on
degrad

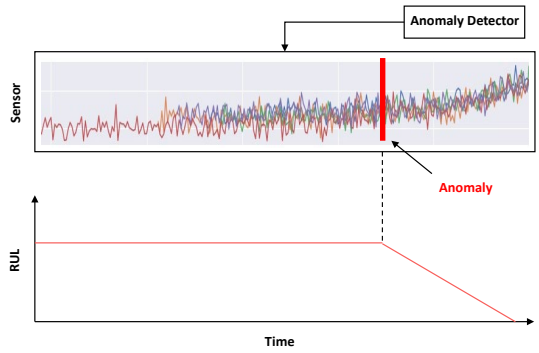


Figure 4.3: Illustration of the improved piece-wise linear degradation model.

4.2.2 Data augmentation

In real-world applications, however, run-to-failure data might be more time-consuming and difficult to acquire. In such a context, data augmentation techniques could be used to alleviate this problem. First, random white Gaussian noise is added to each sensor measurement in the original training set with a random signal-to-noise ratio between 70 and 90%. Next, following [81], a random interval of time steps are removed after d_{ft} to also include some time-series that will end some time prior to failure.

4.2.3 Network architecture

Following the recent RUL prediction research [82], LSTMs and feed-forward neural networks (FNNs) are used as the main building blocks in this paper. The LSTM layers are used to learn temporal and long-term dependencies within the features of degradation data. The FNN layers are then used to map all extracted features before a dropout layer is used to reduce overfitting. Dropout [83] randomly drops units of the dropout layer during training. This forces the model to learn to construct generalized representations of the input data. The last layer consists of a fully connected output layer with one unit. This layer handles error calculations and perform RUL predictions. The mean squared error is utilized as the loss function.

The proposed network architecture can be seen in Fig. 4.4. Two layers of LSTM and two layers of FNN are used, with skip-connection on last layer of LSTM and FNN, respectively. The skip connections in the model use element-wise addition to combine

the ac

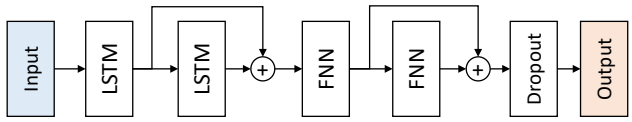
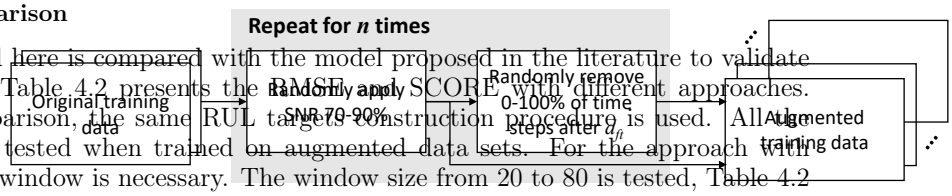


Figure 4.4: Network architecture for RUL predictions.

4.2.4 Experimental results

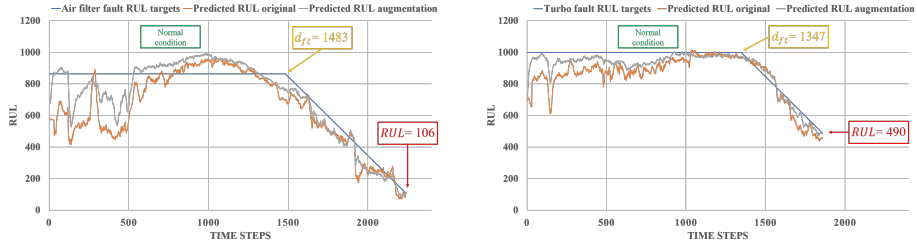
Baseline comparison



The model used here is compared with the model proposed in the literature to validate the approach. Table 4.2 presents the RMSE and SCORE with different approaches. For a fair comparison, the same RUL targets construction procedure is used. All the approaches are tested when trained on augmented data sets. For the approach with CNN, a sliding window is necessary. The window size from 20 to 80 is tested, Table 4.2 provides the best results within the window size. It is shown that the model used in this paper provides the smallest RMSE and SCORE when compared with other approaches.

Table 4.2: Comparison with different approaches for engine RUL predictions.

Approach	RMSE	SCORE
CNN+FNN [84]	96.26	40.32
LSTM+FNN [85]	285.09	67.63
RBM+LSTM+FNN [86]	120.54	128.70
LSTM+FNN+SC	**85.76**	**9.92**



(a) (b)

Figure 4.5: The prediction results on the air filter fault and the turbo fault in the test set.

Effect of data augmentation

Fig. 4.5 compares the RUL predictions on both the air filter fault and the turbo fault in the test set when the model is trained on augmented data sets and the original training set. The RUL predictions are more noisy when no augmented data is used. In addition, when the model is trained on the augmented data set, the RUL prediction is close to

the RUL target not only at the end of the engine's life, but also at the early stage of degradation. It proves the advantage of using the data augmentation technique.

4.3 Chapter summary

This chapter uses diesel engines on ships as an example. Two application scenarios are considered:

- A long-short term memory based variational autoencoder (LSTM-VAE) is proposed for fault detection. The encoder and decoder of VAE are implemented with LSTM to introduce temporal dependencies. The log reconstruction probability can be used as the anomaly score. This model follow the semi-supervised anomaly detection setting that only requires no fault data for training. From the experiment on a maritime diesel engine operating in the real world, it is showed that the LSTM-VAE can accurately detect the air filter clogging fault and it outperforms several baseline methods.
- a LSTM network is used for fault prognostics. Run-to-failure data of two fault-types in two different engine load profiles is collected in the lab. Data augmentation technique is used to augment the training data. Experimental results show that the model provides accurate remaining useful life predictions for two different fault types: air filter and turbo fault. It suggests that the proposed model has high generalization power towards different engine load profiles.

Fault detection and fault prognostics are essential components of optimal maintenance. The model presented in this chapter can be used to implement engine fault detection and prognostics to support smart ship operations.

5 Case study: Sea state estimation

The sea state refers to the general condition of the ocean with respect to wind waves and swell at a certain location in oceanography. A sea state is usually characterized by statistical parameters, e.g., significant wave height, average wave frequency, and peak frequency [87].

Nowadays, the majority of marine vessels are equipped with sensors that measure the ship motions in 6 degrees of freedom. The motion responses reflect the sea state conditions and therefore a ship can be considered as a large wave buoy. From this perspective, a vessel is essentially equipped with an environmental condition estimation system [88]. Estimating the sea state based on the ship motion responses is of interest and has been investigated in the literature. In this chapter, the sea state estimation from ship motion responses is treated as a supervised learning problem. This chapter is based on the research results from papers II, III, and VI.

5.1 An ensemble model to estimate the wave parameters

In this section, a data-driven model is proposed to estimate the sea state based on ship motion responses. Features are generated by means of statistical, temporal, spectral, and wavelet analysis. The features are then used to predict the sea state information with machine learning models.

5.1.1 Multi-domain feature construction

Considering a signal is a discrete time series data $(x_1, x_2, \ldots, x_n)$ with length n, four broad categories of features are constructed to describe sea state pattern.

Domain-knowledge features Two basic features are extracted. *speed*: the forward speed is important for estimating wave characteristics due to the Doppler shift [89], *diff_ angle*: the difference between course angle and heading angle during ship maneuvering.

Statistical features Seven basic statistical features are extracted from each DOF measurement. Six standard features of the signal including *maximum*, *minimum*, *mean*, *variance*, *skew*, and *kurtosis* are considered. Additionally, the q *quantile* information of the signal is extracted, which is the value greater than q of the ordered values from the signal. The variable q is selected as 0.2, 0.4, 0.6, and 0.8.

Temporal features Firstly five temporal features are considered, which include: *absolute sum of change* ($\sum_{i=1}^{n-1} |x_{i+1} - x_i|$), *absolute energy* ($\sum_{i=1}^{n} x_i^2$), *mean second derivative center* ($\frac{1}{2(n-2)} \sum_{i=1}^{n-2} \frac{1}{2}(x_{i+2} - 2x_{i+1} + x_i)$), *zero cross* (the number of the signal crossings zero), *longest strike above mean* (the length of the longest consecutive

subsequence in a signal that is largest than its mean).

Two advanced temporal features are also extracted: *Sample entropy* and *Autocorrelation*.

Welch spectral features Welch method is an approach to transform a signal from the time domain to the frequency domain and estimate the power of a signal at different frequencies. After the signal is transformed into frequency domain, four basic spectral features including *max power spectrum*, *fundamental frequency*, *max frequency*, and *median frequency* are extracted. Additionally, five features related to the shape of the spectrum [90] is also extracted: *centroid*, *variation*, *spread*, *skewness*, *kurtosis*.

Wavelet features The wavelet transform is a time-frequency analysis method which can analyze a signal with multi-scales both in time and frequency domain. The wavelet transform is used to split a signal into different frequency sub-bands. The Daubechies wavelet of order 1 (db1) is selected as the basis function and the decomposition level is five. In total five approximation components and five detail components are obtained. For each components, the *mean*, *variance*, *median*, *skewness*, *kurtosis*, *absolute energy*, *absolute sum of changes*, and *zero cross* are extracted.

5.1.2 Minimum-redundancy maximum-relevance (mRMR) feature selection

In order to select salient features from the constructed multi-domain features, mRMR [91] feature selection framework is utilized. The mRMR criterion is a filter feature selection method which can effectively reduce the redundant features while keeping the relevant features for the model. The mRMR criterion can be expressed as:

$$f_{mRMR}(x_i) = I(y, x_i) - \frac{1}{|S|} \sum_{x \in S} I(x_s, x_i) \tag{5.1}$$

where the function $I(\cdot, \cdot)$ denotes the mutual information. $|S|$ is the size of the feature set and $x_s \in S$ is one feature out of the feature set.

5.1.3 Ensemble model

Three machine learning models are ensemble using voting method, which is simply averaging the predictions from the three models. Tree-based model (GBDT), kernel-based model (SVR) and distance-based model (kNN) are used here to ensure the diversity of sub-models, which also influences the ensemble performance.

5.1.4 Experimental results

The mean absolute error (MAE) is used as the evaluation metric. 5-fold cross-validation is performed to avoid possible selection bias on splitting the dataset.

To evaluate the performance of the proposed method, four baselines model are also implemented: (1) Random Guess: a simple model that makes the predictions by randomly drawing from the training data distribution. (2) Linear Regression with Elastic Net regularization (EN): a regularized linear regression method that linearly combines the l_1 and l_2 penalties, the hyperparameter for l_1 and l_2 are tuned. (3) Multilayer Perceptron (MLP): a class of feedforward artificial neural network, ReLU is used as the activation function and Adam is used as the optimizer. The learning rate and the weight for l_2 regularization are tuned. (4) Random Forest (RF): an ensemble model that

uses the decision trees as base learners and bagging to improve the performance. The maximum depth, the minimum number of samples required to be at a leaf node, number of features to consider when looking for the best split are tuned.

In addition, an end-to-end model SeaStateNet [92] is implemented. Since the model is originally designed for classification, the output nodes in the last layer are changed to three and the loss function is changed to mean square error in order to adapt to the dataset.

Table 5.1 reports the MAE of the models, evaluated through the 5-fold cross-validation. The results are presented in $mean \pm std$ format. For the feature-based approaches, GBDT consistently outperforms the other approaches. The high MAEs of the EN model suggest that the sea state characteristics are better captured using nonlinear relationships. The ensemble model consists of GBDT, kNN, and SVR outperforms any individual models. The end-to-end approach SeaStateNet outperform the ensemble model in D_m but have a slightly higher MAE in H_s and T_p. The reason might be that the mean wave direction is not so sensitive to the constructed features. Besides, the errors for H_s and T_p are in an acceptable range, while the error for D_m is relatively high even though it is clearly better than the random guess. The reason might be that most of the time the vessel is operating near the coast and the mean wave direction from the weather forecast system is not so accurate in this region.

Table 5.1: The MAE values of the different methods for wave paramters estimations

Model	Wave Characteristics		
	$H_s\ (m)$	$D_m\ (^\circ)$	$T_p\ (s)$
EN	0.484 ± 0.027	77.59 ± 3.32	2.032 ± 0.172
MLP	0.431 ± 0.045	71.84 ± 6.50	1.851 ± 0.119
RF	0.378 ± 0.024	64.34 ± 4.62	1.686 ± 0.116
kNN	0.359 ± 0.025	60.02 ± 3.58	1.655 ± 0.095
SVR	0.361 ± 0.024	60.96 ± 2.58	1.649 ± 0.100
GBDT	0.337 ± 0.027	59.28 ± 2.26	1.607 ± 0.096
SeaStateNet	0.348 ± 0.019	$\mathbf{53.82 \pm 3.09}$	1.659 ± 0.178
Our ensmble	$\mathbf{0.334 \pm 0.030}$	57.72 ± 1.30	$\mathbf{1.528 \pm 0.084}$

5.2 Incorporating model-based method to estimate the wave parameters

Machine learning methods often require an extensive amount of training data and they only perform well when the training and testing data are sampled independently and identically from the same distribution [8]. The vessel is usually operated in the same route for a specific period, the historical data collected in the real world, therefore, contains limited number of sea state and can not cover the entire range of possible sea states. When the vessel is deployed into a new route or experience a new sea state, the machine learning model trained with historical data is likely to fail. A failing on the sea state estimation might cause severely operational and financial costs.

To overcome this shortcoming, the feasibility of the hybrid approach for sea state estimation using ship motion responses will be investigated. The ML model estimates the

current sea state with predictive uncertainty, while, in parallel, the wave buoy analogy method provides the estimation results using the same ship motion responses. The estimation results from both methods are then fused together.

