Integrated damage detection and time-series data augmentation for floating offshore mooring systems via variational semi-supervised learning

Ocean Engineering 330 (2025) 121199

Contents lists available at ScienceDirect

Ocean Engineering

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

Research paper

Integrated damage detection and time-series data augmentation for floating offshore mooring systems via variational semi-supervised learning

Pranjal Tamuly [a,*], Smriti Sharma [a], Vincenzo Nava [b]

[a] Basque Center for Applied Mathematics, Alameda de Mazarredo, 14, Bilbao, 48009, Spain
[b] Politecnico di Torino, Corso Duca degli Abruzzi, 24, Italy

ARTICLE INFO

Keywords:
Offshore structures
Damage diagnosis
Floating wind turbines
Mooring systems
Semi-supervised learning
Variational autoencoder

ABSTRACT

The dynamics and stability of the semi-submersible offshore platforms are significantly impacted by the degradation of the mooring system. Identifying structural integrity issues in mooring systems through a data-driven approach is challenging due to the infrequency of damage events and the difficulties in recording them. To address these challenges, this study proposes the Time-Series Variational Semi-Supervised Learning (TSVSSL) framework, which effectively bridges the gap between supervised and unsupervised learning by leveraging unlabelled data for damage detection. The proposed framework features a distinctive training procedure in which the encoder-decoder and classifier components are trained concurrently. This process produces a well-clustered latent representation that enhances damage detection and supports class-specific artificial data generation. A numerical study using simulated responses of a 5 MW semi-submersible FOWT under varying metocean conditions demonstrated that the proposed framework outperformed existing deep learning methods in damage detection, achieving superior accuracy, precision, recall, and F1 score. Further, a rejection sampling technique is also introduced to effectively generate artificial data that closely aligns with actual time series displacement response. The novelty of the proposed framework lies in its dual focus on damage detection and artificial data generation marking a significant advancement in the data-driven assessment of mooring systems.

1. Introduction

Offshore wind turbines possess an enormous potential for massive renewable energy production, contributing significantly to power generation for the electric grid. The offshore renewable energy sector has experienced significant growth in Europe in recent years, which is evident in the increase in capacity from 8503 MW in 2014 to 73,185 MW in 2023, according to International Renewable Energy Agency (IRENA) (2024). Thus, ensuring the safety and maintenance of these wind turbines is of utmost importance to minimise downtime and unexpected failures. Therefore, Structural health monitoring (SHM) techniques are employed to optimise repair and maintenance strategies to ensure credible power generation. However, monitoring these FOWTs remains a challenging task due to their difficult accessibility, which poses a significant logistical bottleneck (Duarte et al., 2018; Myhr et al., 2014).

SHM involves solving an inverse problem to evaluate the condition of structural components. In this context, data-driven approaches have been recognised for inverse estimation structural integrity. However, complex inverse problems often present several challenges, including non-unique solutions and inaccuracies caused by modelling assumptions related to boundary conditions and loading scenarios (Rafiei and Adeli, 2018; Yuen et al., 2016). Additionally, noise in the data can cause instability in the estimation process, resulting in ambiguity in the decision-making process. The utilisation of modern machine learning (ML) techniques mitigates the complexities leading to enhanced dependability and estimation accuracy across a broad range of applications (Narasimhan and Nagarajaiah, 2018; Sharma and Subhamoy, 2020, 2021). Despite wider application and popularity, the inverse problem techniques have not been well explored in offshore systems, with fewer attempts being made to use the notion of a digital twin to enhance the reliability of future offshore wind platform design (Tygesen et al., 2018).

The stability of offshore structures, including floating vessels and tension leg platforms, largely depends on station-keeping systems like mooring lines. Extensive research has been conducted to optimise support structure and mooring system designs in order to assess the economic feasibility of FOWT systems (Coulling et al., 2013). The dynamics and stability of the system can be significantly impacted by any damage to the mooring line, which is a significant concern. Previous studies have parameterised mooring damage by focusing on key

* Corresponding author.
E-mail address: ptamuly@bcamath.org (P. Tamuly).

