Mechanical Systems and Signal Processing 216 (2024) 111446

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

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

Floating offshore wind turbine mooring line sections health status nowcasting: From supervised shallow to weakly supervised deep learning

Andrea Coraddu [a,*], Luca Oneto [b], Jake Walker [a], Katarzyna Patryniak [c], Arran Prothero [d], Maurizio Collu [c]

[a] Delft University of Technology, Delft, Netherlands
[b] University of Genoa, Genova, Italy
[c] University of Strathclyde, Glasgow, United Kingdom
[d] Det Norske Veritas, Aberdeen, United Kingdom

ARTICLE INFO

Communicated by D. Nikolaos

Keywords:
Floating offshore wind turbine
Mooring line
Health status
Real time monitoring
Nowcasting
Supervised learning
Weakly supervised learning
Shallow models
Deep models

ABSTRACT

The global installed capacity of floating offshore wind turbines is projected to increase by at least 100 times over the next decades. Station-keeping of floating offshore renewable energy devices is achieved through the use of mooring systems. Mooring systems are exposed to a variety of environmental and operational conditions that cause corrosion, abrasion, and fatigue. Regular physical in-service inspections of mooring systems are the golden standard for monitoring their health status. This approach is often expensive, inefficient, and unsafe, and for this reason, researchers are focusing on developing tools for digital solutions for real-time monitoring. Floating offshore renewable energy devices are usually equipped with a wide range of sensors, some low-cost, low/zero maintenance, and easily deployable (*e.g.*, accelerometers on the tower), contrary to others (*e.g.*, direct tension mooring line measurements), producing real-time data streams. In this paper, we propose exploiting the data coming from the first type of sensors for mooring systems health status nowcasting. In particular, we will first rely on state-of-the-art supervised shallow and deep learning models for predicting the health status of the different sections of the mooring lines. Then, since these supervised models require types and amount of data that are seldom available, we will propose new shallow and deep weekly supervised models that require a very small amount of data regarding worn mooring lines. Results will show that these last models can potentially have practical applicability and impact for real-time monitoring of mooring systems in the near future. In order to support our statements, we will make use of data generated with a state-of-the-art digital twin of the mooring system, OrcaFlex[1], for a floating offshore wind turbine reproducing the physical mechanism of the mooring degradation under different loads and environmental conditions. Results will show errors around 1% in the simplest scenario and errors around 4% in the most challenging one, confirming the potentiality of the proposed approaches.

* Corresponding author.
E-mail addresses: a.coraddu@tudelft.nl (A. Coraddu), luca.oneto@unige.it (L. Oneto), j.m.walker@tudelft.nl (J. Walker), katarzyna.patryniak@strath.ac.uk (K. Patryniak), arran.prothero@dnv.com (A. Prothero), maurizio.collu@strath.ac.uk (M. Collu).

[1] www.orcina.com

https://doi.org/10.1016/j.ymssp.2024.111446
Received 27 February 2023; Received in revised form 31 January 2024; Accepted 18 April 2024
Available online 27 April 2024

1. Introduction

Floating Offshore Wind Turbines (FOWTs) are an increasingly favored source of floating offshore renewable energy, and the total installed capacity is on the cusp of a rapid expansion [1]. The global installed capacity of FOWTs has already exceeded 60 megawatt and is projected to increase at least 100 times over the next decade with over 6 gigawatt of FOWTs projects earmarked for the future [2]. Station-keeping of floating offshore renewable energy devices is achieved through the use of mooring systems [3] that fix the structure to the seabed. A large-scale FOWTs farm may require up to 100÷200 mooring lines, each approximately 100÷250 meters long [4,5]. Many of the current economic barriers to FOWTs are related to preventing station-keeping failures because of their catastrophic consequences on the FOWTs [6,7].

Mooring systems are exposed to a variety of environmental and operational conditions that cause corrosion, abrasion, and fatigue [8,9]. These mechanism have as main results a reduction in the chain diameter reducing then structural integrity and mechanical characteristics of the mooring line [10]. In order to monitor mooring systems' health status and minimize failures, regular inspections are required [11]. Regular physical in-service inspections are currently the golden standard [12]. Generally, this task is carried out using divers or remotely operated vehicles [8]. Diver inspections are not favored due to the risk related to operating in deep water and human bias [13]. Remotely operated vehicles are employed with limited success since they need to be re-calibrated between successive measurements (*e.g.*, to avoid errors in transmitting their positioning and the damage location) [14]. However, the inspection-based approach to mooring systems health status monitoring is expensive and inefficient [12]. For this reason, industry and academia are focusing their attention on developing tools for digital real-time monitoring [14].

State-of-the-art approaches to monitor the health status of mooring systems are based on Integrated Marine Monitoring Systems (IMMS), which exploit numerical methods in combination with a series of measurements to assess their condition [8,15].

Numerical methods exploited in IMMS are mostly based on three families [13]: physical (PMs), data-driven (DDMs), and hybrid (HMs) models. PMs require a deep knowledge of the physical phenomena since they use as a predictor a physical model of the reality [16]. The higher the detail in modeling the equations which describe the physical phenomena, the higher the expected accuracy of the results and the computational time required for the simulation. DDMs, instead, infer the desired model directly from historical data collected and require no need of any a-priory knowledge of the underlying physical phenomena [17]. From a computational point of view, these models have an expensive training phase (*i.e.*, the phase of the construction of the model), but this phase is performed only one time, and once the model is trained, the prediction is computationally inexpensive [18]. However, since these models do not exploit physical knowledge, they often require a significant amount of data for the learning phase [19]. In particular, DDMs often require labeled data, namely data referring to healthy and degraded mooring lines: while the former is abundant and cheaply available, the latter is not in real applications [8]. HMs are a combination of PMs and DDM where the DDM model compensates the secondary effects not modeled by the PM, and the PM helps the DDM in reducing the amount of historical data required to train it [20]. Nevertheless, HMs require to have a reasonably accurate and computationally inexpensive PM available which is most of the time not the case for mooring lines [16].

IMMS mostly rely on directly monitoring the tension in the mooring lines as a method of assessing their integrity [21]. Pilot Projects, such as the Hywind floating wind farm in Scotland, demonstrate that direct tension monitoring devices are an effective solution to the problem of mooring system health monitoring [22]. In particular, for the Hywind Pilot Park, direct line sensors were installed into the bridles of individual mooring lines on FOWTs. More broadly, *e.g.*, for offshore oil and gas floating structures, multiple solutions have been proposed and commercially applied, such as the subsea inclinometers [23,24]. These devices measure line angle, which is then converted into the tension estimate through simple equations or precomputed look-up tables [25]. The direct impact of the mooring system integrity on the safety of the onboard personnel drove initiatives such as the 2017 Joint Industry Project on Mooring Integrity [25], which considered the currently available solutions: direct tension measurement, tension-angle measurement, tension–stress (electromagnetic) measurement, sonar and visual (cameras). The very same project also studied less common approaches: hull-mounted sonars (monitoring the mooring line declination), seabed-mounted sonars (monitoring the catenary shape), optical scanning apparatus for ropes (monitoring strain in the taut mooring lines), and depth sensors (monitoring the changes in pressure). These solutions, while effective, have some drawbacks [25–27]: (i) the high costs of direct line tension sensors installation make it infeasible for large FOWTs installation, (ii) direct line tension sensors operating sub-sea for an extended period of time require itself a monitoring system (*i.e.*, a meta-monitoring), increasing the complexity of the final system, (iii) incapability of detecting a failure in the lower segments of the chain (touchdown point and along the seabed towards the anchor), and (iv) insufficient measurement (refresh) rates for detecting rapidly increasing tension. For this reason, a significant number of low-cost, low/zero maintenance, and easily deployable alternative sensors have been developed to perform health status monitoring of mooring systems monitoring through alternative measurements. Examples of these sensors are anemometers and lidars for wind speed and direction and accelerometers and gyroscopes anchored to the FOWT tower, which provide information about the weather conditions and FOWT tower displacements and accelerations. Previous works [18,19] have shown that with these inputs, it is possible to predict in a real application the mooring line tensions, removing the necessity of relying on costly and hard-to-maintain sensors [26].

There is a large body of literature on the problem of monitoring the health status of mooring systems based on numerical models. As previously described, each method is characterized by a specific type of leveraged numerical model, *i.e.*, PM, DMM, or HM, the data sources fed to the models, *i.e.*, low-cost, low/zero maintenance, and easily collectable (*e.g.*, accelerometers data) or not (*e.g.*, direct tension data), and what they try to estimate (*e.g.*, fatigue or wear).

The evaluation of the health status of the mooring systems with PMs requires a numerical prediction of the dynamic performance during the life cycle of the floating structure [28]. Depending on the particular structure, anchoring system and location, the

frequency or time domain PMs are employed to model the dynamic performance of the system and the effect of the drag forces, motion amplitude, and wave amplitude on the tension in the mooring line sections [29]. Authors of [30] leveraged a time domain quasi-static approach to account for the non-linear effects in wave and motion amplitude, which can have a significant impact on fatigue in severe environmental conditions. Authors of [31] applied a dynamic approach, improving the quasi-static numerical prediction by accounting for the drag and inertia forces, as well as the bending and torsional elasticity of the mooring line segments. Authors of [32] developed a quasi-dynamic mooring line model, where the distributed forces along the mooring line are computed as a function of the drag and inertial forces (similarly to the dynamic approach of [31]); however, the static solution of the catenary shape is considered, aiming at high accuracy with low computational cost. Authors of [33] used a hybrid method between Computational Fluid Dynamics and Finite Element Analysis dynamic models providing a solution for accurate, but more computationally expensive, prediction of the dynamic response of highly coupled FOWT systems.

To model the wear in the catenary mooring lines other kinds of PMs need to be developed. In [34], the authors modeled the wear of the chain links of a spar FOWT based on the motion predicted by a time domain multi-body dynamic model, with the contact stiffness and friction based on the Hertzian contact theory [35] and Finite Element Analysis solution. The method has been later validated through the chain diameter measurements, as reported in [36]. Authors of [37] demonstrated the use of Finite Element Analysis to aid the selection of frequency for the ultrasonic guided waves approach for chain links health measurements. Authors of [38] applied Finite Element Analysis to analyze the acoustic emission wave propagation along the structure to determine mooring systems crack initiation and growth. Authors of [12], leveraged Finite Element Analysis to estimate the residual strength of the mooring system components, aiding the decision about further maintenance steps. The Noble Denton led Mooring Integrity for Floating Offshore Installations Joint Industry Project [39] formulated an analytical method to calculate inter-link wear of the catenary mooring system, accounting for the effects of adhesion, corrosion, surface fatigue and erosion, both at the design and operation stages.

There have been a number of successful studies that demonstrate the application of DDMs to reliably address monitoring offshore devices [40–48].

Authors of [46] developed deep neural network approach to extract damage-sensitive features from vibration frequency data coming from a lab-scale offshore jacket structure. Their model to classify damage under uncertainty was trained on data from a finite element model and intact experimental data. The proposed damage detection model was shown to outperform traditional DDMs. In subsequent work on the same structure Mousavi et al. [47], authors created a dictionary learning-based approach to classify damage to the structure under varying working loads. Results showed that the exploratory method presented in this work was promising for predicting certain classes of damage under varying working loads.