5.2.1 Methodology

Fig. 5.1 shows the schematic illustration of the proposed hybrid method. Historical data containing ship motion and corresponding sea state information is collected to train a machine learning model. The machine learning pipeline consists of feature extraction, feature selection, and model training. The Gaussian process is chosen since it not only provides predictions but also uncertainty. The wave buoy analogy (WBA) method builds on a comparison between measurements of response spectrum and calculated ones. By minimizing the discrepancy between the measured and calculated spectrum, the sea state is determined. Then the uncertainty-aware confusion module receives the sea state estimation results from these two methods. The WBA estimation results are used to compensate for the ML results according to its uncertainty. In this way, the hybrid estimation results are the combination of the estimation results made by the ML model and the WBA method.

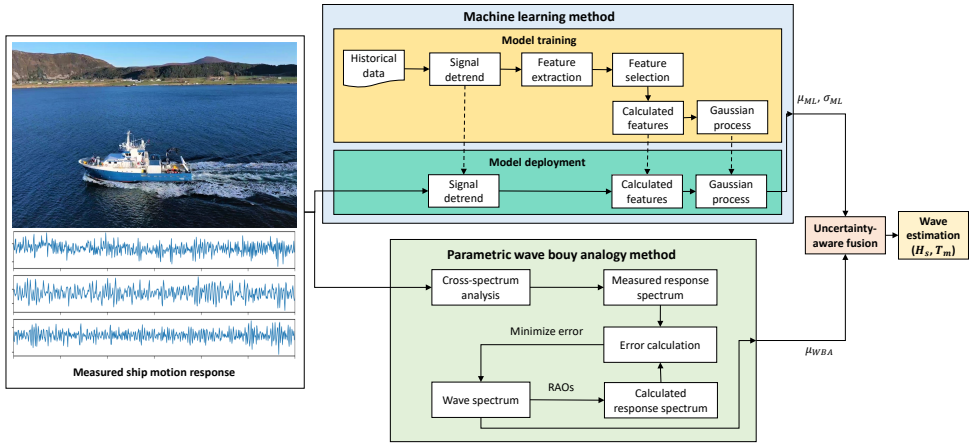


Figure 5.1: Schematic illustration of the proposed hybrid approach. The upper rectangle is the ML model and the lower rectangle is the model-based method.

The WBA method will not be detailed described in this section since it is not the focus of the dissertation. For the ML model, the same feature construction and selection procedures as Section 5.1 is used. Only the learning model is changed to Gaussian process since it can represent the predictive uncertainty naturally.

Uncertainty-aware fusion

As shown in Fig. 5.2, the estimation results from the machine learning model and the wave buoy analogy method are assumed to follow a distribution as $P(y|ML)$ and $P(y|WBA)$, respectively. Since $P(y|ML)$ and $P(y|WBA)$ are independent, the final result can be obtained through eq.(5.2). In this way, the hybrid estimation results would

move towards the WBA results if the uncertainty of the ML results are high.

$$P(y|ML, WBA) = P(y|ML) \cdot P(y|WBA) \tag{5.2}$$

The $P(y|ML)$ follows a Gaussian distribution with mean μ_{ML} and variance σ^2_{ML}, which can be obtained through Gaussian process model. For the wave buoy analogy method, the uncertainty is not easy to measure directly, a Gaussian distribution is also assumed for $P(y|WBA)$ with mean μ_{WBA} and variance σ^2_{WBA}. Then the final estimation result is:

$$y_{ML,WBA} = \mu_{ML} + \frac{\sigma^2_{ML}(\mu_{WBA} - \mu_{ML})}{\sigma^2_{ML} + \sigma^2_{WBA}} \tag{5.3}$$

Here σ^2_{WBA} is a parameter which can be tuned to adjust the final results towards ML or WBA results.

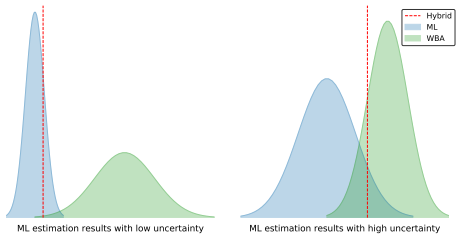
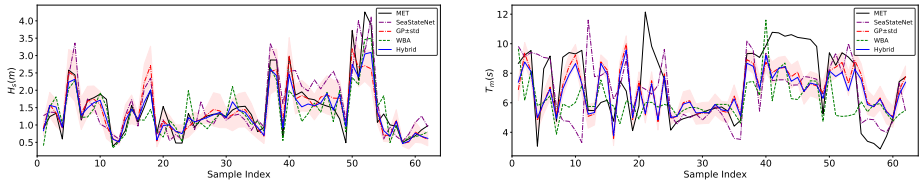


Figure 5.2: Illustration of the uncertainty-aware fusion.

5.2.2 Experimental results

Fig. 5.3a shows the significant wave height for each sample, where MET stands for the “actual" value from the Norwegian Meteorological Institute. Fig. 5.3b presents the same graph for the mean wave period. The value of σ^2_{WBA} are selected as 0.5 and 3.5 for significant wave height and mean wave period, respectively. The GP model provides fairly accurate results in terms of the significant wave height. For the mean wave period, the predictions are mostly distributed in the range of $5s$ to $8s$, therefore it provides relatively bad results for low and high wave periods. Similar results are observed for the WBA method and the SeaStateNet model. The reason might be that the vessel itself is a filter and its motions are only sensitive in a specific range of the wave frequency. The hybrid model predictions are the GP model predictions corrected by the WBA method. As shown in Fig. 5.3a, the GP model predictions with high uncertainty are corrected, which can be easily observed for samples 17, 51, and 52. The GP predictions and hybrid predictions in Fig. 5.3b is quite similar since we put a relatively large σ^2_{WBA}. The reason is that the results from WBA for the mean wave period are relatively less accurate compared with the significant wave height.

Table. 5.2 summarized the overall performance in terms of MAE. The GP model performs better than the SeaStateNet model. The reason might be that our data is limited. It is shown that the GP predictions provide an overall low error when comparing with the WBA method. The hybrid method reduces the MAE in terms of significant wave



(a) (b)

Figure 5.3: Wave estimation by different approaches: (a) significant wave height; (b) mean wave period.

height by about 10% when comparing with the GP method. For the mean wave period, the hybrid method gives a similar error with the GP method. From the experiment, the hybrid method can reduce the estimation errors by correcting the high uncertainty GP predictions with the WBA predictions. Compared with the rest of the models, the hybrid model has the smallest error.

Table 5.2: MAE of different sea state estimation method

Sea State	SeaStateNet	WBA	GP	Hybrid (GP+WBA)
$H_s(m)$	0.392	0.316	0.268	0.248
$T_m(s)$	1.758	1.998	1.533	1.529

5.3 Directional wave spectrum estimation

The above two sections aim to estimate the integrated wave parameters. The integrated wave parameters are a summary expression of the wave spectrum. Ideally, a 2D directional wave spectrum could be estimated to fully describe the sea state. In addition, the 2D directional wave spectrum is fundamental for operational safety analysis such as extreme value analysis.

In such a context, this section aims to build a machine learning model for estimating the 2D directional wave spectrum using ship motion responses.

5.3.1 Methodology

The proposed method consists of two separate networks, as outlined in Figure 5.4. The inputs are the cross spectrum of the ship motion. The cross spectrum is normalized and then fed into the estimation network to be converted into a 2D wave spectrum. The discrimination network takes a 2D wave spectrum as input and distinguishes whether it is generated from the estimation network or is the actual wave spectrum. In the training phase, the estimation network tries to generate a realistic wave spectrum while the discrimination network tries to distinguish it. In this way, the two networks are improved together and the high-order statistics of the output wave spectrum are penalized to force the estimation network to provide continuous and realistic results. At inference time, the discrimination network is omitted, and the estimation network is used to output the

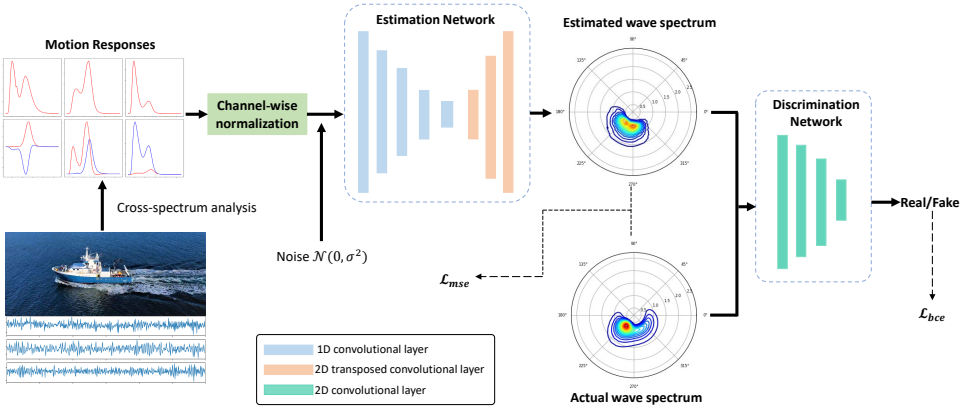


Figure 5.4: Schematic illustration of the proposed model for 2D directional wave spectrum estimation using ship motion responses.

estimated 2D wave spectrum from ship motion responses.

Adversarial training

A hybrid loss which is a weighted sum of two terms is used. The first is the mean square error that encourages the estimation model to predict the wave spectrum. The second loss term is based on the adversarial convolutional network. This loss term is large if the adversarial network can discriminate the output of the estimation network from the actual wave spectrum. The aim of the adversarial term is to penalize mismatches in the high-order spectral power value statistics, e.g., the power value of wave spectrum should be smooth in the near region, which is not accessible by the mean square loss function.

Given a training ship motion responses x and a corresponding wave spectrum y, the estimator E and the discriminator D would be competed in a two-player min-max optimization routine:

$$\min_E \max_D \mathcal{L}(E, D) = \mathcal{L}_{mse}\left(E(x), y\right) - \lambda[\mathcal{L}_{bce}(D(y), 1) + \mathcal{L}_{bce}(D(E(x)), 0)] \tag{5.4}$$

where $\mathcal{L}_{mse}$ is the mean square loss, $\mathcal{L}_{mse}(\hat{z}, z) = |\hat{z} - z|^2$. $\mathcal{L}_{bce}$ is the binary cross-entropy loss, $\mathcal{L}_{bce}(\hat{z}, z) = -z \log \hat{z} - (1 - z) \log(1 - \hat{z})$. λ is a hyperparameter to balance these two different losses.

The training of the estimation model minimizes the mean square error loss while at the same time trying to fool the discriminator model. The objective function of the estimation model is:

$$\mathcal{L}_E = \mathcal{L}_{mse}\left(E(x), y\right) - \lambda \mathcal{L}_{bce}(D(E(x)), 0) \tag{5.5}$$

In practice, the term $-\mathcal{L}_{bce}(D(x, E(x)), 0)$ is replaced by $+\mathcal{L}_{bce}(D(x, E(x)), 1)$ [93]. This means that the probability that the adversarial model predicts the estimated wave

spectrum to be the actual one is maximized, instead of minimizing the probability that the adversarial model predicts the estimated wave spectrum to be synthetic.

For the adversarial model, only the binary classification loss is related. Therefore, training the adversarial model is equal to minimizing the following objective function:

$$\mathcal{L}_D = \mathcal{L}_{bce}(D(y), 1) + \mathcal{L}_{bce}(D(E(x)), 0) \tag{5.6}$$

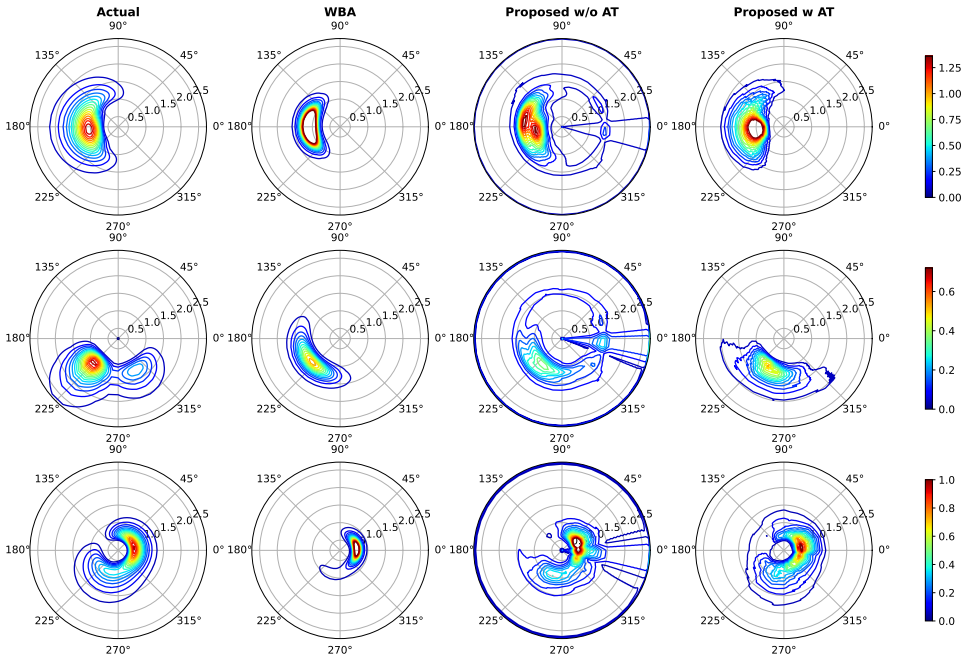


Figure 5.5: Examples of contour plots of the estimated directional wave spectrum based on perfect motion spectrum.

5.3.2 Experimental results

In this part, the performance of the proposed method will be evaluated. Two baseline models are implemented for comparison:

- **Bayesian wave buoy analogy method**: This method is a model-based method for directional wave spectrum estimation using ship motion responses. The wave spectrum is represented in a discrete frequency-directional domain. The fundamental idea is to minimize the difference between the measured and the calculated spectrum. However, this forms an ill-posed inverse problem, and therefore smooth prior is introduced to solve the problem in the Bayesian framework. In this paper, a two hyperparameters method [94] is used. The two hyperparameters are responsible for the smooth prior of wave spectrum in the discrete frequency and discrete direction, respectively. Details of this method is described in [94].