https://doi.org/10.1016/j.oceaneng.2025.121199
Received 19 November 2024; Received in revised form 18 March 2025; Accepted 7 April 2025
Available online 19 April 2025

aspects such as fatigue damage, strength capacity, tension, and durability (Santos and Rosas, 2021; Soliman and El-Shanawany, 2015; D'Agostino et al., 2019). An extensive study of these aspects can shed light on the structural integrity and performance of mooring lines undergoing long-term deterioration (Ma et al., 2013; Gao et al., 2010; Nguyen and Soares, 2020). Research on the failure of mooring systems under elevated environmental conditions has highlighted potential risks to these systems (Brindley and Comley, 2014). Several methods, such as fuzzy logic-based damage diagnosis, have been investigated to enhance the reliability and resilience of mooring systems (Jahangiri et al., 2016). Medina et al. employed AI-driven models to predict motions and mooring loads of Spar floating wind turbines, integrating hydrodynamic and aerodynamic factors for enhanced accuracy in wave and wind conditions (Medina-Manuel et al., 2024). In addition, research has underscored the utilisation of simulation techniques for real-time stress monitoring in a variety of mooring configurations, which has the potential to improve predictive maintenance and mitigate hazards (Chen et al., 2023c). Furthermore, critical data for optimising mooring designs and preventing fatigue failure is provided by insights into localised stress responses in mooring chains, which are influenced by their geometrical and material nonlinearity (Chen et al., 2023b). The significance of customising mooring configurations to withstand adverse environmental conditions is emphasised by site-specific research, such as mooring designs for high-capacity FOWTs in the South China Sea (Chen et al., 2023a). One of the recent techniques that gained popularity for detecting mooring line damage is floating platform dynamic response monitoring. This method employs sensors positioned on the floating platform to capture anomalies in its behaviour, which signals underlying damage to the mooring lines. Detecting mooring line damage through floating platform response offers several advantages over conventional in-situ techniques, such as ROV inspections, acoustic sensors, or visual assessments (Al-Jabri et al., 2006; Sidarta et al., 2023; Stambaugh and Inalpolat, 2016). This approach is particularly advantageous for FOWT installed in deepwater conditions where traditional in-situ techniques involve high operation costs and the logistical challenges associated with underwater inspection. Furthermore, it allows continuous monitoring of damage by capturing changes in the dynamic characteristics of the platform. However, the effective application of data-driven damage assessment using platform response requires well-annotated datasets, which are challenging to obtain due to the difficulty of recording real data in deep-sea environments impacted by natural degradation and extreme events (Wang et al., 2001; Sharma et al., 2023).

Nowadays, deep learning (DL) approaches combined with data-driven methods are gaining popularity for monitoring FOWT mooring lines and identifying early damages. Gorostidi et al. (2023) proposed an auto-encoder-based deep neural network for detecting bio-fouling failure and anchorage displacement in mooring systems using displacement and rotation data. A recent study by Sharma and Nava (2024) monitored mooring systems using Convolution Neural Network (CNN) combined with Auto-Regressive (AR) coefficients accounting for the wave randomness and metocean conditions. The novel technique demonstrated the potential of AR coefficients coupled with CNN for detecting and classifying damages in mooring systems of FOWT. The proposed approach is also compared with previously existing ML methods in a supervised way. The accuracy of the classification models using the ML methods completely relies on the quality and quantity of the data.

In real-world applications, training models using collected datasets presents several challenges, largely due to factors such as class imbalances and limited labelled data sets caused by the high costs and complexities of experimental procedures (Chen et al., 2018; Pan et al., 2020). These results in a decline in model performance, manifesting as increased misclassifications due to biased learning towards majority class data (Farrar and Worden, 2012; Zhang et al., 2020). Generally, data is collected through sensors autonomously, without human involvement. It is often challenging to acquire a substantial dataset for supervised damage diagnosis. Labelling collected samples is often time-consuming and costly, necessitating human expertise regarding system states. Consequently, datasets pertaining to actual industrial applications are typically unlabelled. Attempts to label these unlabelled samples do not ensure the accuracy of the labels, as they are influenced by the confirmational data biases of the engineers interpreting the data (Verstraete et al., 2020). Consequently, challenges to mainstream supervised learning approaches will arise from both label scarcity and label accuracy issues. A practical approach to address these challenges is taking advantage of semi-supervised learning algorithms, which can effectively harness both the limited labelled data and the extensive unlabelled data concurrently (Verstraete et al., 2020; Razavi-Far et al., 2018; Liu and Gryllias, 2020). Moreover, limited labelled data increases the risk of overfitting, hindering the ability of the model to generalise (Sun et al., 2018; Yao et al., 2021). Hence, the majority of the successes of DL-based damage detection methods have been achieved through the framework of supervised learning with substantial balanced labelled data. The amount of labelled training data required to achieve satisfactory results for a specific problem depends on the accuracy of the training data labels and the degree of separation of features among different categories (LeCun et al., 2015; Schmidhuber, 2015). This poses a major challenge for practical industrial applications since labelling collected vibration signals requires domain expertise and engineering experience.