Specifically regarding monitoring station-keeping devices, the authors of [41] investigated fatigue damage of mooring lines using Artificial Neural Networks observing a significant influence of the environmental conditions (waves, wind, and currents) on the tension bands in the mooring lines. Additionally, the authors of [43–45] demonstrated the benefits of using DDMs for monitoring FOWTs when it comes to scaling predictive tools site-wide. Nevertheless, the works of [43–45] are primarily focused on catching extreme tensions and snap loads in Mooring Lines rather than specifying an area of damage for preventative measures before it leads to failure. Outwith the FOWTs industry, the authors of [40] employed DDMs to approximate the response of a mooring system in place of a computationally expensive PM, where they used a non-linear autoregressive exogenous model as a surrogate for a Finite Element Analysis model to accurately simulate the mooring line tension response over a period of multiple hours. Furthermore, the authors of [42] developed a Long Short Term Memory network for real-time monitoring of a mooring system for a Floating Production Storage Offshore. Their approach leveraged a synthetic dataset developed using a dynamic PM and was able to predict the mooring line tension under different environmental conditions.

While the above studies are an important first step, there are just a few studies that demonstrate the application of DDMs to address, with costs awareness, monitoring station-keeping devices, *i.e.*, without direct tension monitoring [49–52]. The authors of [49] developed a Position Response Learning System able to classify if a mooring line has catastrophically failed without relying on direct tension measurements. Their system relies on synthetic data of a Floating Production Storage Offshore vessel to develop a Deep Neural Network capable of detecting a mooring line failure: the system succeeded in its task with a classification accuracy of 98%. Part of this work focused on detecting the most important features for the model to achieve good performance. The GPS location, the acceleration of the platform in 6 degrees of freedom, and the environmental conditions were the features that led to the highest accuracy. While the Position Response Learning System could be used to classify which specific mooring line has broken with 60÷70% accuracy, this model does not provide insight into the degradation mechanism or the reason for the disconnection. Additionally, as the mooring system degrades with time and the system dynamics deviate from the initial conditions, the Position Response Learning System will be required to periodically relearn the response of the system to account for these changes. Authors of [50] investigated the use of a Convolutional Neural Network to classify the condition of a mooring line for a Floating Production Storage Offshore using trace images. Specifically, they rely on the offset traces of the platform in the horizontal plane alongside the platform accelerations in 6 degrees of freedom as input to the model. The dataset was constructed using a PM supplied with different points in time across a wide range of operating conditions. The system was able to classify the mooring system condition as intact or damaged but could not give an indication of the possible failure mechanism until a close inspection of the line was carried out. The authors of [52] exploited a Deep Learning model to detect damage of a floating production storage offshore mooring system. Their study relied on synthetic data developed from a dynamic PM under different environmental conditions to infer the damage to different tension legs: the ability to detect an area the damage was limited to the tension leg instead of localizing the area of damage. Authors of [48] developed a convolutional neural network based approach to detect the damage magnitudes of the mooring lines

of a barge platform. The proposed approach showed that when the dynamics of the mooring lines were investigated from structure creep to failure the structural damages were clearly distinguished by the response of the yaw motions.

There is a very limited amount of research in the context of HMs with limited success for the reasons described in the introduction. For example, HMs have been exploited to obtain the fatigue characteristics of individual mooring lines, simultaneously exploiting data coming from sensors mounted on the floating structure (*e.g.*, accelerometers, gyroscopes, and satellite positioning) and the knowledge encapsulated in already known PMs. The authors of [53] exploited the position and motion signals coming from the sensors mounted on a Hywind Scotland Wind Turbine in combination with the transfer functions coming from a PM to infer the mooring line tension cycles. Then, the conventional S-N curve method [54] and peridynamic fatigue analysis approaches [55] were trialed to access the accumulated fatigue damage. Significant discrepancies in the line tension were obtained this way, and the actual tension measured with load cells was observed. This was attributed to both the limitations in the monitoring system (*e.g.*, lack of acceleration data and a noisy signal obtained from the load cells) and limitation of the exploited PM (*e.g.*, the assumption of idealized positions of the anchors and the discrepancies between the blade-pitch controller implemented in the PM and the one used by the real Wind Turbine).

The current methods for monitoring FOWTs mooring systems faces a significant challenge in effectively utilizing data from low-cost, low-maintenance, and easily deployable sensors, such as tower accelerometers, for nowcasting the health status of mooring systems. This highlights the need to develop novel methods that can exploit the available data streams and infer the health status without the need for costly inspections. The primary obstacle to achieve this goal is the lack of extensive labeled data on worn mooring lines, which hampers the development and training of DDMs. In general, DDMs require sufficient data to accurately predict the degradation state of mooring lines, yet acquiring such data in real-world scenarios is exceedingly rare.

Based on these premises, in this paper, we propose to build models able to estimate the degradation state of a mooring line based on just low/zero maintenance and easily deployable alternative sensors, namely weather conditions and FOWT tower displacements and accelerations. In particular, since no reasonably accurate and computationally inexpensive PM is available for this purpose, we will leverage DDMs purely. Moreover, since in real applications, labeled data referring to healthy and degraded mooring lines are scarcely available, this study aims at designing and proposing multiple alternatives for the reduction of the use of labeled data towards a weekly supervised health status mooring line monitoring, targeting to reduce as much as possible the necessity of labeled data at least to a level which is realistic to retrieve in the wild. For this purpose, we will first rely on state-of-the-art supervised shallow and deep learning models for predicting the health status of the different sections of the mooring lines. Then, since these supervised models require types and amount of data that are seldom available, we will propose new shallow and deep weekly supervised models that require a very small amount of data regarding worn mooring lines. Results will show that these last models have the potential to have practical applicability and impact for real-time monitoring of mooring systems of floating offshore renewable energy in the near future. To support our statements, we will make use of data generated with a state-of-the-art digital twin of the mooring system to reproduce the physical mechanism of the mooring degradation under different loads and environmental conditions.

The rest of the paper is organized as follows. In this section we provided an overview on FOWT mooring systems, a review of the related work on numerical models for modeling mooring systems, a description of the open challenges in the literature, and the contributions of this work. Section 2 will describe the problem and the available data to address it. Section 3 will detail the proposed methodology. Section 4 will assess the effectiveness of the proposal of Section 3 in addressing the problem described in Section 2 exploiting the data described in the very same section. Finally, Section 5 concludes the paper.

2. Problem description and available data

This section is devoted to the formalization of the problem and the description of the available data to address it.

The FOWT considered for this study is the OC4-DeepCwind, a semi-submersible floating wind turbine developed by the National Renewable Energy Laboratory in partnership with industry partners.[2] The platform consists of a central buoyant hull supporting the tower, surrounded by an array of three large-volume cylindrical hulls. The large second moment of waterplane area of the platform provides high stability, significantly reducing the motions of the system in wind and waves, which, among other benefits, reduces the stress on the turbine's components and increases its lifespan. It is designed to be anchored to the seabed in deep waters (hence the name "DeepCwind") where the wind is stronger and more consistent than closer to shore. The OC4-DeepCwind platform sits on the surface of the water and is anchored to the seabed with mooring lines. The OC4-DeepCwind turbine has a rated capacity of 5 MW and a rotor diameter of 172 m. Fig. 1 depicts the OC4-DeepCwind highlighting the mooring system and the considered sensing equipment. In particular, we will consider

- gyroscopes positioned on the tower to measure surge, sway, and heave angular velocity, acceleration, and displacement;
- accelerometers positioned on the tower to measure surge, sway, and heave velocity, acceleration, and displacement;
- load-pins and declinometers, anchored where the mooring line is attached to the fairleads, to measure mooring line tension and tension-angle;
- lidar to measure the wind speed.

Weather information, providing, *e.g.*, significant wave height and main waves direction, are usually retrieved from the Meteorological Service.[3] Note that, in this study, we considered each mooring line to be composed of three sections of the same

[2] www.nrel.gov/docs/fy14osti/60601.pdf

[3] www.metservice.com

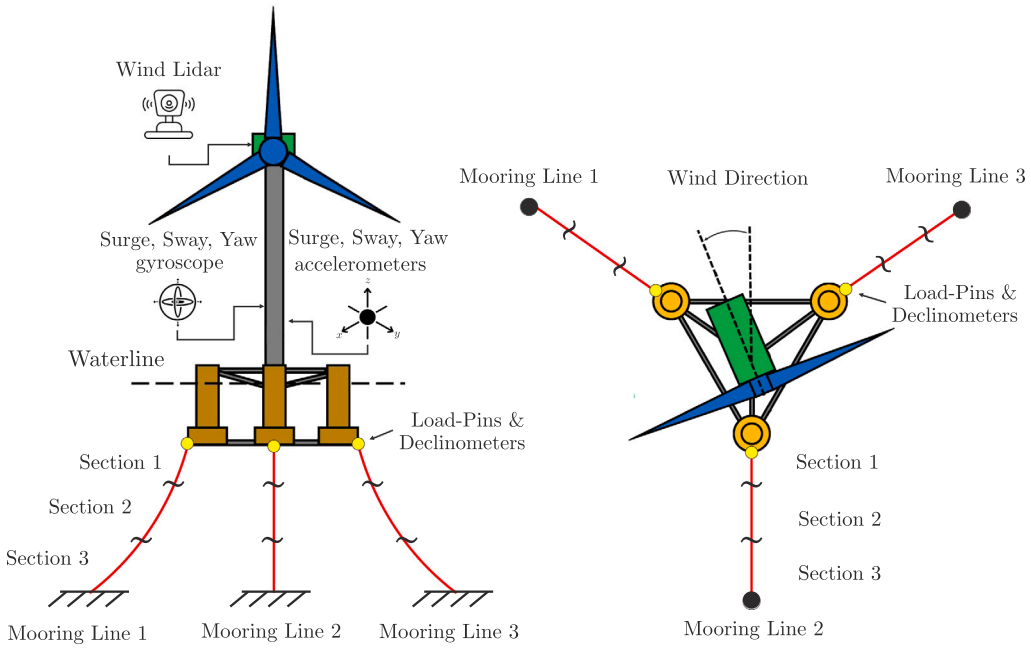


Fig. 1. OC4-DeepCwind highlighting the mooring system and the considered sensing equipment.

length, studying the health status of each section. The first section is the one attached to the FOWT, the third one is the one attached to the seabed, and the second one is placed in between.

Since there is no real data available, providing also information about the health status of the different sections of the different mooring lines, we will leverage data generated with a state-of-the-art digital twin of the mooring system. To develop the digital twin, we leverage OrcaFlex,[1] a leading software package for marine systems design and analysis of a FOWT, able to reproduce the physical mechanism of the mooring degradation under different loads and environmental conditions and consequently the realistic data that are required in our study.