- **Neural network model without adversarial training**: This model is the estimation network proposed in this paper. The discriminator network is neglected by setting the hyperparameter λ as 0. This model is implemented to show the effect of adversarial training.

In the following, the Bayesian wave buoy analogy method is denoted as "WBA", the neural network model without adversarial training is denoted as "Proposed w/o AT", and the proposed neural network model with adversarial training is denoted as "Proposed w AT".

Figure 5.5 presents the estimated directional wave spectrum from three random samples in the test set. The colors of values larger than the color bar upper limits remain the same as that of the upper limit. It is shown that the Bayesian WBA method provides a similar shape of the spectrum as the actual ones but the values are less accurate. The reason is that the performance of this method depends on the two hyperparameters and the initial guess of the wave spectrum. Several combinations of hyperparameters and initial guesses are used to yield the best-estimated spectrum. For the neural network model, the model without adversarial training clearly presents spurious lines in the wave spectrum. Even though the shape of the estimated wave spectrum is similar to the actual wave spectrum, it has high total wave energy. The model with adversarial training better enforces the spatial consistency of the wave spectrum. It also smooths and strengthens the high energy density area of the wave spectrum.

Table 5.3: MAE of different methods on the test set for 2D wave spectrum estimation

Methods	Pixel	Integrated wave parameters			
		$H_s(m)$	$T_m(s)$	$D_m(^\circ)$	σ_s
WBA	0.033	0.606	0.573	**12.88**	0.234
Proposed w/o AT	0.043	1.265	0.952	18.80	0.353
Proposed w AT	**0.018**	**0.239**	**0.361**	13.95	**0.153**

Table 5.3 summarizes the overall performance in terms of MAE. The pixel error represents the error in the 2D wave spectrum. Four integrated wave parameters are also examined: significant wave height H_s, mean wave direction D_m, mean wave period T_m, and directional spreading parameter σ_s. Compared with the neural network model without adversarial training, the error of the WBA method in terms of pixel-level and integrated wave parameters is relatively low. By incorporating adversarial training, these errors are reduced significantly. In this comparison, our model with adversarial training has the smallest error.

Generalization to JONSWAP-type wave spectrum

In this part, the generalization ability of the model is evaluated with the JONSWAP-type wave spectrum. The JONSWAP type spectrum has a more pronounced peak in the spectrum than the Pierson-Moskowitz (PM) type wave spectrum. The JONSWAP wave spectrum and the Torsethaugen wave spectrum (a double peak JONSWAP-type spectrum) are used to generate two extra test sets with 100 samples, respectively. The

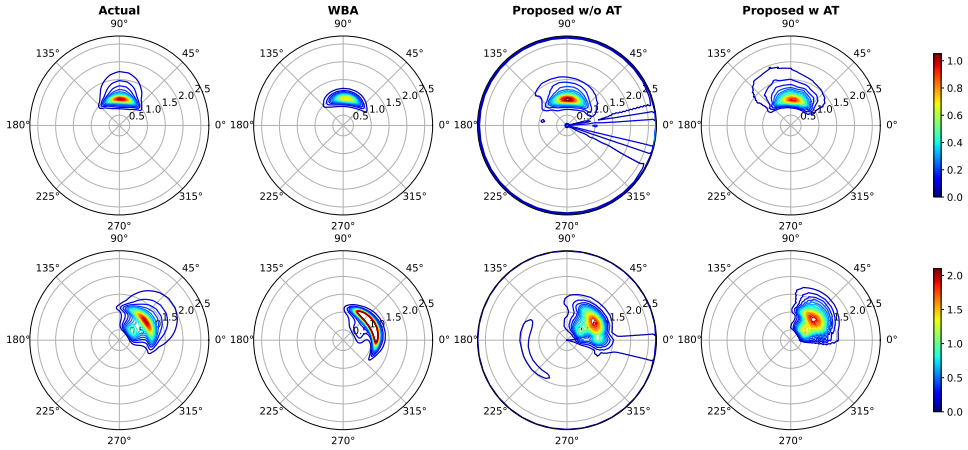


Figure 5.6: Examples of contour plots of the estimated directional wave spectrum for JONSWAP-type wave spectrum.

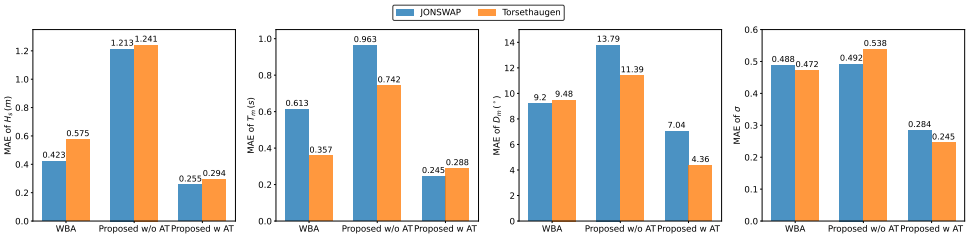


Figure 5.7: MAE of the integrated wave parameters for the JONSWAP and Torsethaugen wave spectrum.

trained model is then used to estimate the 2D wave spectrum. Figure 5.6 shows the estimated 2D wave spectrum from two examples in the two extra test sets, respectively. The proposed model presents a less narrow spectrum than the actual one, which might be due to the Pierson-Moskowitz type wave spectrum used in the training data. Nonetheless, the proposed model still provides a reasonable estimate.

Figure 5.7 summarizes the MAE of H_s, T_m, D_m, σ for the JONSWAP and Torsethaugen wave spectrum. The proposed model achieves the lowest deviation among these three methods. It demonstrates that the proposed model successfully captures the relation between ship motion and wave spectrum, therefore, it is able to estimate the type of wave spectrum not present in the training data.

Robustness to noisy ship motion measurements

In this section, the effect of cross spectrum analysis and the noises in ship motion on the estimated results will be evaluated. The cross spectrum analysis in this paper is performed through the Welch method. White noise is added and four different SNR

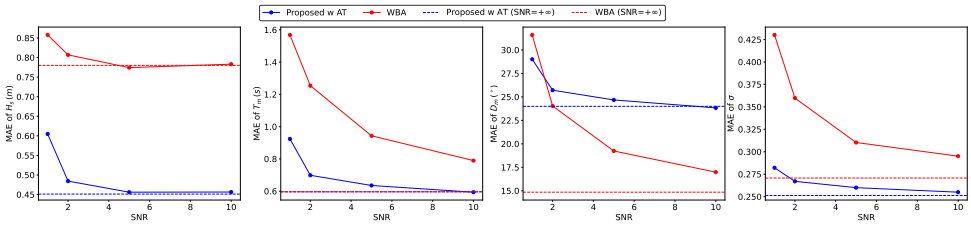


Figure 5.8: MAE of the integrated wave parameters for the motion responses under different SNR levels.

levels, 10, 5, 2, 1, are investigated. For simplification, the time series ship motion without noise added is denoted as "SNR=$+\infty$". In "SNR=$+\infty$", only the effect of cross spectrum analysis is included. Figure 5.8 compares the MAE of integrated wave parameters in WBA and the proposed model under different SNR levels. The proposed model is less sensitive to noise than the WBA. The WBA method shows low error in D_m while the proposed model has low error in H_s, T_m and σ.

5.4 Chapter summary

This chapter aims to estimate the sea state condition from ship motion responses. The data-driven methods are used, where three aspects are considered:

- The problem is treated as a regression problem and three integrated wave parameters are considered. Statistical, temporal, spectral, and wavelet features are extracted from the ship motion data. An ensemble model is then built. Experimental results from real-world operation data show that the model can estimate the significant wave height and mean wave period with high accuracy.
- To improve the performance of wave parameters estimation when the data is far away from the training set, a hybrid model consisting of model-based and data-driven models is proposed. The data-driven model provides not only the estimation results but also the uncertainty of the estimation results. The model-based approach then compensates for the results of the data-driven approach based on uncertainty. A substantial decrease in the mean absolute error was observed for the significant wave height from the experimental results.
- To estimate directional wave spectrum, an estimation network and discriminant network based on convolutional neural networks are proposed. The high-order inconsistencies of the wave spectrum from the estimation network are penalized by the estimation network, thereby forcing the estimation network to produce accurate and realistic results. Simulation studies show that the proposed model guarantees the smoothness of the wave spectrum and provides accurate estimation results.

The sea state conditions are of key importance for ship operation. The methods developed in this chapter might improve the accuracy of sea state estimation from motion responses.

6 Conclusion and further work

This dissertation has proposed and discussed the use of data-driven methods for decision supports in smart ship operations. It has presented research findings concerning three case studies for developing data-driven models for decision support of vessels. All contributions in this dissertation aim to enhance vessel performance and further aid autonomous vessels. The data-driven method is not a new technology, but it has been developed rapidly in recent years due to the enhancement of computing power and the increase of data resources. The work presented here hopes to promote the adoption of data-driven methods in the maritime domain.

6.1 Summary of contributions

Assessing the status of the internal components of the ship as stated in RO1 is one of the key goals of this dissertation. The solution is to develop data-driven fault diagnosis and prognostics models. Deep neural networks including convolutional neural network and long short-term memory are extremely powerful and easy to use since it requires less manual feature extraction process. The fault detection and isolation problem can be formulated as a binary or multi-class classification problem if sufficient fault or failure data is available. However, the fault data might be difficult to collect as stated in RO2. Since normal data is easy to collect, the outlier/anomaly detection technique can be used to construct a fault detection model (fault isolation cannot be achieved). In this dissertation, a fault detection method that makes use of the above technique and further considers temporal dependencies is developed and verified on a maritime diesel engine. As stated in RO3, another goal of this dissertation is to make the environmental conditions that the ship is currently operating in more accessible. The task is then to use ship motion responses to estimate the sea state information. The problem is formulated as a regression problem. It is also shown that incorporating prior knowledge can benefit the data-driven method, which could increase its accuracy as well as reliability, as stated in RO4.

The main contributions of this dissertation are as follows:

- Present the fundamentals to use data-driven methods for decision support to enhance vessel performance. The effectiveness of the methods is shown through three different case studies.
- Propose a data-driven fault detection method for maritime components. The method uses only normal operation data for training.
- Propose a data-driven model to estimate the detailed 2D directional wave spectrum from ship motion responses.

- Present methods to account for the predictive uncertainty of data-driven method and techniques to compensate its prediction to increase the reliability.

6.2 Summary of publications

The summary of publications are as follow:

Paper I presents a deep convolutional neural network to detect and isolate potential thruster failures for DP vessels based on the control signals and logged ship motion data. The model is trained with the historical data set that contains normal and fault data. The focal loss is used to handle the unbalanced dataset since the amount of normal operation data is much larger than that of fault data. The method is validated in a simulated environment and can detect and isolate the failure with high accuracy.

Paper II presents a data-driven model for performing real-time onboard wave parameters estimation using ship motion responses for vessels. Features from multi-domain such as statistical, temporal, spectral, and wavelet analysis are constructed from time-series ship motion data. The model is an ensemble from three diverse models: tree-based model (GBDT), kernel-based model (SVR), and distance-based model (kNN). Data collected from real-world scenarios show that the method can provide relatively accurate results in terms of significant wave height and peak period.

Paper III builds on the approach presented in paper II. This paper uses the Gaussian process to construct the model, which can account for the predictive uncertainty naturally. The estimated results from the model-based wave buoy analogy method are used to compensate for the prediction according to the uncertainty. The proposed method aims to reduce the possibility of failure in the ML model when the encountered sea state is not in the training set.

Paper IV proposes a long-short term memory based variational autoencoder (LSTM-VAE) for anomaly detection for maritime systems. The encoder and decoder of VAE are implemented with LSTM to introduce temporal dependencies. This method enables feasible and robust detection without further assumptions on data. The proposed method follows the semi-supervised framework that only the data in normal operation is necessary for training. Experiment on a maritime diesel engine operating in the real world shows that the LSTM-VAE can accurately detect the air filter clogging fault.

Paper V presents an LSTM network for fault prognostics of the marine diesel engine. Run-to-failure data of two fault-types in two different engine load profiles are collected in a hybrid engine lab. Data augmentation technique is used to augment the training data. The optimal network architecture is obtained. Experimental results show that the model provides accurate remaining useful life predictions for two different fault types.

Paper VI is a continuation of the works presented in paper II and III. Paper II and III formulate the sea state estimation problem as a regression problem, and only integrated wave parameters can be obtained. This paper focuses on 2D directional wave spectrum estimation. An estimation network and discriminant network based on convolutional neural networks are built together. The high-order inconsistencies of the wave spectrum from the estimation network are penalized by the estimation network. Simulation studies show that the proposed model guarantees the smoothness of the wave spectrum and provides accurate estimation results.

6.3 Future work

This thesis has mainly focused on two aspects for decision support of vessels: fault diagnostics and prognostics, and sea state estimation. These are all achieved by data-driven methods despite different kinds of models being used. The below bullet points provide suggestions for how the presented research may be extended.

- In general, the models used in this dissertation are trained with historical data and then deployed. The lifetime of the modern vessel is about 25 to 30 years. During this period, the vessel can continuously collect data and an automatic label pipeline can be built. The performance of machine learning models can be improved when more data is available. In a research perspective, incremental learning [95], continual learning, or lifelong learning [96] are suggestions for future research. This is extremely important for neural networks because catastrophic forgetting [97] can occur when training an already trained neural network with a new dataset. Deployed machine learning models can improve themselves over time as the ship operates.