In more detail, we exploited OrcaFlex to synthetically generate a time series of data generated by the gyroscope, the accelerometer, the mooring lines load-pins and the mooring lines declinometers at 8 different collinear wind and wave directions (namely $\{0, 15, 30, \dots, 105\}$ degrees) under different health status conditions of the different sections of the different mooring lines. In more detail, we exploited OrcaFlex to synthetically generate a time series of data generated by the gyroscope, the accelerometer, the mooring lines load-pins, and the mooring lines declinometers at 8 different collinear wind and wave directions (namely $0, 15, 30, \dots, 105$ degrees) under different health status conditions of the different sections of the different mooring lines. For each experiment, the wind speed has been consistently maintained at a constant 4 meters per second and the significant wave height has been set at 1.1 meters with a peak wave period of 8.7 s. The choice to maintain constant environmental conditions, with a fixed wind speed, significant wave height, and a peak wave period was made to ensure that the primary focus of the study was on the effect of the degradation of the mooring lines themselves, rather than the influence of varying environmental forces.

Note that authors are aware that some sensors (*e.g.*, lidar) can be prone to errors [56]. Nevertheless, since data have been generated with OrcaFlex, these inaccuracies and their impact have not been considered in this work. Exploiting the platform wet geometry symmetry, and the fact that the mass distribution is largely symmetrical with respect to the vertical axis, only the angles between 0 and 105 degrees have been considered in this analysis. Furthermore, following the same approach described in the technical specification IEC61400-3-2 [57], the wind speed and wave significant height and peak period have been maintained constant over each load case considered. Finally, OrcaFlex allowed us to consider all the possible combinations of the degradation state of the sections of the mooring lines, *i.e.*, Section $i \in \{1, 2, 3\}$ of Mooring Line $j \in \{1, 2, 3\}$ can be healthy (no reduction in the chain diameter due to the main degradation mechanism of corrosion, abrasion, and fatigue as stated in the introduction) or degraded (10% reduction in the chain diameter), for a total of $2^9 = 512$ combinations. Then, by multiplying together the 8 different collinear wind and wave directions with the 512 combinations of possible decay states, we had to run a total of 4096 Orcaflex simulations with each one producing 2 hours of gyroscope, accelerometer, mooring lines load-pins, and the mooring lines declinometers data. The details of these data have been summarized in Table 1.

It is important to note that the threshold of 10% in the reduction in the chain diameter has been selected according to the literature. In fact, as the authors of [58,59] state, for chain components, the relationship between diameter reduction and breaking load reduction is non-linear: ≈10% reduction to the diameter of a mooring chain is considered to reduce the minimum breaking load of the component by ≈20%. Authors of [60] state that historically widely accepted industry criteria for reducing component breaking strength before replacement is 10% for permanent offshore units. Although the industry is trying to move away from this

Table 1
Available data description.

Sensor	Data acquired		Type	Sampling frequency	Duration
Gyroscope	Roll	Velocity Acceleration Displacement	Time Series	10 Hz	2 hours
	Pitch	Velocity Acceleration Displacement			
	Yaw	Velocity Acceleration Displacement			
Accelerometer	Surge	Velocity Acceleration Displacement			
	Sway	Velocity Acceleration Displacement			
	Heave	Velocity Acceleration Displacement			
Lidar	Wind Direction		Scalar	–	–
Mooring Lines 1, 2, and 3 Load-Pins	Tension		Time Series	10 Hz	2 h
Mooring Lines 1, 2, and 3 Declinometer	Tension angle				

prescriptive criteria stated in [60] to a risk-based inspection regime, the use of a maximum 10% diameter reduction is considered by authors of this paper as conservative for modeling a corroded chain component.

In this work, we want to face multifold problems. The general one is to predict the status (*i.e.*, whether the reduction in the chain diameter is above or below 10%) of the mooring system based on the data of Table 1. More specifically, we want to address this general problem by trying to

- predict the status of all the sections of each mooring line, not simply the whole mooring system;
- avoid the use of mooring line tension and tension angle sensors that are, from one side, very informative on the reduction in the chain diameter but, from the other side, are costly and require maintenance themselves;
- reduce, as much as possible, the need for labeled data in general (*i.e.*, data with associated mooring line health status) and specific data about unhealthy mooring sections that are seldom available in real applications.

3. Methodology

In this section, we will describe the proposed methodology for addressing the problem described in Section 2 leveraging the data described in the same section.

The problem we want to address in this work can be mapped to conventional learning from data framework [61,62], in which one has to consider an input space $\mathcal{X}$ (*i.e.*, the data of Table 1) and an output space $\mathcal{Y}$ and the goal is to estimate the unknown rule μ which maps $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$. In our case, the output space is binary $\mathcal{Y} = \{\pm 1\}$ (*i.e.*, $Y = -1$ means reduction in the chain diameter is below 10% and $Y = +1$ means above 10%). In order to estimate the unknown rule μ a series of historical examples, namely the dataset $\mathcal{D}_n : \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ (*i.e.*, the 4096 samples described in Section 2) is available. When $\mathcal{D}_n$ contains both elements in the input space and the associated labels from the output space, the problem is named supervised and, more specifically, in our case, the problem is named binary classification [61]. When $\mathcal{D}_n$ contains just elements in the input space, which means that the associated labels from the output space are not explicitly known (or known only for just a few elements), the problem is called unsupervised (or weekly/semi-supervised) [61]. Finally, when $\mathcal{D}_n$ contains elements from the input space and only few of them label, but we know that most of the input space is coming from elements associated with $Y = 1$, the problem is called anomaly (novelty/outlier) detection [63,64].

A learning from data algorithm $\mathcal{A}$ tries to approximate μ based on $\mathcal{D}_n$. More specifically, an algorithm $\mathcal{A}_{\mathcal{H}}$, where $\mathcal{H}$ are the hyperparameters that characterize the algorithm, maps $\mathcal{D}_n$ into a model f inside a space of possible ones $\mathcal{F}$. Currently, there are two main families of learning algorithms in the literature: namely shallow [61] and deep [62]. The difference relies on how the input space $\mathcal{X}$ is preprocessed. For shallow learning algorithms, $\mathcal{X}$ is first mapped, by means of a handcrafted feature engineering phase, into a vector $\boldsymbol{\phi}(X) \in \mathbb{R}^d$, named representation, able to well represent $\mathcal{X}$ while discarding the unuseful information [65,66]. For deep learning algorithms, $\mathcal{X}$ is mapped into a representation $\boldsymbol{\phi}(X) \in \mathbb{R}^d$ by means of an architecture able to learn the best mapping without

fixing it a priori [62,67]. Then, based on this representation $\boldsymbol{\phi}(X)$, different simple shallow algorithms (*e.g.*, Linear Models [68], Ensemble Methods [69,70], and Bayesian Methods [71]) make the actual prediction of $y \in \mathcal{Y}$.

The error that the model learned by $\mathscr{A}_{\mathcal{H}}$, namely f, commits in approximating μ is measured with reference to a metric [72,73] and is deeply influenced by choice of the algorithm $\mathscr{A}$ and its hyperparameters $\mathcal{H}$. In this work, since we want to detect if a section of mooring exhibits a reduction in the chain diameter below 10% ($Y = -1$) or above 10% ($Y = +1$) we will measure the overall percentage of error (%ER), the percentage of false positive (%FP), and the percentage of false negative (%FN). This error must be optimized during the Model Selection phase [74]. Finally, the performance of the final model needs to be assessed during the Error Estimation phase [74]. Obviously, $\mathcal{D}_n$ cannot be exploited to perform all the necessary phases described above since we cannot use the data in $\mathcal{D}_n$ to construct a model and assess its performance simultaneously. For this reason, $\mathcal{D}_n$ is split into three mutually exclusive sets of data called learning $\mathcal{L}_l$, validation $\mathcal{V}_v$, and test $\mathcal{T}_t$ sets such that

$$\mathcal{D}_n = \mathcal{L}_l \cup \mathcal{V}_v \cup \mathcal{T}_t, \quad \mathcal{L}_l \cap \mathcal{V}_v = \varnothing, \quad \mathcal{L}_l \cap \mathcal{T}_t = \varnothing, \quad \mathcal{V}_v \cap \mathcal{T}_t = \varnothing. \tag{1}$$

Then, the $\mathcal{L}_l$ is exploited to learn f with $\mathscr{A}_{\mathcal{H}}$, $\mathcal{V}_v$ to choose $\mathscr{A}$ and $\mathcal{H}$, and $\mathcal{T}_t$ to assess the performance of the f learned with the best $\mathscr{A}$ and $\mathcal{H}$. To ensure the statistical relevance of the results, the procedure needs to be repeated multiple times (in our case 30 times), reporting the average results [74]. Note that, when we deal with an anomaly detection problem and then we use an anomaly detection algorithm, $\mathcal{L}_l$ does not need to contain labeled samples and so just $\mathcal{V}_v$ and $\mathcal{T}_t$ needs to contain labeled samples. This strongly reduces the necessity of labeled samples to the smallest possible amount [75,76].

In the next section, we will incrementally address the multifold problems described in Section 2, *i.e.*, predict whether the reduction in the chain diameter is above or below 10% based on the data of Table 1, as follows

- first, in Scenario 1 (see Section 3.1) we will solve a binary classification problem (*i.e.*, whether the reduction in the chain diameter is above or below 10% for all the mooring line sections) using both shallow and deep learning in a fully supervised way using
  - a complete dataset, *i.e.*, we suppose to have the whole dataset of Table 1 for each combination of the health status of each mooring line section (Scenario 1 A);
  - a partial dataset, *i.e.*, we assume to have the dataset of Table 1 for each combination of the health status of each mooring line section but without the information about tension and tension angle (Scenario 1B);
- then, in Scenario 2 (see Section 3.2) we will solve an anomaly detection problem, *i.e.*, whether the reduction in the chain diameter is above 10% for all the mooring line sections, using both shallow and deep learning in a weakly supervised way using a partial dataset, *i.e.*, we suppose to have the dataset of Table 1 without the information about tension and tension angle, to know that most of the samples are coming from healthy mooring line sections, and to have just minimal amount of labeled data.

The purpose of these models is to reach the ability to address all the requests of the practical application described in Section 2 and to have a different baseline for comparison in Section 4 to measure what is the impact on the performance of the resulting data-driven monitoring systems incrementally adding the requests of the practical application.

3.1. Fully supervised: from Shallow to Deep Models

Scenario 1 A and 1B can be mapped into a series of simple binary classification problems (more precisely 9 binary classification problems since we have 3 mooring lines each composed of 3 sections), *i.e.*, $\mathcal{Y} = \{\pm 1\}$ where $Y = -1$ means that the reduction in the chain diameter is below 10% while $Y = +1$ means that it is above. However, there is a difference with the input space $\mathcal{X}$, *i.e.*, for Scenario 1 A $\mathcal{X}$ counts of all the information of Table 1 while for Scenario 1B $\mathcal{X}$ counts the same information except for the tensions and tension angles.

In order to solve these binary classification problems we will consider both shallow and deep models.