- The goal of the proposed data-driven model is to facilitate decision support or automation of ship operations. When the input data is far from the training set, the model may fail. This is important for critical operational scenarios. Confidence intervals for the model predictions should be provided so that the operator can better judge whether to accept the model predictions. Therefore out-of-distribution detection or model uncertainty [14] can be further investigated. Additionally, the model may need to provide transparent explanations of how it makes such predictions in order to gain the trust of the operator. This includes building an explainable ML model at the beginning or techniques for visualizing and interpreting the predictions of a black-box model [98]. Future research and development might explore them.

- The prior knowledge of the vessel might be used to incorporate with the data-driven method to provide more accurate predictions. For the sea state estimation, data covering the entire range of sea states is almost impossible to collect in the real world. The prior knowledge, including the mathematical model of the vessel and the expert knowledge, can be used to ease the problem. How to organically combine prior knowledge and data-driven method might be investigated in the future. This will also benefit the development of the digital twin since it makes use of data-information models as well as mathematical models.

- A decision support interface can be designed to communicate information to vessel operators, whether onboard or at a remote control center. This will be an important step towards the availability of methods that will benefit ship operations. It could also facilitate continual or lifelong learning of the deployed models.

References

[1] L. I. Hatledal, R. Skulstad, G. Li, A. Styve, and H. Zhang, "Co-simulation as a fundamental technology for twin ships," 2020.

[2] O. Levander, "Autonomous ships on the high seas," *IEEE spectrum*, vol. 54, no. 2, pp. 26–31, 2017.

[3] H. Wang, "Intelligent agent-assisted decision support systems: Integration of knowledge discovery, knowledge analysis, and group decision support," *Expert Systems with Applications*, vol. 12, no. 3, pp. 323–335, 1997.

[4] A. L. Ellefsen, "A data-driven prognostics and health management system for autonomous and semi-autonomous ships," 2020.

[5] D. J. Power, *Decision support systems: concepts and resources for managers*. Greenwood Publishing Group, 2002.

[6] D. Solomatine, L. M. See, and R. Abrahart, "Data-driven modelling: concepts, approaches and experiences," *Practical hydroinformatics*, pp. 17–30, 2009.

[7] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[8] O. Bousquet, S. Boucheron, and G. Lugosi, "Introduction to statistical learning theory," in *Summer school on machine learning*, pp. 169–207, Springer, 2003.

[9] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[10] S. Sagawa, A. Raghunathan, P. W. Koh, and P. Liang, "An investigation of why overparameterization exacerbates spurious correlations," in *International Conference on Machine Learning*, pp. 8346–8356, PMLR, 2020.

[11] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

[12] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer school on machine learning*, pp. 63–71, Springer, 2003.

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

[14] K. Lee, K. Lee, H. Lee, and J. Shin, "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," *Advances in neural information processing systems*, vol. 31, 2018.

[15] P. Han, A. L. Ellefsen, G. Li, F. T. Holmeset, and H. Zhang, "Fault detection with lstm-based variational autoencoder for maritime components," *IEEE Sensors Journal*, vol. 21, no. 19, pp. 21903–21912, 2021.

[16] P. Han, G. Li, X. Cheng, S. Skjong, and H. Zhang, "An uncertainty-aware hybrid approach for sea state estimation using ship motion responses," *IEEE Transactions on Industrial Informatics*, 2021.

[17] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[18] J. Franklin, "The elements of statistical learning: data mining, inference and prediction," *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83–85, 2005.

[19] D. Yao, C. Zhang, Z. Zhu, J. Huang, and J. Bi, "Trajectory clustering via deep representation learning," in *2017 international joint conference on neural networks (IJCNN)*, pp. 3880–3887, IEEE, 2017.

[20] H. Tang, Y. Yin, and H. Shen, "A model for vessel trajectory prediction based on long short-term memory neural network," *Journal of Marine Engineering & Technology*, pp. 1–10, 2019.

[21] R. Skulstad, G. Li, T. I. Fossen, B. Vik, and H. Zhang, "Dead reckoning of dynamically positioned ships: Using an efficient recurrent neural network," *IEEE Robotics & Automation Magazine*, vol. 26, no. 3, pp. 39–51, 2019.

[22] T. Uyanık, Ç. Karatuğ, and Y. Arslanoğlu, "Machine learning approach to ship fuel consumption: A case of container vessel," *Transportation Research Part D: Transport and Environment*, vol. 84, p. 102389, 2020.

[23] C. Gkerekos and I. Lazakis, "A novel, data-driven heuristic framework for vessel weather routing," *Ocean Engineering*, vol. 197, p. 106887, 2020.

[24] Z. Yuan, J. Liu, Q. Zhang, Y. Liu, Y. Yuan, and Z. Li, "Prediction and optimisation of fuel consumption for inland ships considering real-time status and environmental factors," *Ocean Engineering*, vol. 221, p. 108530, 2021.

[25] Y. Tan, J. Zhang, H. Tian, D. Jiang, L. Guo, G. Wang, and Y. Lin, "Multi-label classification for simultaneous fault diagnosis of marine machinery: a comparative study," *Ocean Engineering*, vol. 239, p. 109723, 2021.

[26] X. Xu, Z. Zhao, X. Xu, J. Yang, L. Chang, X. Yan, and G. Wang, "Machine learning-based wear fault diagnosis for marine diesel engine by fusing multiple data-driven models," *Knowledge-Based Systems*, vol. 190, p. 105324, 2020.

[27] Z. Tian, "An artificial neural network method for remaining useful life prediction of equipment subject to condition monitoring," *Journal of intelligent Manufacturing*, vol. 23, no. 2, pp. 227–237, 2012.

[28] L. Guo, N. Li, F. Jia, Y. Lei, and J. Lin, "A recurrent neural network based health indicator for remaining useful life prediction of bearings," *Neurocomputing*, vol. 240, pp. 98–109, 2017.

[29] X. Cheng, G. Li, A. L. Ellefsen, S. Chen, H. P. Hildre, and H. Zhang, "A novel densely connected convolutional neural network for sea-state estimation using ship motion data," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 9, pp. 5984–5993, 2020.

[30] S. Mandal and N. Prabaharan, "Ocean wave forecasting using recurrent neural networks," *Ocean engineering*, vol. 33, no. 10, pp. 1401–1410, 2006.

[31] Y. Law, H. Santo, K. Lim, and E. Chan, "Deterministic wave prediction for unidirectional sea-states in real-time using artificial neural network," *Ocean Engineering*, vol. 195, p. 106722, 2020.

[32] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[33] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE transactions on information theory*, vol. 13, no. 1, pp. 21–27, 1967.

[34] S. A. Dudani, "The distance-weighted k-nearest-neighbor rule," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 4, pp. 325–327, 1976.

[35] J. M. Keller, M. R. Gray, and J. A. Givens, "A fuzzy k-nearest neighbor algorithm," *IEEE transactions on systems, man, and cybernetics*, no. 4, pp. 580–585, 1985.

[36] S. Bermejo and J. Cabestany, "Adaptive soft k-nearest-neighbour classifiers," *Pattern Recognition*, vol. 33, no. 12, pp. 1999–2005, 2000.

[37] K. Taunk, S. De, S. Verma, and A. Swetapadma, "A brief review of nearest neighbor algorithm for learning and classification," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pp. 1255–1260, IEEE, 2019.

[38] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*, pp. 144–152, 1992.

[39] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 1999.

[40] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[41] J. Platt, N. Cristianini, and J. Shawe-Taylor, "Large margin dags for multiclass classification," *Advances in neural information processing systems*, vol. 12, 1999.

[42] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik, "Support vector clustering," *Journal of machine learning research*, vol. 2, no. Dec, pp. 125–137, 2001.

[43] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural processing letters*, vol. 9, no. 3, pp. 293–300, 1999.

[44] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.

[45] Y. Freund, R. E. Schapire, *et al.*, "Experiments with a new boosting algorithm," in *icml*, vol. 96, pp. 148–156, Citeseer, 1996.

[46] J. H. Friedman, "Stochastic gradient boosting," *Computational statistics & data analysis*, vol. 38, no. 4, pp. 367–378, 2002.

[47] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

[48] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, 2017.

[49] G. Matheron, "The intrinsic random functions and their applications," *Advances in applied probability*, vol. 5, no. 3, pp. 439–468, 1973.

[50] H. Liu, Y.-S. Ong, X. Shen, and J. Cai, "When gaussian process meets big data: A review of scalable gps," *IEEE transactions on neural networks and learning systems*, vol. 31, no. 11, pp. 4405–4423, 2020.

[51] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain.," *Psychological review*, vol. 65, no. 6, p. 386, 1958.

[52] S. Linnainmaa, "Taylor expansion of the accumulated rounding error," *BIT Numerical Mathematics*, vol. 16, no. 2, pp. 146–160, 1976.

[53] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[54] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[55] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An introduction to variational methods for graphical models," *Machine learning*, vol. 37, no. 2, pp. 183–233, 1999.

[56] O. Fabius and J. R. Van Amersfoort, "Variational recurrent auto-encoders," *arXiv preprint arXiv:1412.6581*, 2014.

[57] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," in *iclr*, 2017.

[58] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," *Advances in neural information processing systems*, vol. 28, 2015.

[59] K. Fukushima and S. Miyake, "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition," in *Competition and cooperation in neural nets*, pp. 267–285, Springer, 1982.

[60] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[61] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[62] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[63] T. H. Bryne, "Nonlinear observer design for aided inertial navigation of ships," 2017.

[64] R. Mahony, T. Hamel, P. Morin, and E. Malis, "Nonlinear complementary filters on the special linear group," *International Journal of Control*, vol. 85, no. 10, pp. 1557–1573, 2012.

[65] N. Hogben and F. Cobb, "Parametric modelling of directional wave spectra," in *Offshore Technology Conference*, OnePetro, 1986.

[66] D. Fathi, "Shipx vessel responses (veres), ship motions and global loads, users' manual," *Marintek AS, Trondheim. ShipX Vessel Responses (VERES)*, 2004.

[67] D. N. Veritas, "Dynamic positioning system–enhanced reliability dynpos-er," *DNV Guidance*, 2011.

[68] B. Yang, R. Liu, and E. Zio, "Remaining useful life prediction based on a double-convolutional neural network architecture," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9521–9530, 2019.

[69] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[70] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 807–814, 2010.

[71] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," in *2017 international joint conference on neural networks (IJCNN)*, pp. 1578–1585, IEEE, 2017.

[72] K. Knutsen, G. Manno, and B. Vartdal, "Beyond condition monitoring in the maritime industry," *DNV GL Strategic Research & Inovation Position Paper*, 2014.

[73] A. L. Ellefsen, V. Æsøy, S. Ushakov, and H. Zhang, "A comprehensive survey of prognostics and health management based on deep learning for autonomous ships," *IEEE Transactions on Reliability*, vol. 68, no. 2, pp. 720–740, 2019.

[74] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[75] A. L. Ellefsen, E. Bjørlykhaug, V. Æsøy, and H. Zhang, "An unsupervised reconstruction-based fault detection algorithm for maritime components," *IEEE Access*, vol. 7, pp. 16101–16109, 2019.

[76] P. Malhotra, A. Ramakrishnan, G. Anand, L. Vig, P. Agarwal, and G. Shroff, "Lstm-based encoder-decoder for multi-sensor anomaly detection," *arXiv preprint arXiv:1607.00148*, 2016.

[77] D. Park, Y. Hoshi, and C. C. Kemp, "A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1544–1551, 2018.

[78] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," *Special Lecture on IE*, vol. 2, no. 1, 2015.

[79] T. Kurtoglu, O. J. Mengshoel, and S. Poll, "A framework for systematic benchmarking of monitoring and diagnostic systems," in *2008 International Conference on Prognostics and Health Management*, pp. 1–13, IEEE, 2008.

[80] A. L. Ellefsen, S. Ushakov, V. Æsøy, and H. Zhang, "Validation of data-driven labeling approaches using a novel deep network structure for remaining useful life predictions," *IEEE Access*, vol. 7, pp. 71563–71575, 2019.

[81] L. Jayasinghe, T. Samarasinghe, C. Yuen, J. C. N. Low, and S. S. Ge, "Temporal convolutional memory networks for remaining useful life estimation of industrial machinery," *arXiv preprint arXiv:1810.05644*, 2018.

[82] H. Miao, B. Li, C. Sun, and J. Liu, "Joint learning of degradation assessment and rul prediction for aeroengines via dual-task deep lstm networks," *IEEE Transactions on Industrial Informatics*, vol. 15, pp. 5023–5032, Sep. 2019.

[83] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, Jan. 2014.

[84] X. Li, Q. Ding, and J.-Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering & System Safety*, vol. 172, pp. 1–11, 2018.

[85] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," in *2017 IEEE international conference on prognostics and health management (ICPHM)*, pp. 88–95, IEEE, 2017.

[86] A. L. Ellefsen, E. Bjørlykhaug, V. Æsøy, S. Ushakov, and H. Zhang, "Remaining useful life predictions for turbofan engine degradation using semi-supervised deep architecture," *Reliability Engineering & System Safety*, vol. 183, pp. 240–251, 2019.

[87] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.

[88] A. H. Brodtkorb, U. D. Nielsen, and A. J. Sørensen, "Sea state estimation using vessel response in dynamic positioning," *Applied Ocean Research*, vol. 70, pp. 76–86, 2018.

[89] U. D. Nielsen, "Estimation of directional wave spectra from measured ship responses," in *12th International Congress of the International Maritime Association of the Mediterranean: Maritime Transportation and Exploitation of Ocean and Coastal Resources*, pp. 1103–1112, 2005.

[90] G. Peeters, B. L. Giordano, P. Susini, N. Misdariis, and S. McAdams, "The timbre toolbox: Extracting audio descriptors from musical signals," *The Journal of the Acoustical Society of America*, vol. 130, no. 5, pp. 2902–2916, 2011.

[91] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[92] X. Cheng, G. Li, R. Skulstad, S. Chen, H. P. Hildre, and H. Zhang, "Modeling and analysis of motion data from dynamically positioned vessels for sea state estimation," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 6644–6650, IEEE, 2019.