Shallow models rely on a simple idea. First, $\mathcal{X}$ is mapped in a new space by means of the mapping $\boldsymbol{\phi} : \mathcal{X} \to \mathbb{R}^d$ based on previous knowledge about the problem or based on classical signal processing techniques [66,77]. Then, on this new space, a linear (*e.g.*, Support Vector Machines [68], Kernel Methods [68], or Ensemble Methods [69,70]) shallow model is applied. Simply feeding the raw features of Table 1 to the shallow model will never work [78]. In our case, for the scalar variables of Table 1, we kept them as is, while for the time series we performed a feature engineering phase following state-of-the-art approaches proposed in [79–83] where classical signal processing techniques have been applied in order to extract from the time series a vector of representation composed by variables such as the mean, the median, and the signal magnitude area for both the time and frequency domains for a total of $d = 1010$ for Scenario 1 A and $d = 650$ for Scenario 1B. The Fast Fourier Transform was used to obtain frequency components.

Table 2 shows the list of measures applied in both time and frequency domain signals.

At the end of this feature engineering step, we applied a series of state-of-the-art top-performing classification algorithms[4] [84,85]: Linear and Nonlinear (Gaussian Kernel) Support Vector Machines [68] (respectively LSVM and KSVM), Random Forest (RF) [70], and XGBoost (XGB) [69]. Each one of these models have a series of hyperparameters that need to be tuned. LSVM has the regularization hyperparameter C while KSVM has both C and the kernel coefficient γ to be tuned. In RF we need to tune

[4] Results in Kaggle www.kaggle.com, the most popular Machine Learning competition website, shows how Support Vector Machines, Random Forest, and XGBoost algorithms are the top winners.

Table 2
Features engineered from the time series of Table 1.

Function	Description
mean	Mean value
var	Variance
mad	Median absolute value
max	Largest value in array
min	Smallest value in array
sma	Signal magnitude area
energy	Average sum of squares
iqr	Interquartile range
entropy	Signal Entropy
correlation	Correlation coefficient between series
kurtosis	Signal Kurtosis
skewness	Signal Skewness
maxFreqInd	Largest frequency component
argMaxFreqInd	Index largest frequency component
meanFreq	Frequency signal weighted average
skewnessFreq	Frequency signal Skewness
kurtosisFreq	Frequency signal Kurtosis
ampSprec	Amplitude Spectrum of the frequency signal
angle	Phase angle of the frequency signal

Table 3
Hyperparameters and hyperparameters search space for all algorithms tested in this work.

Algorithm	Hyperparameters
LSVM	C : $\{0.001, 0.01, 0.1, 1, 10, 100\}$
KSVM, OCSVM	C : $\{0.001, 0.01, 0.1, 1, 10, 100\}$ γ: $\{0.1, 0.01, 0.001, 0.0001\}$
RF	n_f : $\{d^{1/3}, d^{1/2}, d^{3/4}\}$ n_l : $\{1, 3, 5, 10\}$ n_d : $\{5, 7, 10\}$ n_t : $\{1000\}$
XGB	l_r : $\{0.01, 0.02, 0.03, 0.04, 0.05\}$ n_d : $\{3, 5, 10\}$ m_l : $\{0, 0.1, 0.2\}$ n_b : $\{0.6n, 0.8n, 1n\}$ n_f : $\{0.5d, 0.8d, 1d\}$
TCN	l_r: $\{0.0001, 0.0005, 0.001, 0.005, 0.01\}$ $d_{r,0}$: $\{0.1, 0.15, \ldots, 0.5\}$ C : $\{0.00001, 0.00005, 0.000001\}$ h_l : $\{1, 2, 3, 4\}$ n_i : $\{16, 32, 64, 128, 256\}$ $k_{s,i}$: $\{3, 5, 7, 9, 11\}$
GKNN	k : $3^{\{1,2,3,4,5,6\}}$

the number of features to randomly sample from the entire set of features at each node of each tree n_f, the maximum number of elements in each leaf of each tree n_l, and the maximum depth of each tree n_d. As RF performance improves increasing the number of trees, n_t, we set it to 1000 as a reasonably large number but yet computationally tractable. Finally, in XGB we need to tune the learning rate of the gradient l_r, the max depth of arch tree n_d, the minimum loss reduction m_l, number of training samples to randomly sample from the whole training set for each tree creation n_b, and the number of feature to randomly sample from the entire set of features at each node of each tree n_f. The summary of these hyperparameters with the associated search space is reported in Table 3.

In order to further improve the performance of the shallow models, we decided to add a dimensionality reduction step to remove all the uninformative variables, which may be numerous due to the comprehensive feature engineering phase [86]. In particular, for each training phase of each model, we applied the permutation feature importance method [70,87,88], using the mean decrease of accuracy as a metric, and removed all the features with no positive impact according to this metric. Then we retrained the model on this reduced feature set. The pipeline we proposed for shallow models is depicted in Fig. 2. As a final note, with shallow models, we need to create a different model for each one of the 9 targets [61].

Shallow models have one main limitation: their dependency on handcrafted and experience-based features identified through a feature engineering step which may include too many irrelevant features or leave out important features. Deep models allow us to overcome these limitations. Relying on state-of-the-art architectures able to address temporal analysis, such as the classical and the bidirectional Long-Short-Term Memory network (LSTM) [89,90], it is possible to learn the features from the data. Even if

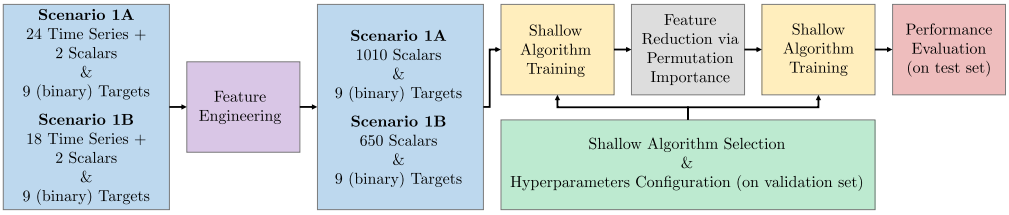


Fig. 2. Pipeline for the fully supervised shallow models.

powerful, classical and bidirectional LSTM networks have recently shown to have some limitations [83,91] in practical applications. To overcome LSTM limitations, we decided, similarly to [83,91], to substitute the LSTM blocks with Temporal Convolutional Network (TCN) residual blocks [92–94] which are able to focus on multiple temporal scales for each raw time series independently. The proposed architecture is reported in Fig. 3. There are three main peculiarities of the proposed deep multiple temporal scale architecture based on TCN: (i) the convolutions in the architecture are causal, namely, there is no information leakage from future to past, (ii) the architecture can handle different sequences lengths and map them to an output sequence of the same length like the LSTMs, and (iii) the architecture is able to handle long effective history. For what concerns (i) the TCN uses causal convolutions. For what concerns (ii), it is due to the use of 1D fully-convolutional network model where each hidden layer has the same length of the input layer; zero padding of length (kernel size − 1) is added to preserve the previous length. As for (iii) we employed dilated convolution that enables a large receptive field [95] without employing too deep TCN residual blocks. In particular, we defined a standard architecture where the different raw time series are fed to the TCN layer which is in charge of extracting the representation that is then fed directly to a dense L2 regularized layer to produce the required 9 outputs directly. The network has been trained with the ADAM optimizer empowered with one-cycle learning rate. This architecture has a series of hyperparameters that need to be carefully tuned: the learning rate l_r, the dropout rate $d_{r,0}$ on the last TCN layer and the final dense fully connected layer, the number of TCN blocks h_l for each time series, the number of filters in each block n_i ($i \in \{1, \dots, h_l\}$), the kernel dimension $k_{s,i}$ for each series s and each block i, and the L2 regularization on the weight of the entire network C (see Table 3). Note that, in this case, the hyperparameters configuration space explodes since we have, possibly, different configurations of the kernel dimension for each time series. For this reason, to reduce this hyperparameters configuration space and consequently reduce the number of weights of the final network (both for LSTM and TCN based deep models), we decided on a twofold strategy. The first one is to reduce the number of input series from the original 24 for Scenario 1 A and 18 for Scenario 1B. In fact, it is reasonable to assume that many of these series contain redundant information. In order to address this issue, we rely on shallow models as suggested in [83]. In particular, similarly to the feature reduction phase implemented in shallow models, we implemented a time series reduction phase. Using the permutation importance again with a mean decrease of accuracy as a metric, we permuted not the engineered features but the original time series discarding all the time series that non positively contribute according to the mean decrease of accuracy. Thanks to this reduction in the number of input time series, the TCN based architecture strongly reduces the number of weights to tune and the hyperparameters configuration search space. The second strategy to reduce the hyperparameter search space is to randomly search on it, not performing the grid search as suggested in [96]. The pipeline we proposed for deep models is depicted in Fig. 4.

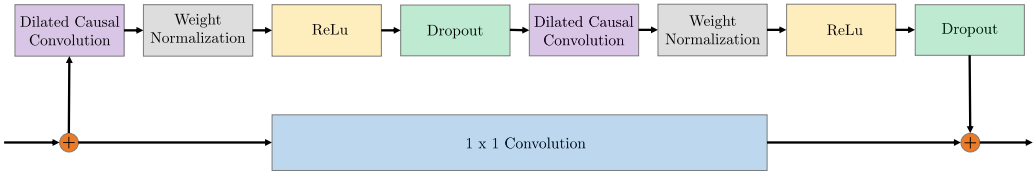
3.2. Weakly supervised: from Shallow to Deep Models

Scenario 2 can be mapped into a series of simple anomaly detection problems (more precisely 9 since we have 3 mooring lines each composed of 3 sections), *i.e.*, an anomaly is when the reduction in the chain diameter is above 10%. The input space $\mathcal{X}$ of Scenario 2 counts all the information of Table 1 except for tensions and tension angles.

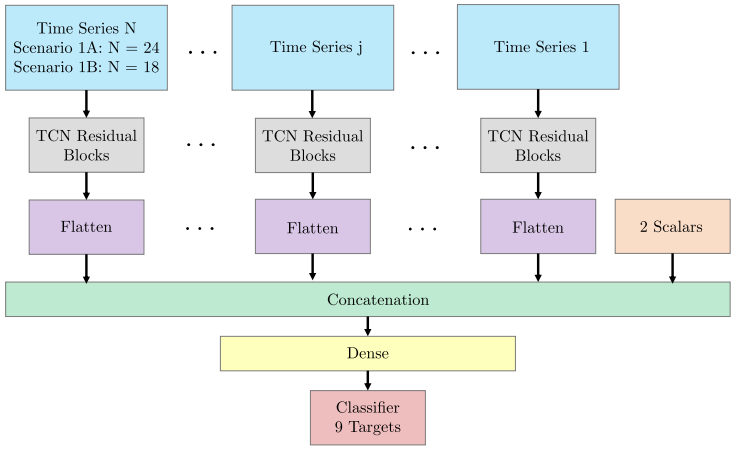
In order to solve these anomaly detection problems, we can rely on, as we did in Section 3.1, both shallow and deep models.