[93] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv preprint arXiv:1406.2661*, 2014.

[94] U. D. Nielsen, "Introducing two hyperparameters in bayesian estimation of wave spectra," *Probabilistic Engineering Mechanics*, vol. 23, no. 1, pp. 84–94, 2008.

[95] A. Gepperth and B. Hammer, "Incremental learning algorithms and applications," in *European symposium on artificial neural networks (ESANN)*, 2016.

[96] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54–71, 2019.

[97] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[98] I. Lage, A. S. Ross, B. Kim, S. J. Gershman, and F. Doshi-Velez, "Human-in-the-loop interpretability prior," *Advances in neural information processing systems*, vol. 31, 2018.

Appendix

A

Paper I

B

Paper II

C

Paper III

D

Paper IV

E

Paper V

This paper is not included due to copyright restrictions
available at publisher site https://doi.org/10.1109/JSEN.2021.3119151and
Institutional archive
https://hdl.handle.net/11250/2990184

F

Paper VI

Marine Structures 83 (2022) 103159

Contents lists available at ScienceDirect

Marine Structures

journal homepage: www.elsevier.com/locate/marstruc

Directional wave spectrum estimation with ship motion responses using adversarial networks

Peihua Han [a,*], Guoyuan Li [a], Stian Skjong [b], Houxiang Zhang [a,*]

[a] Department of Ocean Operations and Civil Engineering, Norwegian University of Science and Technology (NTNU), 6009 Aalesund, Norway
[b] SINTEF Ocean, 7010 Trondheim, Norway

ARTICLE INFO

Keywords:
Wave spectrum estimation
Generative adversarial networks
Ship intelligence
Decision support

ABSTRACT

The external environmental conditions around a vessel are essential for efficient and safe ship operation, among which the sea state is of key importance. Considering the ship as a large wave buoy, the sea state can be estimated from motion responses without extra sensors installed. This is a challenging task since the relationships between the waves and the ship motions are hard to describe accurately. Machine learning approaches can learn these mapping without an explicit model, which is promising for sea state estimation. Current machine learning approaches represent the sea state as a set of categories or a number of wave parameters while neglecting the 2D wave spectrum. This paper proposes a sea state estimation network that estimates the 2D wave spectrum along with a discrimination network. The discrimination network can detect and correct high-order inconsistencies of the spectrum. Simulation studies are performed to show that the proposed method can provide wave spectrum estimation with high accuracy.

1. Introduction

Environmental conditions are of key importance for efficient and safe ship operations. The external wave conditions are one of the crucial factors affecting the dynamics of a vessel. The continuous sea state information around a ship are valuable for providing onboard decision supports and operational guidance, including takeoff and landing of helicopters, crane operations. By incorporating knowledge about sea states, the safety of the operations can be increased and even more efficient. Therefore, in-situ sea state estimation is important for any type of decision support and system with high level of autonomy.

In oceanography, the general condition of the ocean with respect to wind waves and swell at a certain location is referred to as the sea state. The waves are stochastic with time and it is almost impossible to evaluate on a wave-by-wave basis in the time domain [1]. The ocean waves are considered to be a stochastic process and their statistical properties can be evaluated in the frequency domain. Specifically, the potential and kinematic energies of stochastic waves are represented by the wave spectrum.

Nowadays, the primary tool for collecting accurate ocean wave statistics is floating wave buoys. However, They are not practical for a vessel in maneuvering operation since they are fixed at a specific location. Meteorological satellite can also provide wave statistics, but the resolution is often poor. The x-band wave radar provides in-situ wave spectrum, but it is expensive to install, requires frequent calibration [2], and is yet only equipped on a limited number of vessels. Similar to the wave buoy, the motion responses of a vessel reflect the sea state conditions and therefore a vessel can also be considered as a large wave buoy. The majority of marine vessels today are equipped with sufficient sensors that measure the ship motion in 6 degrees of freedom. Therefore, a vessel is essentially equipped with an environmental condition estimation system [3].

* Corresponding authors.
E-mail addresses: peihua.han@ntnu.no (P. Han), guoyuan.li@ntnu.no (G. Li), Stian.skjong@sintef.no (S. Skjong), hozh@ntnu.no (H. Zhang).

https://doi.org/10.1016/j.marstruc.2022.103159
Received 18 May 2021; Received in revised form 10 December 2021; Accepted 8 January 2022
Available online 1 February 2022

Estimating the sea state based on ship motions has been a topic of interest in the literature. This task is challenging due to the operation of the vessel, as well as the inaccurate relationship between waves and the ship motions. Ship responses, in principle, are non-linearly related to wave excitation. Previous methods usually rely on the response amplitude operators (RAOs) to relate the waves and the ship motions. RAOs are usually calculated by linearizing the results from strip theory or computational fluid dynamics and therefore only valid for light and moderate sea states [4]. In addition, RAOs might need to be tuned with real-world data. Another possible solution is to treat the task as a supervised machine learning problem. The fundamental idea is to learn the mapping from measured ship motion responses to the actual sea state from historical data. The advantage of data-driven methods is that it does not require specific knowledge of the vessels to discover the pattern between ship motions and sea states.

Sea state estimation with ship motion responses based on machine learning approaches is usually regarded as a classification or regression task. The sea state is predefined as multiple categories [5] or represented by several integrated wave parameters [6], e.g., significant wave height and peak period. Pre-defining the sea state categories might be problematic since it is difficult to use limited categories to cover all possible sea states. The resolution of the estimation results might also be too low for practical use. The integrated wave parameters are a summary expression of the wave spectrum. These two methods, either classification or regression, only provide limited information on the sea state. Ideally, a 2D directional wave spectrum could be estimated to fully describe the sea state. In addition, the 2D directional wave spectrum is fundamental for operational safety analysis such as extreme value analysis.

In such a context, this work aims to build a machine learning model for estimating the 2D directional wave spectrum using ship motion responses. The proposed model follows the generative adversarial networks [7] architecture. Two separate deep convolutional neural networks, an estimation network, and a discrimination network are established. The estimation network uses the ship motion as input and estimates 2D wave spectrum. The discrimination network tries to classify the 2D wave spectrum as real or fake. In this way, an adaptive loss is learned and unrealistic wave spectrum will not be tolerated. Simulation studies show that the proposed method can provide estimates of wave spectrum based on ship motions. To the best of our knowledge, it is the first time that an adversarial network is used in sea state estimation. The main contributions of this paper are highlighted as follows:

- A novel model is developed to estimate the 2D directional wave spectrum using the measured ship motion responses. It can estimate a wide range of sea state conditions.
- Extensive simulation studies are performed to validate the proposed method and comparison with model-based method is made.
- The proposed model performs well in estimating different types of spectra and is robust regarding noisy measurements.

The remainder of this paper is organized as follows: A literature review on sea state estimation using ship motion responses is given in Section 2. The proposed adversarial neural network is introduced in Section 3. The experimental setup and experiment are discussed in Section 4. Section 5 concludes the paper.

2. Literature review

Estimating the sea state information based on the motion responses has been investigated in the literature. Previous works differ in whether the estimation problem is formulated in the frequency domain or time domain. In the frequency domain solution, the time series motion responses are first transformed into the frequency domain through fast Fourier transform or autocorrelation analysis. The RAOs are used to relate the wave spectrum to the motion spectrum. To obtain the wave spectrum, the fundamental idea is to minimize the difference between the measured ship spectrum and the calculated ship spectrum [8]. A wave spectra, e.g., JONSWAP, Bretschneider with the cos^{2s} spreading model, can be assumed. In this way, a nonlinear optimization process is formed, the wave parameters in the hypothetical wave spectrum can be obtained through optimization techniques [6,9]. This method is computationally intensive and may not converge since the objective function is nonlinear and non-convex. A non-parametric approach, in which the wave spectrum is represented in a discrete frequency-directional domain, can also be applied. The problem is an ill-posed problem and therefore different kinds of prior are used, e.g., the smoothness of wave spectrum [10,11] and the sparsity of wave spectrum [11]. These methods can be extended to ships with forward speed by incorporating the Doppler shift function [10]. The effectiveness of this method is shown with a container ship [12].

For the time domain solution, the focus is on real-time sea state updates obtained from continuous response measurements. A framework based on the Kalman filter is established [13,14], in which an irregular wave represented as a number of regular waves. In the Kalman filter framework, the amplitude and frequency of the regular waves are treated as states. The waves are considered constant between two discrete time intervals. A similar second-order nonlinear observer is developed to estimate the frequency of wave [15]. In addition, the optimization can be performed directly in the time domain [16]. However, the latter two approaches can only estimate a single sinusoid wave.

The above methods are model-based approaches that require a model to relate the wave and the ship motion. Machine learning is another solution that learns that mapping from measured ship motion responses to the sea state. The sea states are usually predefined into various categories [5,17] or represented as several integrated wave parameters [18,19] depending on whether this task is formulated as a classification task or a regression task. Various machine learning models, e.g., multi-layer perceptron, Gaussian process, deep learning models, have been utilized. However, these methods cannot provide a detailed 2D wave spectrum that is usually required in practical applications. [20] estimated the 2D wave spectrum using convolutional neural network. The problem is still considered as a regression problem. They estimated 8 parameters of the Ochi–Hubble-type spectrum from the neural network, and then reconstructed the 2D wave spectrum. In this paper, no specific form of wave spectrum is assumed. The focus of this paper is to bridge the gap by developing a machine learning model that estimates the 2D wave spectrum directly without assuming the structure of wave spectrum.

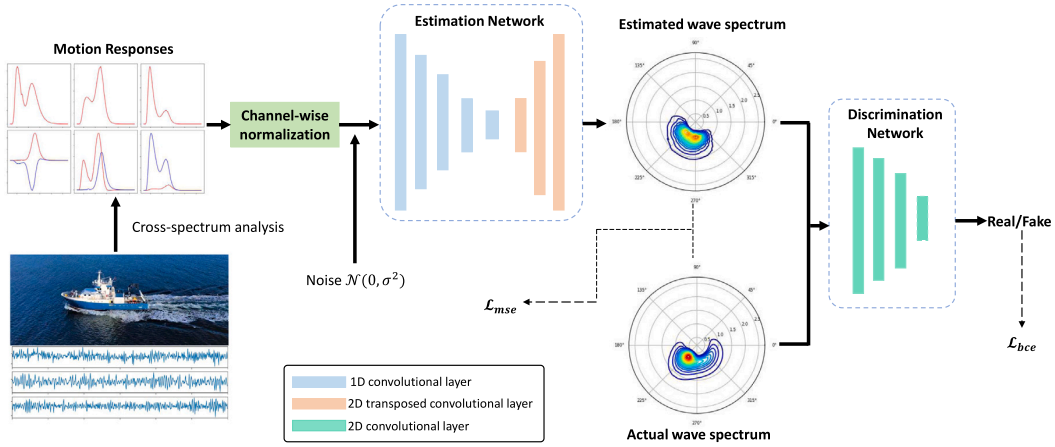


Fig. 1. Schematic illustration of the proposed model for 2D directional wave spectrum estimation using ship motion responses.

3. Methodology

The proposed method consists of two separate networks, as outlined in Fig. 1. The inputs are the cross spectrum of the ship motion. The cross spectrum is normalized and then fed into the estimation network to be converted into a 2D wave spectrum. In this paper, no specific form of wave spectrum is assumed, and the output 2D wave spectrum from the estimation network is represented as a 36 by 100 matrix. In other words, there are 36 discrete directions and 100 discrete frequencies. The discrimination network takes a 2D wave spectrum as input and distinguishes whether it is generated from the estimation network or is the actual wave spectrum. In the training phase, the estimation network tries to generate a realistic wave spectrum while the discrimination network tries to distinguish it. In this way, the two networks are improved together and the high-order statistics of the output wave spectrum are penalized to force the estimation network to provide continuous and realistic results. At inference time, the discrimination network is omitted, and the estimation network is used to output the estimated 2D wave spectrum from ship motion responses.

3.1. Channel-wise normalization

Since the input for the proposed network is the cross-spectrum of the ship motion, the cross-spectrum is assigned into different channels to form multi-channel 1D inputs. The inputs are then normalized to the range [0, 1] with respect to its channel. Specifically, each channel (each component of the cross-spectrum) maintains its statistics and it is normalized individually.

3.2. Data augmentation with noise

Data augmentation is a technique for improving the robustness and training of neural networks. The idea is to simulate various expected variations in the datasets by manipulating the training samples. Since the inputs for the proposed estimation network is a spectral representation of the motion responses, the augmented spectral signal is formulated as follow:

$$\begin{aligned} P_n &= P + P \odot \alpha \\ \alpha &\sim \mathcal{N}(0, \sigma^2) \\ \sigma &\sim \mathcal{U}(0, 0.1) \end{aligned} \tag{1}$$

where P_n and P are the augmented and original spectrum, respectively. $\odot$ is the element-wise Hadamard product. $\mathcal{N}$ denotes the normal distribution while $\mathcal{U}$ denotes the uniform distribution. In this approach, the original spectrum is augmented randomly in each training epoch since the noise level σ^2 is drawn from a distribution. The noise added also depends on the value of the spectrum. Fig. 2 shows two examples of the augmented spectrum.

3.3. Network architectures

Estimation network. The proposed estimation network follows an encoder–decoder structure. In the network, the input is passed through a series of 1D convolution layers that progressively downsample, to a bottleneck layer, then the process is reversed, and upsampling is achieved by a series of transposed 2D convolution layers. In this way, the network takes the 1D data as inputs and outputs a 2D wave spectrum. The network uses modules in the form of convolution-BatchNorm-ReLu. The ResNet block [21] is used

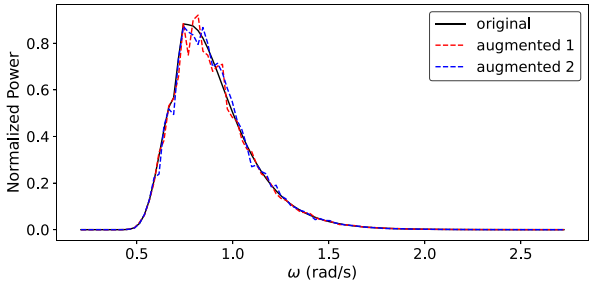


Fig. 2. Examples of augmentation on the spectral inputs.

in this network since it provides better performance in many applications. For the output layers, the Sigmoid activation function is applied.