Shallow models rely, similarly to what is explained in Section 3.1, on a simple pipeline. First, $\mathcal{X}$ is mapped in a new space by means of the mapping $\boldsymbol{\phi} : \mathcal{X} \rightarrow \mathbb{R}^d$ based on previous knowledge about the problem or based on classical signal processing techniques [66,77]. In this section, we will rely on the same feature engineering phase described in Section 3.1. Then, in this new space, top shallow learning algorithms for anomaly detection problems are exploited. According to [64,73], two anomaly detection methods based on Kernel Methods [68] and K-Nearest Neighborhood [68] respectively, are the top choices in this context. In particular One-Class Support Vector Machines (OCSVM) [68] is a boundary-based anomaly detection method inspired by Kernel Methods, which enclose the inlier class in a minimum volume hypersphere and like Kernel Methods can also be extended to non-linearly transformed spaces using the Kernel trick. The hyperparameters of OCSVM are the same as the ones of KSVM. The Global K-Nearest Neighborhood (GKNN), inspired by the K-Nearest Neighborhood, has been originally introduced as an unsupervised distance-based outlier detection method [97]. The hyperparameter GKNN is the number of neighbors to be considered k. The summary of these hyperparameters with the associated search space is reported in Table 3. Note that, contrary to what has been done in Section 3.1, we cannot further improve the performance by adding a dimensionality reduction step since we do not have enough labeled samples. The pipeline we proposed for shallow models is depicted in Fig. 5.

Shallow models for anomaly detection have the same main limitation described in Section 3.1: they depend on handcrafted and experience-based features identified through a feature engineering step which may include too many irrelevant features or leave



(a) High level representation of the architecture.



(b) TCN-based layer architecture details.

Fig. 3. The proposed Deep Multi Scale Models architecture based on TCN.

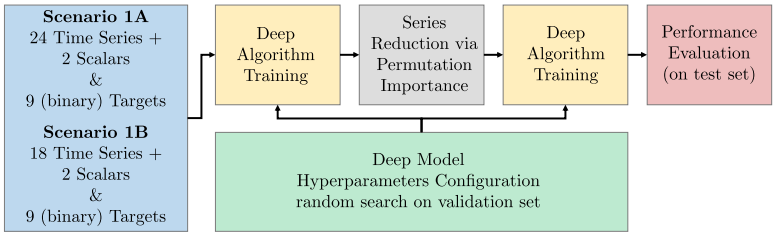


Fig. 4. Pipeline for the fully supervised Deep Models.

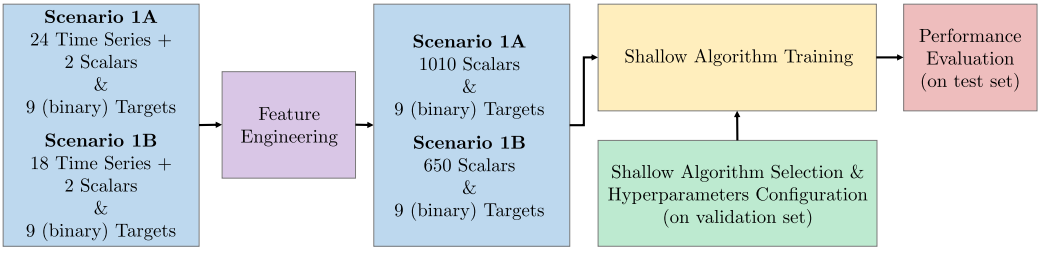


Fig. 5. Pipeline for the weakly supervised shallow models.

out important features. Deep models, in Section 3.1, allowed us to overcome this limitation, learning this representation from the data. Unfortunately, the same approach cannot be adopted here since we do not have labeled samples to learn this representation.

For this reason, we will learn this representation in a fully unsupervised way [62]. In particular, we replace as target to Fig. 3(a) the reconstruction of the original series, namely we give as target to the neural network of Fig. 3(b) the original time series. This

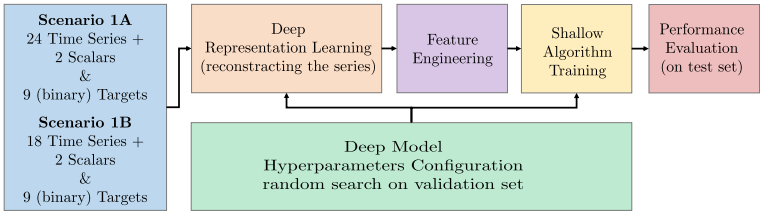


Fig. 6. Pipeline for the weakly supervised deep models.

allows us to learn a representation in a fully unsupervised way [98]. The hyperparameters of this network are the same as the one of Section 3.1 reported in Table 3. Then, on top of this representation, we will rely on the shallow models that we just described (OCSVM and GKNN) with the hyperparameters of Table 3. Also, in this case, we could not reduce the number of series as we did for deep models in Section 3.1 since no labeled data are available. The pipeline we proposed for deep models is depicted in Fig. 6.

4. Experimental results

In this section, we will show the results of applying the methodology described in Section 3 to the problems described in Section 2 leveraging the data reported in the very same section.

In particular, Section 4.1 will report the results in the fully supervised Scenarios 1 A and 1B, comparing the results of Shallow and Deep Models. Section 4.2, instead, report the results in the weakly supervised Scenarios 2 comparing, as in Section 4.1, the results of Shallow and Deep Models.

Sections 4.1 and 4.2 also report a summary showing the decrease in performance for both Shallow and Deep Models, going from Scenarios 1 A, *i.e.*, the less realistic but also challenging one, to Scenarios 2, *i.e.*, the most realistic but also challenging one.

4.1. Fully supervised: from Shallow to Deep Models (Scenario 1)

This section is devoted to the experimental evaluation of Shallow and Deep Models in Scenarios 1 A and 1B (see Section 3), the fully supervised ones, testing the methodology presented in Section 3.1 leveraging the data described in Section 2. Note that Scenarios 1 A and 1B differ just in the data, or better features, that are available to make predictions about the health status (*i.e.*, whether the reduction in the chain diameter is above or below 10%) of the 3 sections of the 3 mooring lines. In particular:

- Section 1 A exploits all the features of Table 1
- Section 1B exploits all the features of Table 1 except the information about tension and tension angle (*i.e.*, information coming from sensors that are costly, require maintenance, and are hard to deploy as described in Section 1).

Note that in this case $l = 0.8n$, $v = t = 0.1n$, namely 80% of the data has been used for learning, 10% for validation and 10% for test.

4.1.1. Exploiting tension and tension angle measurements (Scenario 1 A)

Let us consider Section 1 A.

Table 4 reports the quality (in terms of %ER, %FP, and %FN) averaged on the mooring lines and sections of the different Shallow (LSVM, KSVM, RF, and XGB) and Deep (TCN) algorithms described in Section 3.1.

Table 5, instead, reports the quality (in terms of %ER, %FP, and %FN) for each mooring line and each section for the best Shallow (XGB) and Deep (TCN) algorithms employed. The best Shallow and Deep algorithms have been chosen in accordance with Table 4 considering the metric %ER.

From Tables 4 and 5 it is possible to observe that

- quality of the results is, in general, high. In fact, apart from the LSVM, all models report %ER below 5%;
- %FP and %FN are quite aligned to the %ER, meaning that the models are performing well in predicting whether the reduction in the chain diameter is both above or below 10%;
- among the shallow model XGB resulted in being the best performing one immediately followed by RF. This is not surprising and in line with what is described in Section 3.1;
- the deep model based on TCN strongly outperform XGB showing the actual potentiality of the proposed architecture;
- results are quite consistent also when looking at the quality of the best Shallow (XGB) and Deep (TCN) algorithms on the different sections of the mooring lines;
- it is interesting to note that the 2nd section, the central one, seems to be consistently the harder one to be able to predict correctly with the higher %ER.

Table 4
(Scenario 1A): %ER, %FP, and %FN, averaged on the mooring lines and sections, for the different shallow (LSVM, KSVM, RF, and XGB) and deep (TCN) algorithms employed.

Algorithm		%ER	%FP	%FN
Shallow	LSVM	7.4 ±1.1	7.4 ±1.1	7.3 ±1.2
	KSVM	4.4 ±0.6	4.4 ±0.7	4.3 ±0.6
	RF	3.1 ±0.5	3.2 ±0.5	3.0 ±0.5
	XGB	2.1 ±0.3	2.0 ±0.3	2.2 ±0.3
Deep	TCN	1.0 ±0.2	1.0 ±0.2	1.0 ±0.2

Table 5
(Scenario 1A): %ER, %FP, and %FN, for each mooring line and each section, for the best shallow (XGB) and deep (TCN) algorithms employed.

Mooring Line	Section	XGB %ER	XGB %FP	XGB %FN	TCN %ER	TCN %FP	TCN %FN
1	1	1.8 ±0.2	1.8 ±0.2	1.8 ±0.2	0.9 ±0.1	0.9 ±0.1	0.8 ±0.1
	2	2.1 ±0.3	1.8 ±0.2	2.4 ±0.4	1.1 ±0.2	1.0 ±0.2	1.1 ±0.2
	3	2.2 ±0.4	1.9 ±0.3	2.6 ±0.4	1.0 ±0.1	0.9 ±0.1	1.0 ±0.1
2	1	2.1 ±0.3	1.9 ±0.2	2.3 ±0.4	0.9 ±0.1	0.8 ±0.1	0.9 ±0.1
	2	2.5 ±0.3	2.3 ±0.3	2.8 ±0.4	1.3 ±0.2	1.3 ±0.2	1.3 ±0.2
	3	1.9 ±0.2	2.0 ±0.2	1.7 ±0.3	0.9 ±0.2	0.8 ±0.2	1.0 ±0.2
3	1	2.1 ±0.4	2.1 ±0.4	2.1 ±0.3	1.1 ±0.1	1.0 ±0.1	1.1 ±0.1
	2	2.6 ±0.4	2.6 ±0.3	2.5 ±0.4	1.2 ±0.1	1.3 ±0.2	1.0 ±0.1
	3	1.7 ±0.2	1.6 ±0.2	1.8 ±0.2	1.1 ±0.2	1.1 ±0.1	1.0 ±0.2

Table 6
(Scenario 1B): %ER, %FP, and %FN, averaged on the mooring lines and sections, for the different shallow (LSVM, KSVM, RF, and XGB) and deep (TCN) algorithms employed.

Algorithm		%ER	%FP	%FN
Shallow	LSVM	12.0 ±2.5	12.0 ±2.6	12.1 ±2.3
	KSVM	7.5 ±1.6	7.5 ±1.5	7.5 ±1.7
	RF	5.5 ±1.1	5.4 ±1.1	5.6 ±1.2
	XGB	4.3 ±0.9	4.3 ±0.9	4.2 ±1.0
Deep	TCN	3.1 ±0.7	3.1 ±0.6	3.1 ±0.7

4.1.2. Neglecting tension and tension angle measurements (Scenario 1B)

Let us consider Section 1B.

Table 6 reports, analogously to Table 4, the quality (in terms of %ER, %FP, and %FN) averaged on the mooring lines and sections of the different shallow (LSVM, KSVM, RF, and XGB) and deep (TCN) algorithms described in Section 3.1.

Table 7 reports, analogously to Table 5, the quality (in terms of %ER, %FP, and %FN) for each mooring line and each section, for the best shallow (XGB) and deep (TCN) algorithms employed. The best Shallow and Deep algorithms have been chosen in accordance with Table 4 considering the metric %ER.