Discrimination network. The discrimination network follows a convolutional neural network structure, in which the modules in the form of convolution-BatchNorm-LeakyReLu are used. The LeakyReLu activation function is used since it can stabilize the training [22].

Details of the architectures of the estimation network and discrimination network are presented in Appendix.

3.4. Adversarial training

A hybrid loss which is a weighted sum of two terms is used. The first is the mean square error that encourages the estimation model to predict the wave spectrum. The second loss term is based on the adversarial convolutional network. This loss term is large if the adversarial network can discriminate the output of the estimation network from the actual wave spectrum. The aim of the adversarial term is to penalize mismatches in the high-order spectral power value statistics, e.g., the power value of wave spectrum should be smooth in the near region, which is not accessible by the mean square loss function.

Given a training ship motion responses x and a corresponding wave spectrum y, the estimator E and the discriminator D would be competed in a two-player min–max optimization routine:

$$\min_E \max_D \mathcal{L}(E, D) = \mathcal{L}_{mse}(E(x), y) - \lambda[\mathcal{L}_{bce}(D(y), 1) + \mathcal{L}_{bce}(D(E(x)), 0)] \tag{2}$$

where $\mathcal{L}_{mse}$ is the mean square loss, $\mathcal{L}_{mse}(\hat{z}, z) = |\hat{z} - z|^2$. $\mathcal{L}_{bce}$ is the binary cross-entropy loss, $\mathcal{L}_{bce}(\hat{z}, z) = -z \log \hat{z} - (1 - z) \log(1 - \hat{z})$. λ is a hyperparameter to balance these two different losses.

The training of the estimation model minimizes the mean square error loss while at the same time trying to fool the discriminator model. The objective function of the estimation model is:

$$\mathcal{L}_E = \mathcal{L}_{mse}(E(x), y) - \lambda \mathcal{L}_{bce}(D(E(x)), 0) \tag{3}$$

In practice, the term $-\mathcal{L}_{bce}(D(x, E(x)), 0)$ is replaced by $+\mathcal{L}_{bce}(D(x, E(x)), 1)$ [7]. This means that the probability that the adversarial model predicts the estimated wave spectrum to be the actual one is maximized, instead of minimizing the probability that the adversarial model predicts the estimated wave spectrum to be synthetic.

For the adversarial model, only the binary classification loss is related. Therefore, training the adversarial model is equal to minimizing the following objective function:

$$\mathcal{L}_D = \mathcal{L}_{bce}(D(y), 1) + \mathcal{L}_{bce}(D(E(x)), 0) \tag{4}$$

3.5. Implementation details

The proposed model is implemented in Pytorch. To optimize the proposed network, we alternate between one gradient descent step on E, then one step on D. The Adam solver [23] with minibatch is used to minimize the objective function for E and D. The minibatch size is set as 256 in the training procedure. For the estimation network E, a learning rate of 1×10^{-4} with l_2 regularization term of 1×10^{-3} is used. For the discriminating network D, a learning rate of 1×10^{-5} with l_2 regularization term of 1×10^{-3} is used. The hyperparameter λ is set as 0.01 to balance the losses.

4. Experimental setup

4.1. Data

The wave spectrum-ship motion pairs are generated from simulations. In the simulations, a double-peak wave spectrum [24] is adopted since it covers a wide range of possible spectrum shapes and it models both the wind waves and the swell waves. The

Table 1
Sampling range for the wave spectrum parameters ($i = 1, 2$).

$H_{s,i}$	$\omega_{m,i}$	$\theta_{m,i}$	s	λ_i
[0.5, 4]	[(1/8)π, (2/5)π]	[0, 2π]	[1, 26]	[0.8, 1.5]

directional wave spectrum is given by:

$$E_g(\omega,\theta) = \frac{1}{4}\sum_{i=1}^{2}\frac{\left(((4\lambda_i+1)/4)\omega_{m,i}^4\right)^{\lambda_i}}{\Gamma(\lambda_i)}\frac{H_{s,i}^2}{\omega^{4\lambda_i+1}}A(s_i)\times\cos^{2s_i}\left(\frac{\theta-\theta_{m,i}}{2}\right)\exp\left[-\frac{4\lambda_i+1}{4}\left(\frac{\omega_{m,i}}{\omega}\right)^4\right] \tag{5}$$

where H_s is the significant wave height, θ_m is the mean wave direction and ω_m is the model angular frequency. s and λ are two shape parameters. Γ denotes the Gamma function. The function $A(s)$ is defined as:

$$A(s) = \frac{2^{2s-1}\Gamma^2(s+1)}{\pi\Gamma(2s+1)} \tag{6}$$

Note that the above wave spectrum model $E_g(\omega,\theta)$ is only used to generate the simulation data for this study and will not be used in our estimation network model. NTNU's research vessel R/V Gunnerus with a length between perpendiculars of 28.9 m, a breadth of 9.6 m, and a draught of 2.7 m is used as the example vessel [25]. The complex-valued response amplitude operators (RAOs) of the vessel are obtained from ShipX [26]. The ship motion cross-spectra is then calculated as:

$$S_{ij}(\omega) = \int_{-\pi}^{\pi}\Phi_i(\omega,\theta)\overline{\Phi_j(\omega,\theta)}E_g(\omega,\theta)d\theta \tag{7}$$

where $\Phi(\omega,\theta)$ is the complex-value transfer function and $\overline{\Phi(\omega,\theta)}$ is its complex conjugate.

In this study, the wave spectrum is discretized into a 36 × 100 grid after generating from Eq. (5), where 36 different headings with interval of 10° and 100 angular frequencies from 0.2 rad/s to 3 rad/s is considered. It is equal to the output wave spectrum shape from our estimation network, and therefore validation can be easily performed. Three corresponding ship motions, *sway velocity*, *pitch*, *heave*, are used. This results in 9 power spectra (6 real part and 3 imaginary part) and therefore the size of response spectrum is 9 × 100. The used wave spectrum consists of 10 parameters $[H_{s,1}, \omega_{m,1}, \theta_{m,1}, s_1, \lambda_1, H_{s,2}, \omega_{m,2}, \theta_{m,2}, s_2, \lambda_2]$. These parameters are sampled randomly to generate 1000 different wave spectrum, the sampling range is described in Table 1. Note that s is an integer. The corresponding ship motion cross spectrum is then calculated, forming a dataset with 1000 wave spectrum-ship motion pairs. The dataset is then divided into 500 as training set and the rest 500 as test set. The reason why 500 samples are used in the test set is because these samples can cover the wave space of interest.

4.2. Time series generation

Ship motions, in principle, are measured in the time domain. To generate time series of ship motions under a specific wave spectrum, we follow the procedure in [27]. The time-domain ship motion response $R(t)$ can be expressed as follow:

$$\begin{aligned} R(t) &= \sum_{n=1}^{N}\sum_{m=1}^{M} a_{mn}|\Phi(\omega_m,\theta_n)|\cos\left(\omega_m+\epsilon_{mn}\right) \\ a_{mn} &= \sqrt{2E(\omega_m,\theta_n)\Delta\omega_m\Delta\theta_n} \\ \epsilon_{mn} &= \arctan\left(\frac{\Im[\Phi(\omega_m,\theta_n)]}{\Re[\Phi(\omega_m,\theta_n)]}\right) \end{aligned} \tag{8}$$

where M is the discrete number of wave frequencies and N is the discrete number of headings. ϕ denotes the complex transfer function and E is the wave spectrum. $\Delta\omega_m$ and $\Delta\theta_n$ are the increments of the discrete wave frequencies and the discrete headings. It is noteworthy that for an equidistant frequency discretization, the time series response $R(t)$ will repeat itself after a period of $2\pi/\Delta\omega$. A simple way to handle this problem is to use non-equidistant frequency discretization:

$$\omega_{i+1} = \omega_i + c\cdot p_i \tag{9}$$

where c is a small factor and it is chosen as 0.01 while p_i is a stochastic variable with values between 0 and 1. We generate 1800 s long time series responses for sway velocity, pitch, and heave.

To simulate the noisy measurements, Gaussian white noise is then added to the time series motion response. The signal-to-noise ratio (SNR) is used in this study to measure the noise level. The SNR is defined in Eq. (10), where σ_{signal} and σ_{noise} is the standard deviation of the measured motion response and noise, respectively.

$$SNR = \frac{\sigma_{signal}^2}{\sigma_{noise}^2} \tag{10}$$

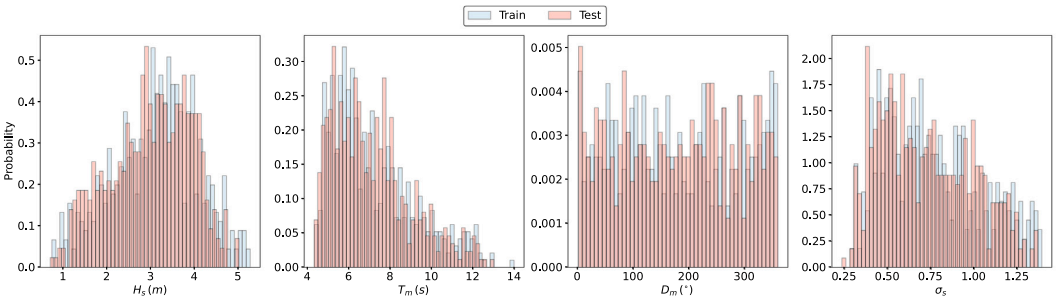


Fig. 3. Distribution of the integrated wave parameters in the generated dataset.

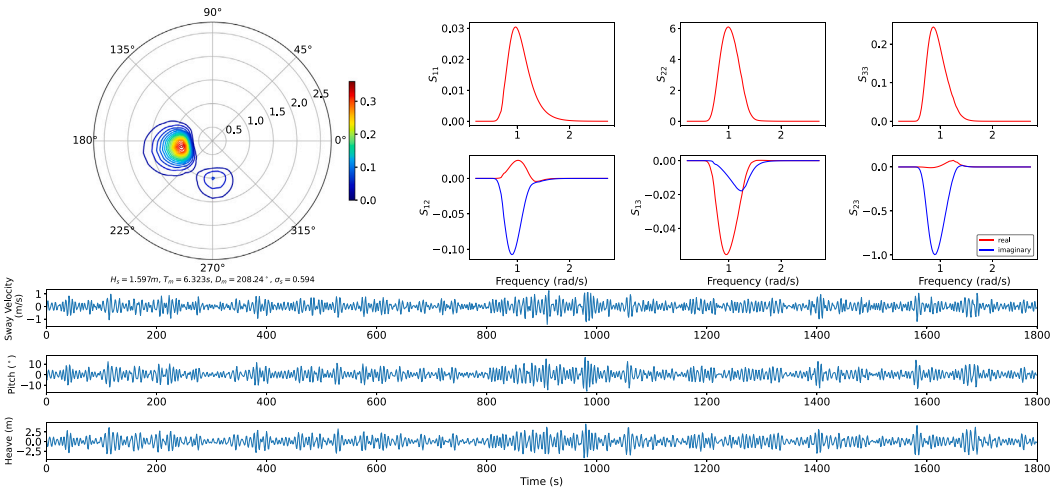


Fig. 4. A sample from the generated dataset. The left upper graph is the 2D wave spectrum and its integrated wave parameters. The right upper graphs are the cross spectrum of motion responses. The lower three graphs are the time series data of these three ship motions.

4.3. Integrated wave parameters

The overall outcome of the proposed model is given by a directional wave spectrum $E(\omega, \theta)$. For comparison, the integrated wave parameters are then evaluated. The spectral moment of order n is defined as [28]:

$$m_n = \iint \omega^n E(\omega, \theta) d\omega d\theta \tag{11}$$

Thus, the significant wave height H_s and the mean wave period T_m can be calculated as follows:

$$\begin{aligned} H_s &= 4\sqrt{m_0} \\ T_m &= m_{-1}/m_0 \end{aligned} \tag{12}$$

The mean wave direction D_m and the mean directional spread σ_s is given by:

$$\begin{aligned} D_m &= \arctan(d/c) \\ \sigma_s &= \left(2 - \frac{2}{m_0}\sqrt{d^2 + c^2}\right)^{0.5} \end{aligned} \tag{13}$$

where d and c are defined as:

$$\begin{aligned} d &= \iint E(\omega, \theta) \sin\theta d\omega d\theta \\ c &= \iint E(\omega, \theta) \cos\theta d\omega d\theta \end{aligned} \tag{14}$$

Table 2
MAE of different methods on the test set.

Methods	Pixel	Integrated wave parameters			
		H_s (m)	T_m (s)	D_m (°)	σ_s
WBA	0.033	0.606	0.573	**12.88**	0.234
Proposed w/o AT	0.043	1.265	0.952	18.80	0.353
Proposed w AT	**0.018**	**0.239**	**0.361**	13.95	**0.153**

The mean directional spread σ_s is a parameter representing the spread of the spectrum. Specifically, σ_s decreases as the shape parameter s increase in the cos2s spreading function. The smaller the σ_s, the directional spread is broader. The wave spreads equally in all directions when σ_s is close to 1.4.

4.4. Description on the generated data

Fig. 3 shows the distribution of the integrated wave parameters of the generated dataset. The significant wave height H_s ranges from around 0.7 m to 5.3 m. The mean wave period T_m is around 2 s to 14 s while the mean directional spread σ_s is around 0.2 to 1.4. The mean wave direction D_m is distributed uniformly from 0° to 360°. This dataset covers a wide range of sea states that the vessel might encounter in the real world.