From Tables 6 and 7, it is possible to derive comments quite similar to the ones derived from Tables 4 and 5. The only difference is in the slight decrease in performance with respect to the one obtained in Section 1 A. This, as discussed in the previous sections, is not surprising since we are removing information to make the desired predictions (*i.e.*, tension and tension angle time series). What is interesting at this point is to evaluate the performance loss obtained by removing this information.

For this reason, in Table 8 a comparison between Scenario 1 A versus Scenario 1B is reported. In particular, we reported the quality (in terms of %ER, %FP, and %FN) averaged on the mooring lines and sections, for the best shallow (XGB) and deep (TCN) algorithms employed in each of the two scenarios under consideration.

From Table 8, it is possible to observe that the loss in performance is not negligible (*i.e.*, we go from $\approx 2\%$ to $\approx 4\%$ of %ER for XGB and from $\approx 1\%$ to $\approx 3\%$ for TCN) but the resulting performance is, for sure, acceptable in a practical application taking into account that we remove the necessity of installing and maintaining other components (*i.e.*, the tension and tension angle sensors).

4.2. Weakly supervised: from Shallow to Deep Models (Scenario 2)

This section is devoted to the experimental evaluation of Shallow and Deep Models in Scenarios 2 (see Section 3), the weakly supervised one, testing the methodology presented in Section 3.2 leveraging the data described in Section 2.

Note that Scenario 2 is, in some sense, an evolution of Scenario 1B. In fact, analogously to Scenario 1B, in Scenario 2, we do not leverage information coming from costly sensors with maintenance needs and hard to deploy (*i.e.*, information about tension and

Table 7
(Scenario 1B): %ER, %FP, and %FN, for each mooring line and each section, for the best shallow (XGB) and deep (TCN) algorithms employed.

Mooring Line	Section	XGB %ER	XGB %FP	XGB %FN	TCN %ER	TCN %FP	TCN %FN
1	1	3.8 ±0.6	4.0 ±0.6	3.6 ±0.6	2.9 ±0.6	2.8 ±0.5	2.9 ±0.8
	2	4.9 ±1.2	5.2 ±1.2	4.7 ±1.3	3.3 ±0.5	3.5 ±0.5	3.0 ±0.5
	3	3.7 ±0.9	3.5 ±1.0	4.0 ±0.9	3.2 ±0.6	3.0 ±0.6	3.4 ±0.6
2	1	3.7 ±0.7	3.8 ±0.7	3.7 ±0.8	2.5 ±0.5	2.8 ±0.4	2.3 ±0.6
	2	4.7 ±0.8	4.9 ±0.8	4.5 ±0.9	4.0 ±0.8	3.7 ±0.6	4.4 ±0.9
	3	4.2 ±1.2	4.3 ±1.2	4.2 ±1.3	2.7 ±0.6	2.6 ±0.8	2.8 ±0.5
3	1	4.1 ±0.9	3.8 ±0.7	4.4 ±1.2	2.7 ±0.7	3.2 ±0.8	2.3 ±0.6
	2	5.1 ±1.0	5.5 ±1.0	4.7 ±0.9	3.6 ±1.0	3.6 ±1.0	3.7 ±0.9
	3	4.1 ±1.0	3.9 ±1.0	4.2 ±1.0	2.9 ±0.5	3.0 ±0.4	2.9 ±0.6

Table 8
Scenario 1A versus Scenario 1B: %ER, %FP, and %FN, averaged on the mooring lines and sections, for the best shallow (XGB) and deep (TCN) algorithms employed.

Algorithm	Scenario	%ER	%FP	%FN
Best Shallow (XGB)	1A	2.1 ±0.3	2.0 ±0.3	2.2 ±0.3
	1B	4.3 ±0.9	4.3 ±0.9	4.2 ±1.0
Deep (TCN)	1A	1.0 ±0.2	1.0 ±0.2	1.0 ±0.2
	1B	3.1 ±0.7	3.1 ±0.6	3.1 ±0.7

tension angle) to make predictions. The problem is further complicated because we have just a minimal amount of labeled data, and most of the samples are from healthy mooring line sections. Instead, the goal is the same as Scenarios 1 A and 1B, namely, predicting whether the chain diameter reduction is above or below 10%.

Note that in this case, $t = 0.1n$, namely 10% of the data has been used for the test. υ, namely how much data has been used for validation, has been described later in this section while the rest of the data has been used for learning (note that in this scenario in the learning set, only healthy data is present).

At this point, we can report the results.

Table 9 reports the quality of the different shallow (OCSVM and GKNN) and deep (TCN+OCSVM and TCN+GKNN) algorithms averaged on the mooring lines and sections described in Section 3.2 varying the number of labeled samples υ employed in the validation set $\mathcal{V}_\upsilon$.

From Table 9, it is possible to observe that

- the Shallow models and the TCN+GKNN, in this case, are not able to achieve satisfying performance with %ER $\gg 5\%$;
- TCN+OCSVM, when a reasonable amount of labeled samples (*i.e.*, $\upsilon = 60$ or $\upsilon = 90$) is available, are able to achieve satisfying performance of %ER $\approx 4\%$. When the amount of labeled samples is too low (*i.e.*, $\upsilon = 30$) also TCN+OCSVM is not able to achieve and %ER $\leq 5\%$;
- also, in this case, the errors are generally consistent, but with respect to Scenarios 1 A and 1B, we can see that there is, generally, a higher value of the %FP. This is not surprising as in the training set, most of the samples in D_n are coming from healthy mooring line sections $Y = -1$;
- all the models based on GKNN (*i.e.*, shallow GKNN and deep TCN+GKNN) are consistently underperforming the one based on OCSVM (*i.e.*, shallow OCSVM and deep TCN+OCSVM); Also in this case, the results are not surprising since, as discussed in Section 3.2, OCSVM have shown in the past to be much more powerful than GKNN.

To further confirm these observations, we report in Table 10, the quality of the best shallow (OCSVM) and best deep (TCN+OCSVM) algorithms employed for each mooring line and each section with $\upsilon = 60$ labeled samples in the validation set $\mathcal{V}_\upsilon$. The value $\upsilon = 60$ has been chosen as the best compromise between a realistic amount of labeled data for practical application and the performance of the model. The best shallow and deep algorithms have been chosen in accordance with Table 9 considering the metric %ER.

From Table 10, it is possible to observe that

- results are quite consistent over the different sections of the mooring lines, but, as noted for Tables 5 and 7, the 2nd section of each mooring line results to be the harder to estimate correctly;
- the %FP is usually above the one of the %FN but, at least form the TCN+OCSVM %FP is always $\approx 5\%$ so acceptable from a practical point of view.

Moreover, analogously to what happened when moving from Scenario 1 A to Scenario 1B, moving from Scenario 1B to Scenario 2 implies a decrease in performance which is, in this case, a little more noticeable. This, as discussed in the previous sections, is again not surprising since we are removing information to make the desired predictions (*i.e.*, tension and tension angle time series) but also labeled samples to the algorithms. Also, in this case, it is interesting to analyze better the performance loss obtained by further removing information from the algorithm.

Table 9
(Scenario 2): %ER, %FP, and %FN, averaged on the mooring lines and sections, for the different shallow (OCSVM and GKNN) and deep (TCN+OCSVM and TCN+GKNN) algorithms employed varying the number of labeled samples υ employed in the validation set $\mathcal{V}_\upsilon$.

υ	Algorithm		%ER	%FP	%FN
30	Shallow	OCSVM	10.8 ±3.5	10.1 ±3.0	11.5 ±3.9
		GKNN	15.5 ±5.1	14.0 ±4.6	17.1 ±5.6
	Deep	TCN+OCSVM	5.7 ±1.6	4.9 ±1.3	6.4 ±1.8
		TCN+GKNN	9.8 ±3.0	8.8 ±2.8	10.7 ±3.3
60	Shallow	OCSVM	8.3 ±2.4	7.5 ±2.1	9.0 ±2.7
		GKNN	12.2 ±3.9	11.4 ±3.8	13.0 ±4.0
	Deep	TCN+OCSVM	4.3 ±1.3	3.9 ±1.2	4.7 ±1.4
		TCN+GKNN	7.3 ±2.3	6.7 ±2.1	7.9 ±2.5
90	Shallow	OCSVM	6.8 ±1.8	6.3 ±1.7	7.2 ±1.8
		GKNN	9.5 ±2.6	8.5 ±2.1	10.4 ±3.0
	Deep	TCN+OCSVM	3.4 ±1.0	3.2 ±0.9	3.6 ±1.1
		TCN+GKNN	6.0 ±1.9	5.4 ±1.6	6.7 ±2.1

Table 10
(Scenario 2): %ER, %FP, and %FN, for each mooring line and each section, for the best shallow (OCSVM) and best deep (TCN+OCSVM) algorithms employed with $\upsilon = 60$ labeled samples in the validation set $\mathcal{V}_\upsilon$.

Mooring Line	Section	OCSVM %ER	OCSVM %FP	OCSVM %FN	TCN+OCSVM %ER	TCN+OCSVM %FP	TCN+OCSVM %FN
1	1	7.9 ±1.9	8.2 ±1.7	7.6 ±2.1	4.0 ±1.0	4.2 ±1.1	3.8 ±0.9
	2	8.4 ±2.1	7.2 ±2.3	9.7 ±2.0	4.7 ±1.6	4.5 ±1.3	5.0 ±2.0
	3	7.4 ±2.0	6.0 ±1.8	8.7 ±2.3	4.2 ±1.2	4.2 ±0.9	4.2 ±1.5
2	1	7.6 ±2.3	7.7 ±1.6	7.6 ±2.9	4.2 ±1.3	3.3 ±1.2	5.1 ±1.4
	2	9.2 ±3.0	8.4 ±2.1	10.0 ±3.9	4.9 ±1.6	4.0 ±1.2	5.9 ±2.0
	3	7.7 ±2.1	7.1 ±2.2	8.3 ±2.0	3.7 ±1.1	3.5 ±1.4	3.9 ±0.9
3	1	8.1 ±2.2	7.4 ±2.6	8.8 ±1.8	4.0 ±1.1	3.7 ±1.2	4.3 ±1.0
	2	10.2 ±3.3	8.8 ±2.5	11.6 ±4.2	5.1 ±1.8	4.3 ±1.3	6.0 ±2.3
	3	8.1 ±2.7	7.1 ±2.0	9.0 ±3.5	3.8 ±0.9	3.2 ±0.9	4.4 ±0.9

Table 11
Scenario 1A versus Scenario 1B versus Scenario 2: %ER, %FP, and %FN, averaged on the mooring lines and sections, for the best shallow (XGB for Scenarios 1A and 1B and OCSVM for Scenario 2) and best deep (TCN for Scenarios 1A and 1B and TCN+OCSVM for Scenario 2) algorithms employed. For Scenario 2 $\upsilon = 60$ labeled samples in the validation set $\mathcal{V}_\upsilon$.

Algorithm		Scenario	%ER	%FP	%FN
Shallow	XGB	1A	2.1 ±0.3	2.0 ±0.3	2.2 ±0.3
		1B	4.3 ±0.9	4.3 ±0.9	4.2 ±1.0
	OCSVM	2	8.3 ±2.4	7.5 ±2.1	9.0 ±2.7
Deep	TCN	1A	1.0 ±0.2	1.0 ±0.2	1.0 ±0.2
		1B	3.1 ±0.7	3.1 ±0.6	3.1 ±0.7
	TCN+OCSVM	2	4.3 ±1.3	3.9 ±1.2	4.7 ±1.4

For this reason, in Table 11, a comparison between Scenario 1 A, Scenario 1B, and Scenario 2 is reported. In particular, we reported the quality of the best shallow and best deep algorithms employed in each of the three scenarios under consideration averaged on the mooring lines and sections. For Scenario 2 $\upsilon = 60$ labeled samples in the validation set $\mathcal{V}_\upsilon$.

From Table 11, it is possible to observe that the loss in performance is, in this case, consistent

- for shallow models we go from $\approx 2\%$ in Scenario 1 A to $\approx 4\%$ in Scenario 1B of %ER for XGB and to $\approx 8\%$ in Scenario 2 for OCSVM
- for deep models we go from $\approx 1\%$ in Scenario 1 A to $\approx 3\%$ in Scenario 1B of %ER and to $\approx 4\%$ in Scenario 2 for TCN-OCSVM with a slightly higher %FN with respect to the %FP.

Nevertheless the resulting performance are, for sure, acceptable in a practical application $\approx 5\%$ taking into account that we remove the necessity of installing and maintaining other components (*i.e.*, the tension and tension angle sensors) and we limited the necessity for labeled samples that are seldom available in practical applications.

5. Conclusions

In this paper, we proposed exploiting data from low-cost, low-maintenance, and easily deployable sensors (*e.g.*, tower accelerometers) for mooring systems health status nowcasting based on Deep machine learning models considering a very small amount of data regarding worn mooring lines.

In line with the growing need for renewable energy production, the global installed capacity of floating offshore wind turbines is projected to increase by at least 100 times over the next decades. Station-keeping of floating offshore renewable energy devices is achieved through the use of mooring systems, which are exposed to a variety of environmental and operational conditions causing corrosion, abrasion, and fatigue. Regular physical in-service inspections of mooring systems for monitoring their health status are, still nowadays, the golden standard but they are often expensive, inefficient, and unsafe, and for this reason, digital solutions for real-time monitoring are required. Floating offshore renewable energy devices are usually equipped with a wide range of sensors, some low-cost, low/zero maintenance, and easily deployable (*e.g.*, accelerometers on the tower) contrary to others (*e.g.*, direct tension mooring line measurements), producing real-time data streams that can be effectively leveraged by machine learning models for real-time monitoring. Of course, being able to perform this monitoring employing just the low-cost, low-maintenance, and easily deployable sensors is a plus. Moreover, first relying on state-of-the-art supervised Shallow and Deep learning models for predicting the health status of the different sections of the mooring lines we showed that it is possible to perform real-time monitoring with just low-cost, low-maintenance, and easily deployable sensors. Then, since these supervised models require types and amounts of data that are seldom available, we proposed new Shallow and Deep weekly supervised models that require a very small amount of data regarding worn mooring lines. Results showed that these last models have the potential to have practical applicability and impact for real-time monitoring of mooring systems in the near future. In order to support our statements, we made use of data generated with a state-of-the-art digital twin of the mooring system, OrcaFlex,[1] for a floating offshore wind turbine reproducing the physical mechanism of the mooring degradation under different loads and environmental conditions. Results have shown errors around 1% in the simplest scenario (fully supervised with all the information available) and errors around 4% (weakly supervised one with only cheap information available) in the most challenging one, confirming the potentiality of the proposed approaches. It is also worth noting that some limitations are still present. For instance, even though the sensors are low-maintenance, they still need to be checked and calibrated regularly to ensure accurate data collection. Moreover, the sensors might be susceptible to environmental influences (like temperature, humidity, or saltwater corrosion), which could affect their performance and longevity. In the future, we plan to collaborate with the industry to check the quality of the proposal in a real environment, increasing the readiness level of this technology.

CRediT authorship contribution statement

Andrea Coraddu: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Luca Oneto:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Jake Walker:** Investigation, Validation, Visualization, Writing – original draft, Writing – review & editing. **Katarzyna Patryniak:** Investigation, Methodology, Visualization, Writing – original draft. **Arran Prothero:** Data curation, Investigation, Methodology, Resources. **Maurizio Collu:** Conceptualization, Formal analysis, Methodology, Supervision, Validation, Writing – original draft.

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.

Data availability

Data will be made available on request.

References

[1] Global Wind Energy Council, Global wind report 2021, 2021, https://gwec.net/global-wind-report-2021/.
[2] Scottish Renewables, Floating wind the UK industry ambition, 2019, https://www.scottishrenewables.com/publications/522-floating-wind-the-uk-industry-ambition.
[3] ORE Catapult, Mooring and anchoring systems - market projections, 2021, https://ore.catapult.org.uk/?orecatapultreports=mooring-anchoring-systems-market-projections.
[4] Carbon Trust, Floating wind joint industry project - summary report phase 1, 2018, https://www.carbontrust.com/resources/floating-wind-joint-industry-project-summary-report-phase-1.
[5] I. Maria, L. Mattias, D. Siobhan, B. Friedemann, et al., Review of the state of the art of mooring and anchoring designs, technical challenges and identification of relevant DLCs, 2020, https://corewind.eu/wp-content/uploads/files/publications/COREWIND-D2.1-Review-of-the-state-of-the-art-of-mooring-and-anchoring-designs.pdf.
[6] Carbon Trust, Floating wind joint industry project - summary report phase 2, 2020, https://www.carbontrust.com/resources/floating-wind-joint-industry-project-phase-2-summary-report.
[7] G. Ma, L. Zhong, X. Zhang, Q. Ma, H.S. Kang, Mechanism of mooring line breakage of floating offshore wind turbine under extreme coherent gust with direction change condition, J. Marine Sci. Technol. 25 (4) (2020) 1283–1295.
[8] M.G. Brown, T.D. Hall, D.G. Marr, M. English, R.O. Snell, Floating production mooring integrity JIP - key findings, in: Annual Offshore Technology Conference, 2005.
[9] S. Butterfield, W. Musial, J. Jonkman, P. Sclavounos, L. Wayman, Engineering challenges for floating offshore wind turbines, in: Offshore Wind Conference, 2007.
[10] A.L. Yaghin, R.E. Melchers, Long-term inter-link wear of model mooring chains, Mar. Struct. 44 (2015) 61–84.
[11] F.G. Rivera, G. Edwards, E. Eren, S. Soua, Acoustic emission technique to monitor crack growth in a mooring chain, Appl. Acoust. 139 (2018) 156–164.

[12] R. Gordon, M. Brown, E. Allen, Mooring integrity management: A state-of-the-art review, in: Offshore Technology Conference, Houston, Texas, May 2014, 2014.
[13] A. Angulo, G. Edwards, S. Soua, T.H. Gan, Mooring integrity management: Novel approaches towards in situ monitoring, in: Structural Health Monitoring, 2017.
[14] ABSG Consulting, Study on mooring system integrity management for floating structures, 2015, https://www.bsee.gov/sites/bsee.gov/files/tap-technical-assessment-program//730-aa.pdf.
[15] K. Ma, H. Shu, P. Smedley, D. L´Hostis, A. Duggal, A historical review on integrity issues of permanent mooring systems, in: Offshore Technology Conference, 2013.
[16] M. Borg, M. Collu, A. Kolios, Offshore floating vertical axis wind turbines, dynamics modelling state of the art. Part II: Mooring line and structural dynamics, Renew. Sustain. Energy Rev. 39 (2014) 1226–1234.
[17] L. Oneto, A. Coraddu, F. Cipollini, O. Karpenko, K. Xepapa, P. Sanetti, D. Anguita, Crash stop maneuvering performance prediction: a data-driven solution for safety and collision avoidance, Data-Enabled Discover. Appl. 2 (1) (2018) 1–11.
[18] J. Walker, A. Coraddu, M. Collu, L. Oneto, Digital twins of the mooring line tension for floating offshore wind turbines to improve monitoring, lifespan, and safety, J. Ocean Eng. Mar. Energy 8 (1) (2022) 1–16.
[19] J. Walker, A. Coraddu, L. Oneto, S. Kilbourn, Digital twin of the mooring line tension for floating offshore wind turbines, in: OCEANS, 2021.
[20] H. Elgamiel, N. Markov, V. Grinius, M. Lambert, New mooring simulation approaches in model testing, in: International Ocean and Polar Engineering Conference, 2006.
[21] S. Maroju, K. Delaney, C. Leon, I. Prislin, Estimation of critical platform integrity parameters in the absence of direct measurements in the context of integrated marine monitoring systems, in: International Conference on Ocean, Offshore and Arctic Engineering, 2013.
[22] ORE Catapult, Equinor hywind Scotland windfarm, 2018, https://pod.ore.catapult.org.uk/source/equinor-hywind-scotland-windfarm.
[23] Scanmatic, Mooring line integrity monitoring, 2022, https://www.scanmatic.no/project/mooring-line-integrity-monitoring/?lang=en.
[24] Acteon, Mooring line integrity monitoring, 2022, https://acteon.com/products-services/mooring-line-integrity-monitoring/.
[25] Health and Safety Executive, Mooring failure detection systems for floating offshore installations. mooring integrity joint industry project phase 2, 2017, https://www.hse.gov.uk/research/rrpdf/rr1097.pdf.
[26] Carbon Trust, Floating wind joint industry project - summary report phase 3, 2021, https://www.carbontrust.com/resources/floating-wind-joint-industry-project-phase-iii-summary-report.
[27] Seatools, MoorMate –mooring monitoring system, 2019, https://www.seatools.com.
[28] C. Barrera, T. Battistella, R. Guanche, I.J. Losada, Mooring system fatigue analysis of a floating offshore wind turbine, Ocean Eng. 195 (2020) 106670.
[29] Det Norske Veritas, Offshore standards — DNV-OS-e301. Edition july 2021. Position mooring., 2021, https://www.dnv.com/maritime/Offshore/technical-guidance-otg.html.
[30] J. Jonkman, M. Buhl, Loads analysis of a floating offshore wind turbine using fully coupled simulation, 2007, https://www.nrel.gov/docs/fy07osti/41714.pdf.
[31] M. Hall, A. Goupee, Validation of a lumped-mass mooring line model with DeepCwind semisubmersible model test data, Ocean Eng. 104 (2015) 590–603.
[32] P. Trubat, C. Molins, X. Gironella, Quasi-dynamic mooring line model, Ocean Eng. 243 (2022).
[33] W. Zhong, N. Wang, D. Wan, A coupled CFD and dynamic mooring model for FOWT hydrodynamics, in: International Ocean and Polar Engineering Conference, 2022.
[34] T. Takeuchi, T. Utsunomiya, K. Gotoh, I. Sato, Quantitative wear estimation for floating structures by using 3-d geometry of mooring chain, in: International Conference on Ocean, Offshore and Arctic Engineering, 2020.
[35] H. Hertz, On the contact of rigid elastic solids and on hardness, in: Ch 6: Assorted Papers, MacMillan, 1882.
[36] T. Takeuchi, T. Utsunomiya, K. Gotoh, I. Sato, Development of interlink wear estimation method for mooring chain of floating structures: Validation and new approach using three-dimensional contact response, Mar. Struct. 77 (2021) 102927.
[37] A. Angulo, J. Allwrighta, C. Maresb, T. Gana, S. Souaa, Finite element analysis of crack growth for structural health monitoring of mooring chains using ultrasonic guided waves and acoustic emission, in: International Conference on Structural Integrity, 2015.
[38] M. Rivas-Lopez, W. Flores, O. Sergiyenko, Structural Health Monitoring: Measurement Methods and Practical Applications, BoD, 2017.
[39] Joint Industry Project Steering Committee, et al., Practical method for calculating mooring chain wear for floating offshore installations mooring integrity joint industry project phase 2, 2017, https://www.fabig.com/external-publications/hse-rr1092.
[40] A.C. de Pina, A.A. de Pina, C.H. Albrecht, B.S. Leite Pires de Lima, B.P. Jacob, ANN-based surrogate models for the analysis of mooring lines and risers, Appl. Ocean Res. 41 (2013) 76–86.
[41] C.B. Li, J. Choung, M.H. Noh, Wide-banded fatigue damage evaluation of catenary mooring lines using various artificial neural networks models, Mar. Struct. 60 (2018) 186–200.
[42] P. Li, C. Jin, G. Ma, J. Yang, L. Sun, Evaluation of dynamic tensions of single point mooring system under random waves with artificial neural network, J. Marine Sci. Eng. 10 (5) (2022) 666.
[43] W.T. Hsu, K.P. Thiagarajan, M. Hall, M. MacNicoll, R. Akers, Snap loads on mooring lines of a floating offshore wind turbine structure, in: International Conference on Offshore Mechanics and Arctic Engineering, 2014.
[44] W.T. Hsu, K.P. Thiagarajan, M. MacNicoll, R. Akers, Prediction of extreme tensions in mooring lines of a floating offshore wind turbine in a 100-year storm, in: International Conference on Offshore Mechanics and Arctic Engineering, 2015.
[45] W.T. Hsu, K.P. Thiagarajan, L. Manuel, Extreme mooring tensions due to snap loads on a floating offshore wind turbine system, Mar. Struct. 55 (2017) 182–199.
[46] Z. Mousavi, S. Varahram, M. Ettefagh, M. Sadeghi, S. Razavi, Deep neural networks–based damage detection using vibration signals of finite element model and real intact state: An evaluation via a lab-scale offshore jacket structure, Struct. Health Monit. 20 (1) (2021) 379–405.
[47] Z. Mousavi, S. Varahram, M. Mohammad Ettefagh, M. Sadeghi, Dictionary learning-based damage detection under varying environmental conditions using only vibration responses of numerical model and real intact state: Verification on an experimental offshore jacket model, Mech. Syst. Signal Process. 182 (2023) 109567.
[48] K. Sun, Z. Xu, S. Li, J. Jin, P. Wang, M. Yue, C. Li, Dynamic response analysis of floating wind turbine platform in local fatigue of mooring, Renew. Energy 204 (2023) 733–749.
[49] I. Prislin, S. Maroju, Mooring integrity and machine learning, in: Annual Offshore Technology Conference, 2017.
[50] V. Jaiswal, A. Ruskin, Mooring line failure detection using machine learning, in: Annual Offshore Technology Conference, 2019.
[51] A. Arredondo, J. Altuzarra, A. Mena, J. Fernández, Stress intensity factors in fitness-for-service assessment of cracks in mooring chains, in: International Conference on Offshore Mechanics and Arctic Engineering, 2018.
[52] M. Chung, S. Kim, K. Lee, D.H. Shin, Detection of damaged mooring line based on deep neural networks, Ocean Eng. 209 (2020) 107522.
[53] Carbon Trust, Condition monitoring of floating wind mooring lines, 2021, https://ctprodstorageaccountp.blob.core.windows.net/prod-drupal-files/documents/resource/public/Fugro_ConditionMonitoring.pdf.
[54] Y. Bai, Marine structural design, Elsevier, 2003.
[55] S.A. Silling, R.B. Lehoucq, Peridynamic theory of solid mechanics, in: Advances in Applied Mechanics, 2010.