Fig. 4 presents a sample from the dataset. The sea state is described as a 2D wave spectrum. The integrated wave parameters H_s, T_m, D_m, σ_s are the summation of the 2D wave spectrum. The cross spectrum of motion responses as well as the time series of the ship motion is presented. In the cross spectrum, the subscripts 1, 2, 3 denotes sway velocity, pitch, heave, respectively. The cross spectrum of motion responses will be used as the input and the target is to estimate the 2D wave spectrum.

4.5. Evaluation metrics

To evaluate and compare the performance of the proposed model, the mean absolute error (MAE) is used:

$$MAE = \frac{1}{k}\sum_{i=1}^{k}|\hat{y}_i - y_i| \tag{15}$$

where k is the number of samples, $\hat{y}$ and y is the estimated and actual value, respectively. In this paper, the MAE of the discrete wave spectrum and the MAE of the integrated wave parameters are evaluated. For abbreviation, the MAE of the wave spectrum is referred to as the pixel error in the rest of the paper. For mean wave direction, Eq. (15) is modified into $MAE = \frac{1}{k}\sum_{i=1}^{k}\min\left(|\hat{y}_i - y_i|, 360 - |\hat{y}_i - y_i|\right)$ to consider that 0° and 360° are the same.

5. Experimental results

In this section, the performance of the proposed method will be evaluated. Two baseline models are implemented for comparison:

- **Bayesian wave buoy analogy method**: This method is a model-based method for directional wave spectrum estimation using ship motion responses. The wave spectrum is represented in a discrete frequency-directional domain. The fundamental idea is to minimize the difference between the measured and the calculated spectrum. However, this forms an ill-posed inverse problem, and therefore smooth prior is introduced to solve the problem in the Bayesian framework. In this paper, a two hyperparameters method [29] is used. The two hyperparameters are responsible for the smooth prior of wave spectrum in the discrete frequency and discrete direction, respectively. Details of this method is described in [29].
- **Neural network model without adversarial training**: This model is the estimation network proposed in this paper. The discriminator network is neglected by setting the hyperparameter λ as 0. This model is implemented to show the effect of adversarial training.

In the following, the Bayesian wave buoy analogy method is denoted as "WBA", the neural network model without adversarial training is denoted as "Proposed w/o AT", and the proposed neural network model with adversarial training is denoted as "Proposed w AT".

5.1. Experiment with perfect response spectrum

In this part, the perfect measured response cross spectrum is used for validation. Fig. 5 presents the estimated directional wave spectrum from three random samples in the test set. The colors of values larger than the color bar upper limits remain the same as that of the upper limit. It is shown that the Bayesian WBA method provides a similar shape of the spectrum as the actual ones but the values are less accurate. The reason is that the performance of this method depends on the two hyperparameters and the initial guess of the wave spectrum. In this paper, several combinations of hyperparameters and initial guesses are used to yield the best-estimated spectrum. For the neural network model, the model without adversarial training clearly presents spurious lines in

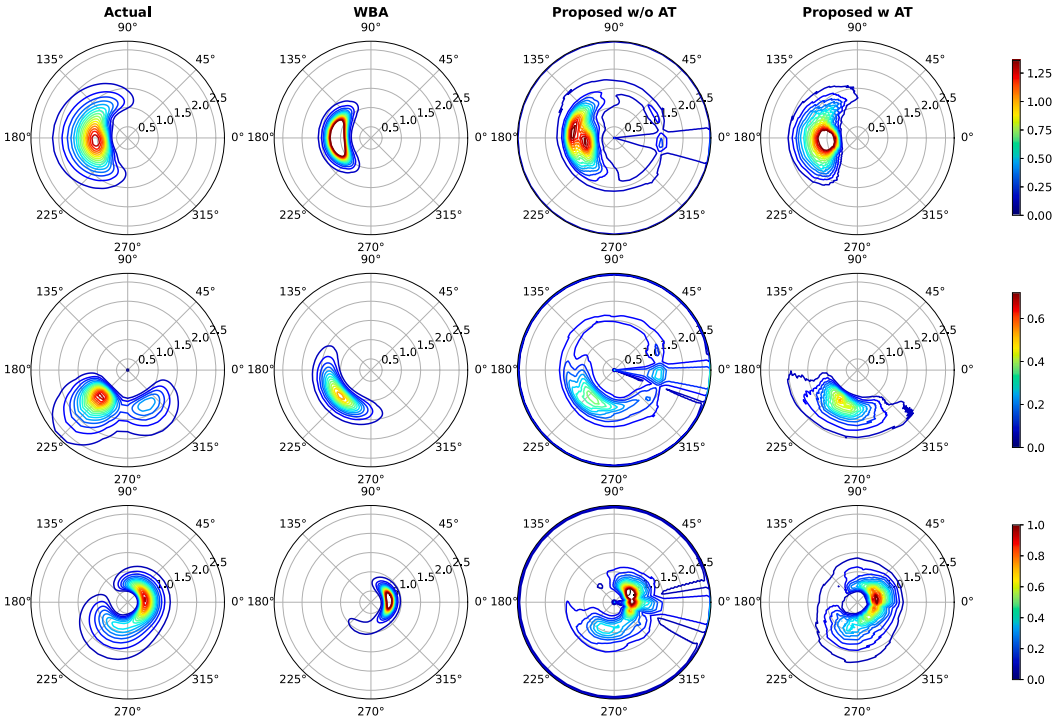


Fig. 5. Examples of contour plots of the estimated directional wave spectrum based on perfect motion spectrum.

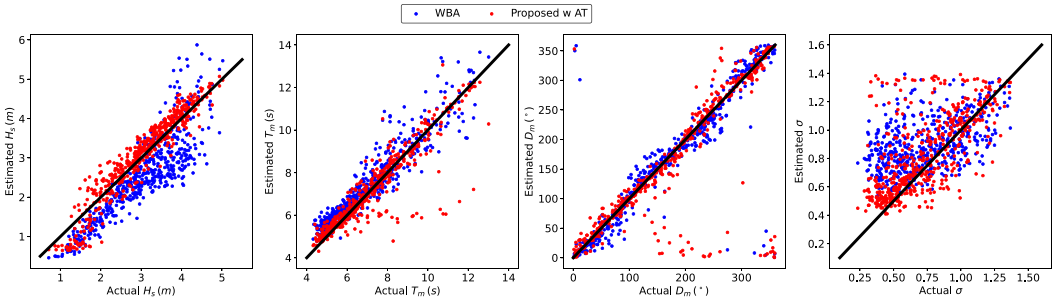


Fig. 6. Actual and estimated integrated wave parameters for perfect response spectrum.

the wave spectrum. Even though the shape of the estimated wave spectrum is similar to the actual wave spectrum, it has high total wave energy. The model with adversarial training better enforces the spatial consistency of the wave spectrum. It also smooths and strengthens the high energy density area of the wave spectrum.

Table 2 summarizes the overall performance in terms of MAE. Compared with the neural network model without adversarial training, the error of the WBA method in terms of pixel-level and integrated wave parameters is relatively low. By incorporating adversarial training, these errors are reduced significantly. In this comparison, our model with adversarial training has the smallest error.

Fig. 6 shows the correlation between the actual and estimated integrated wave parameters of the test data. The black line denotes that the estimated parameter is equal to the actual one. It is observed that both methods provides relatively accurate results. The WBA tends to provide lower estimated H_s than the actual one and it is not that accurate for σ. The proposed method with adversarial training provides more accurate estimation in terms of H_s and σ. However, the proposed network have low variability in terms of estimating T_m and D_m for most samples, some of which are quite different from actual estimates.

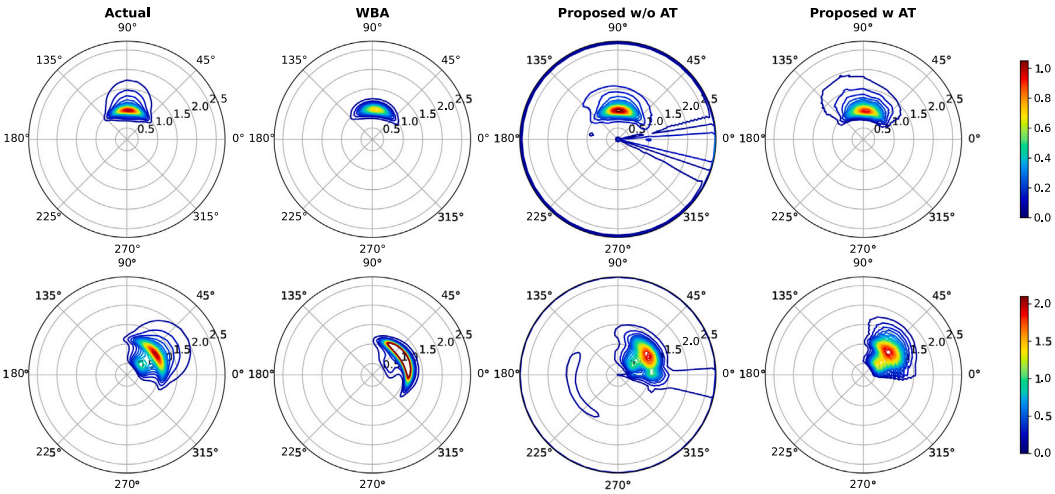


Fig. 7. Examples of contour plots of the estimated directional wave spectrum for JONSWAP-type wave spectrum.

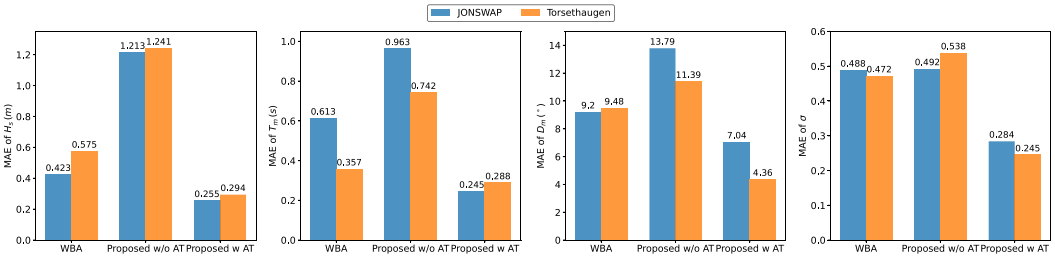


Fig. 8. MAE of the integrated wave parameters for the JONSWAP and Torsethaugen wave spectrum.

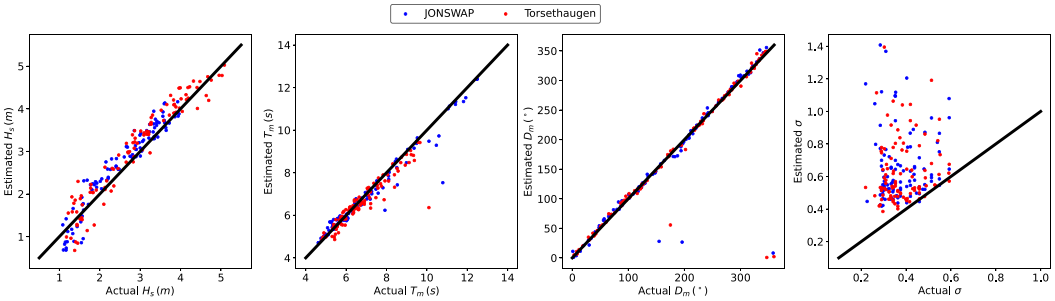


Fig. 9. Actual and estimated integrated wave parameters for JONSWAP and Torsethaugen spectrum (proposed w AT).

5.2. Generalization to JONSWAP-type wave spectrum

As presented in Section 4.1, the training data is generated through a double Pierson–Moskowitz type wave spectrum. This type of spectrum might not cover the possible wave spectrum. Therefore, the zero-shot learning ability of this model to other types of wave spectrum is investigated.

In this part, the generalization ability of the model is evaluated with the JONSWAP-type wave spectrum. The JONSWAP type spectrum has a more pronounced peak in the spectrum than the Pierson–Moskowitz (PM) type wave spectrum. The JONSWAP wave spectrum and the Torsethaugen wave spectrum (a double peak JONSWAP-type spectrum) are used to generate two extra test sets with 100 samples, respectively. The trained model is then used to estimate the 2D wave spectrum. Fig. 7 shows the estimated 2D

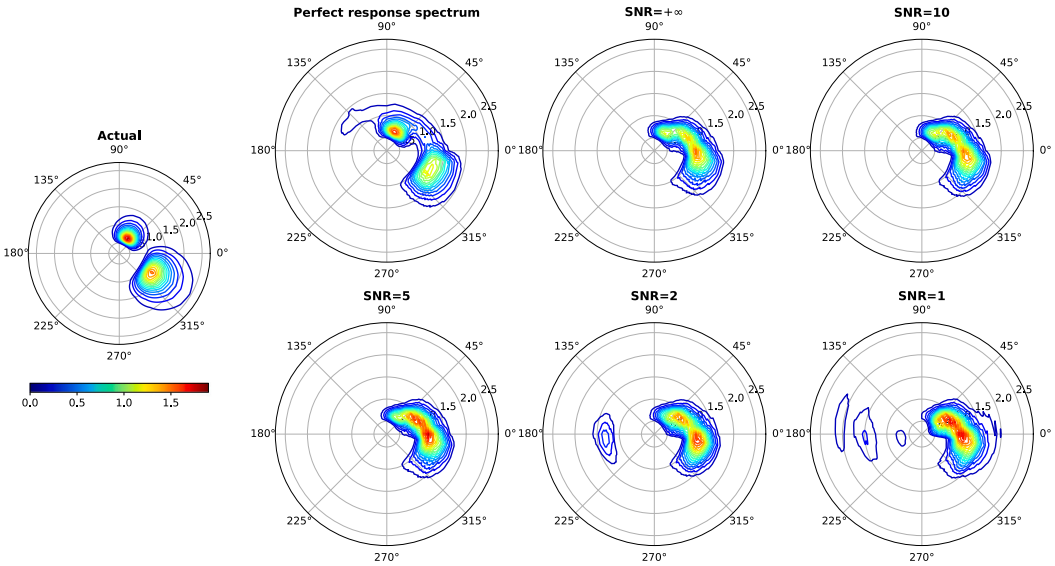


Fig. 10. Examples of contour plots of the estimated directional wave spectrum with different SNR levels.

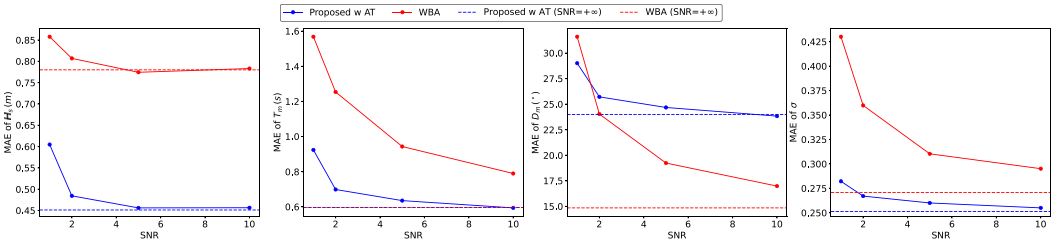


Fig. 11. MAE of the integrated wave parameters for the motion responses under different SNR levels.

wave spectrum from two examples in the two extra test sets, respectively. The proposed model presents a less narrow spectrum than the actual one, which might be due to the Pierson–Moskowitz type wave spectrum used in the training data. Nonetheless, the proposed model still provides a reasonable estimate.

Fig. 8 summarizes the MAE of H_s, T_m, D_m, σ for the JONSWAP and Torsethaugen wave spectrum. The proposed model achieves the lowest deviation among these three methods. It demonstrates that the proposed model successfully captures the relation between ship motion and wave spectrum, therefore, it is able to estimate the type of wave spectrum not present in the training data.

Fig. 9 shows that correlation of actual and estimated integrated wave parameters from JONSWAP and Torsethaugen wave spectrum. It is shown than the proposed model provides accurate estimation in terms of H_s, T_m, and D_m for both wave spectrum. However, the model gives higher σ than the actual one. The reason might be that for the training data samples a broader range of directional spreading functions than the test data here. Specifically, the s parameter in the cos2s spreading function is sampled in the range of [1, 26] for the training data while [5, 26] for the JONSWAP-type spectrum, which results in a smaller range of σ. The model cannot adjust to the distribution shift since it is in zero-shot setting.

5.3. Effect of noisy ship motion measurement

Ship motions are measured in the time domain. In order to use the proposed approach, the ship motion in the time domain must be transformed into the frequency domain through cross spectrum analysis. The cross spectrum analysis typically is performed through fast Fourier transform or multivariate autoregressive modeling, which would inevitably introduce a certain deviation from the actual motion response spectrum. In addition, noise in the measured ship motion would introduce a certain degree of error. In this section, the effect of cross spectrum analysis and the noises in ship motion on the estimated results will be evaluated. The cross spectrum analysis in this paper is performed through the Welch method. White noise is added and four different SNR levels, 10, 5,

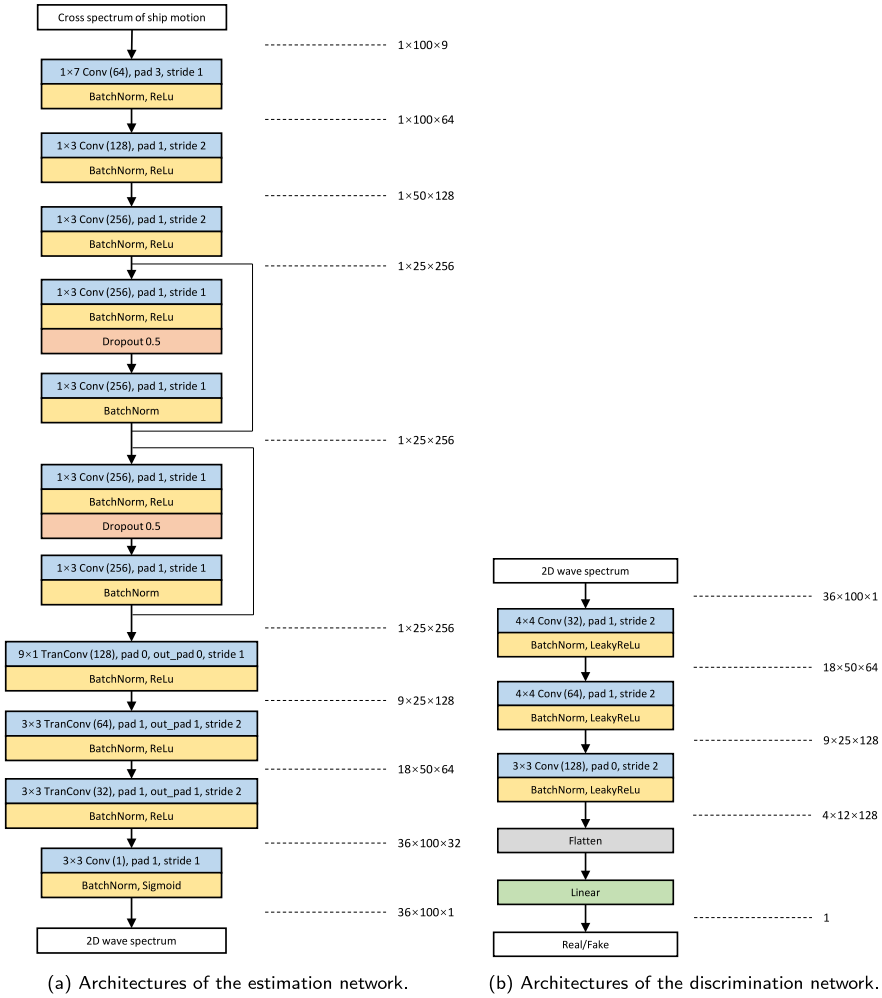


Fig. A.1. Architectures of the estimation and discrimination network.

2, 1, are investigated. For simplification, the time series ship motion without noise added is denoted as "SNR=+∞". In "SNR=+∞", only the effect of cross spectrum analysis is included.

Fig. 10 presents an example of an estimated 2D wave spectrum under different SNR levels. From the estimates for the perfect response spectrum and SNR=+∞, the power value and spectral shape are changed due to the cross spectrum analysis. As the SNR level decreases, the quality of the estimates, usually but not definitely, also decreases. In general, the estimated 2D wave spectrum is relatively close to the actual wave spectrum.

Fig. 11 compares the MAE of integrated wave parameters in WBA and the proposed model under different SNR levels. The proposed model is less sensitive to noise than the WBA. The WBA method shows low error in D_m while the proposed model has low error in H_s, T_m and σ.

6. Conclusion

Estimating the sea state based on the measured ship motion response is a complicated and arduous task. Previous machine learning approaches cannot capture the directional wave spectrum. This paper presents an estimation network and discriminant network based on convolutional neural networks. The high-order inconsistencies of the wave spectrum from the estimation network are penalized by the estimation network, thereby forcing the estimation network to produce accurate and realistic results. Simulation studies show that the proposed model guarantees the smoothness of the wave spectrum and provides accurate estimation results. The

generalizability of the method is demonstrated by estimating the JONSWAP-type spectrum that is not in the training set. Comparison with the model-based Bayesian WBA approach indicates that the proposed model is more robust to measurement noises.

Nonetheless, the proposed method suffers from the typical drawback of the machine learning model, e.g., a large amount of data is required. The necessity of collecting wave spectrum makes it even harder to collect in real-world scenarios. In addition, the training of adversarial networks might be unstable and requires careful tuning. Future works will focus on transferring the model trained in simulated environments to the real world, as well as including the vessels with advancing speeds.

CRediT authorship contribution statement

Peihua Han: Conceptualization, Methodology, Investigation, Software. **Guoyuan Li:** Resources, Supervision. **Stian Skjong:** Resources, Supervision. **Houxiang Zhang:** Project administration, Funding acquisition, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by a grant from the Research Council of Norway through the Knowledge-Building Project for industry "Digital Twins For Vessel Life Cycle Service" (Project nr. 280703) and a grant from the Research Council of Norway through the IKTPLUSS Project "Remote Control Centre for Autonomous Ship Support" (Project nr: 309323).

Appendix. Network architectures

The estimation network and discrimination network architectures used in this case are detailed in Fig. A.1(a) and Fig. A.1(b), respectively. Convolutional layers are denoted as "Conv" while transposed convolutional layers are denoted as "TranConv". The right of the figure suggests the signal dimension in terms of $height \times length \times channel$. For instance, the inputs for the estimation network are 9 components of the 1D motion spectrum ($1 \times 100 \times 9$) and the output is the 2D wave spectrum ($36 \times 100 \times 1$).

References

[1] Ochi MK. Ocean waves: the stochastic approach. (6). Cambridge University Press; 2005.
[2] Stredulinsky DC, Thornhill EM. Ship motion and wave radar data fusion for shipboard wave measurement. J Ship Res 2011;55(2).
[3] Brodtkorb AH, Nielsen UD, Sørensen AJ. Sea state estimation using vessel response in dynamic positioning. Appl Ocean Res 2018;70:76–86.
[4] Nielsen UD. Estimation of directional wave spectra from measured ship responses. In: 12th international congress of the international maritime association of the mediterranean: maritime transportation and exploitation of ocean and coastal resources, 2005. p. 1103–12.
[5] Cheng X, Li G, Ellefsen AL, Chen S, Hildre HP, Zhang H. A novel densely connected convolutional neural network for sea-state estimation using ship motion data. IEEE Trans Instrument Measur 2020;69(9):5984–93.
[6] Han P, Li G, Cheng X, Skjong S, Zhang H. An uncertainty-aware hybrid approach for sea state estimation using ship motion responses. IEEE Trans Ind Inf 2021;1. http://dx.doi.org/10.1109/TII.2021.3073462.
[7] Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial networks. 2014, arXiv preprint arXiv:1406.2661.
[8] Nielsen UD. Estimations of on-site directional wave spectra from measured ship responses. Mar Struct 2006;19(1):33–69.
[9] Tannuri EA, Sparano JV, Simos AN, Da Cruz JJ. Estimating directional wave spectrum based on stationary ship motion measurements. Appl Ocean Res 2003;25(5):243–61.
[10] Iseki T, Ohtsu K. Bayesian estimation of directional wave spectra based on ship motions. Control Eng Pract 2000;8(2):215–9.
[11] Ren Z, Han X, Verma AS, Dirdal JA, Skjetne R. Sea state estimation based on vessel motion responses: Improved smoothness and robustness using Bézier surface and L1 optimization. Mar Struct 2021;76:102904.
[12] Nielsen UD, Dietz J. Ocean wave spectrum estimation using measured vessel motions from an in-service container ship. Mar Struct 2020;69:102682.
[13] Pascoal R, Soares CG. Kalman filtering of vessel motions for ocean wave directional spectrum estimation. Ocean Eng 2009;36(6–7):477–88.
[14] Pascoal R, Perera LP, Soares CG. Estimation of directional sea spectra from ship motions in sea trials. Ocean Eng 2017;132:126–37.
[15] Belleter DJ, Galeazzi R, Fossen TI. Experimental verification of a global exponential stable nonlinear wave encounter frequency estimator. Ocean Eng 2015;97:48–56.
[16] Nielsen UD, Galeazzi R, Brodtkorb AH. Evaluation of shipboard wave estimation techniques through model-scale experiments. In: OCEANS 2016-Shanghai. IEEE; 2016, p. 1–8.
[17] Tu F, Ge SS, Choo YS, Hang CC. Sea state identification based on vessel motion response learning via multi-layer classifiers. Ocean Eng 2018;147:318–32.
[18] Mak B, Düz B. Ship as a wave buoy: Estimating relative wave direction from in-service ship motion measurements using machine learning. In: International conference on offshore mechanics and arctic engineering. Vol. 58882, American Society of Mechanical Engineers; 2019, V009T13A043.
[19] Han P, Li G, Skjong S, Wu B, Zhang H. Data-driven sea state estimation for vessels using multi-domain featuresfrom motion responses. 2021, Accepted by International Conference on Robotics and Automation (ICRA) 2021.
[20] Kawai T, Kawamura Y, Okada T, Mitsuyuki T, Chen X. Sea state estimation using monitoring data by convolutional neural network (CNN). J Mar Sci Technol 2021;26:947–62.
[21] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016. p. 770–8.
[22] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. 2015, arXiv preprint arXiv:1511.06434.
[23] Kingma DP, Ba J. Adam: A method for stochastic optimization. 2014, arXiv preprint arXiv:1412.6980.

[24] Hogben N, Cobb F. Parametric modelling of directional wave spectra. In: Offshore technology conference. OnePetro; 1986.
[25] NTNU. Research vessel R/V gunnerus. 2021, URL https://www.ntnu.edu/oceans/gunnerus.
[26] Fathi D. Shipx vessel responses (veres), ship motions and global loads, users' manual. 2004, Marintek AS, Trondheim. ShipX Vessel Responses (VERES).
[27] St Dinis M, Pierson Jr. WJ. On the motions of ships in confused seas. Technical Report, New York Univ Bronx School of Engineering and Science; 1953.
[28] Faltinsen O. Sea loads on ships and offshore structures. Vol. 1, Cambridge University Press; 1993.
[29] Nielsen UD. Introducing two hyperparameters in Bayesian estimation of wave spectra. Probab Eng Mech 2008;23(1):84–94.

ISBN 978-82-326-6033-9 (printed ver.)
ISBN 978-82-326-6301-9 (electronic ver.)
ISSN 1503-8181 (printed ver.)
ISSN 2703-8084 (online ver.)

NTNU

Norwegian University of
Science and Technology