[56] J. Schulte-Tigges, M. Förster, G. Nikolovski, M. Reke, A. Ferrein, D. Kaszner, D. Matheis, T. Walter, Benchmarking of various LiDAR sensors for use in self-driving vehicles in real-world environments, Sensors 22 (19) (2022) 7146.
[57] International Electrotechnical Commision, Wind energy generation systems–Part 3-2: Design requirements for floating offshore wind turbines, https://webstore.iec.ch/publication/29244.
[58] T. Hørte, S. Okkenhaug, Recommendations for a fatigue design analysis calibrated using structural reliability analysis, in: International Conference on Offshore Mechanics and Arctic Engineering, 2022.
[59] Det Norske Veritas, DNV offshore standards - position mooring, 2021, https://global.ihs.com/doc_detail.cfm?document_name=DNV%2DOS%2DE301&item_s_key=00398869.
[60] American Petroleum Institute, In-service inspection of mooring hardware for floating structures; api recommended practice 2I, 2020, https://global.ihs.com/doc_detail.cfm?document_name=API%20RP%202I&item_s_key=00129180.
[61] S. Shalev-Shwartz, S. Ben-David, Understanding machine learning: From theory to algorithms, Cambridge University Press, 2014.
[62] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.
[63] G. Pang, C. Shen, L. Cao, A.V.D. Hengel, Deep learning for anomaly detection: A review, ACM Comput. Surv. 54 (2) (2021) 1–38.
[64] A. Boukerche, L. Zheng, O. Alfandi, Outlier detection: Methods, models, and classification, ACM Comput. Surv. 53 (3) (2020) 1–37.
[65] M. Kuhn, K. Johnson, Feature engineering and selection: A practical approach for predictive models, CRC Press, 2019.
[66] P. Duboue, The art of feature engineering: essentials for machine learning, Cambridge University Press, 2020.
[67] C.C. Aggarwal, Neural networks and deep learning, Springer, 2018.
[68] J. Shawe-Taylor, N. Cristianini, Kernel methods for pattern analysis, Cambridge University Press, 2004.
[69] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: ACM Sigkdd International Conference on Knowledge Discovery and Data Mining, 2016.
[70] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[71] S. Theodoridis, Machine learning: a Bayesian and optimization perspective, Academic Press, 2015.
[72] J. Lever, Classification evaluation: It is important to understand both what a classification metric expresses and what it hides, Nat. Methods 13 (8) (2016) 603–605.
[73] L. Swersky, H.O. Marques, J. Sander, R.J.G.B. Campello, A. Zimek, On the evaluation of outlier detection and one-class classification methods, in: IEEE International Conference on Data Science and Advanced Analytics, 2016.
[74] L. Oneto, Model Selection and Error Estimation in a Nutshell, Springer, 2019.
[75] A. Coraddu, L. Oneto, D. Ilardi, S. Stoumpos, G. Theotokatos, Marine dual fuel engines monitoring in the wild through weakly supervised data analytics, Eng. Appl. Artif. Intell. 100 (2021) 104179.
[76] F. Cipollini, L. Oneto, A. Coraddu, A.J. Murphy, D. Anguita, Condition-based maintenance of naval propulsion systems: Data analysis with minimal feedback, Reliab. Eng. Syst. Saf. 177 (2018) 12–23.
[77] A. Zheng, A. Casari, Feature engineering for machine learning: principles and techniques for data scientists, O'Reilly Media, Inc., 2018.
[78] J.D. Hamilton, Time series analysis, Princeton University Press, 2020.
[79] J.L. Reyes-Ortiz, L. Oneto, A. Sama, X. Parra, D. Anguita, Transition-aware human activity recognition using smartphones, Neurocomputing 171 (2016) 754–767.
[80] N.D. Cilia, C. De Stefano, F. Fontanella, A. Scotto, A ranking-based feature selection approach for handwritten character recognition, Pattern Recognit. Lett. 121 (2019) 77–86.
[81] V. D'Amato, E. Volta, L. Oneto, G. Volpe, A. Camurri, D. Anguita, Understanding violin players' skill level based on motion capture: a data-driven perspective, Cogn. Comput. 12 (6) (2020) 1356–1369.
[82] A. Roy, B. Banerjee, A. Hussain, S. Poria, Discriminative dictionary design for action classification in still images and videos, Cogn. Comput. 13 (2021) 698–708.
[83] V. D'Amato, L. Oneto, A. Camurri, D. Anguita, Z. Zarandi, L. Fadiga, A. D'Ausilio, T. Pozzo, The importance of multiple temporal scales in motion recognition: from shallow to deep multi scale models, in: IEEE International Joint Conference on Neural Networks, 2022.
[84] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, Do we need hundreds of classifiers to solve real world classification problems? J. Machine Learn. Res. 15 (1) (2014) 3133–3181.
[85] M. Wainberg, B. Alipanahi, B.J. Frey, Are random forests truly the best classifiers? J. Mach. Learn. Res. 17 (1) (2016) 3837–3841.
[86] L. Van Der Maaten, E. Postma, J. Van den Herik, Dimensionality reduction: a comparative, J. Mach. Learn. Res. 10 (66–71) (2009) 13.
[87] A. Fisher, C. Rudin, F. Dominici, All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously., J. Mach. Learn. Res. 20 (177) (2019) 1–81.
[88] C. Molnar, Interpretable machine learning, Lulu.com, 2020.
[89] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, X. Xie, Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks, in: AAAI Conference on Artificial Intelligence, 2016.
[90] J. Shi, D. Peng, Z. Peng, Z. Zhang, K. Goebel, D. Wu, Planetary gearbox fault diagnosis using bidirectional-convolutional LSTM networks, Mech. Syst. Signal Process. 162 (2022) 107996.
[91] V. D'Amato, L. Oneto, A. Camurri, D. Anguita, The importance of multiple temporal scales in motion recognition: when shallow model can support deep multi scale models, in: IEEE International Joint Conference on Neural Networks, IJCNN, 2022.
[92] S. Bai, J.Z. Kolter, V. Koltun, An empirical evaluation of generic convolutional and recurrent networks for sequence modeling, 2018, arXiv preprint arXiv:1803.01271.
[93] S.M. Lee, S.M. Yoon, H. Cho, Human activity recognition from accelerometer data using convolutional neural network, in: IEEE International Conference on Big Data and Smart Computing, 2017.
[94] J. Zhan, C. Wu, X. Ma, C. Yang, Q. Miao, S. Wang, Abnormal vibration detection of wind turbine based on temporal convolution network and multivariate coefficient of variation, Mech. Syst. Signal Process. 174 (2022) 109082.
[95] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, 2015, arXiv preprint arXiv:1511.07122.
[96] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization., J. Mach. Learn. Res. 13 (2) (2012).
[97] S. Ramaswamy, R. Rastogi, K. Shim, Efficient algorithms for mining outliers from large data sets, in: ACM SIGMOD International Conference on Management of Data, 2000.
[98] M.A. Kramer, Nonlinear principal component analysis using autoassociative neural networks, AIChE J. 37 (2) (1991) 233–243.