Semi-supervised learning is particularly suited to scenarios where only a small set of the dataset is labelled, addressing the classification problem in such contexts. This approach involves two steps: first, a feature extractor is pre-trained using unlabelled data in an unsupervised manner. Next, a classifier is trained in a supervised manner using features obtained from labelled data (Lei et al., 2016). Presently, only a few paradigms of semi-supervised learning have been utilised in damage diagnosis. The support vector data description method in Liu and Gryllias (2020) employs cyclic spectral coherent domain indicators to create a feature space and fit a hypersphere, subsequently calculating the Euclidean distance to differentiate between faulty and healthy data. Studies by Chen et al. (2018) and Zhao et al. (2017) use graph-based methods to make networks that connect similar samples in the dataset.

In addition, Generative Adversarial Networks (GANs) offer an alternative approach for limited data situations by generating synthetic data, thereby increasing data diversity (Engelmann and Lessmann, 2021). Gao et al. (2019) proposed a deep convolutional GAN (DCGAN)-based leaf-bootstrapping method for civil engineering problems. Semi-supervised GANs showed promising performance in concrete crack detection by combining data generation with model training (Gao et al., 2021). A similar study by Kingma and Welling (2022) proposed a semi-supervised Variational Autoencoder, to identify structural damage. Despite these advancements, most of these applications focus on vision-based datasets, with little exploration of damage detection using time-series data, especially for mooring line damage detection.

Moreover, there is a continuous demand for effective damage detection techniques that can operate efficiently with limited labelled data. Developing robust models that accurately identify mooring line damage is crucial for improving the safety and reliability of marine operations. In addition to identifying damage states, artificial data is needed to represent a range of damage scenarios. Such data can introduce underrepresented damage situations, thereby enhancing data diversity and improving the ability of machine learning models to detect and classify damage states. Furthermore, artificial data enables the rapid development and testing of algorithms enhancing R& D without the time-consuming process of collecting real-world samples. In general, the design of mooring systems for FOWT often necessitates extensive testing. To fulfill these requirements artificial data is generated by substantial volume of simulated testing scenarios that cover a range of damage conditions. With these in mind, the main contributions of the paper are twofold:

- The study introduces a Time-Series Variational Semi-Supervised Learning (TSVSSL) framework that combines the advantages of Variational Autoencoders (VAEs) with a semi-supervised learning

approach specifically designed for time-series data. This algorithm features a distinctive training procedure where the encoder-decoder and classifier components of the semi-supervised VAE framework are trained concurrently. This simultaneous training enables the framework to perform exceptionally well in both damage detection and data generation.
- The paper introduces a "rejection sampling" method that generates damage-state-specific samples using a trained decoder in combination with a classifier. This ensures the generated synthetic time-series data meet specific class criteria through an iterative acceptance process. Unlike Conditional VAEs (Harvey et al., 2021), which rely on large amounts of labelled training data, the proposed technique operates effectively with limited labelled data, making it particularly advantageous.

The structure of the remainder of the paper is as follows: Section 2 presents the methodology, detailing the development of the proposed TSVSSL framework. Section 3 discusses the numerical experiments, including the dataset, architecture, and training procedure. Section 4 outlines the investigation framework, with the corresponding results and discussion in Section 5. Artificial data generation is addressed in Section 6, and finally, Section 7 provides the conclusion of the paper.

2. Methodology

The section is divided into three subsections, each describing critical aspects of the proposed approach. The first subsection introduces the Variational Autoencoder, differentiating from traditional autoencoders by using variational inference to enable both data reconstruction and generation. The second subsection focuses on the semi-supervised VAE that integrates both labelled and unlabelled data to improve classification performance while retaining generative capabilities. Finally, the TSVSSL framework, which includes the shared encoder, decoder, and classifier networks, is proposed. This section further explains the loss functions corresponding to labelled and unlabelled data, facilitating the learning of meaningful latent representations for classification.

2.1. Variational autoencoder

Traditional autoencoders are neural networks that learn to compress data into a lower-dimensional latent space (encoding) and then reconstruct it back to the original space (decoding). However, these models primarily emphasize accurately reconstructing the input data rather than ensuring the formation of a well-structured latent space. This can result in a fragmented latent representation, making it difficult to leverage for clustering or interpolation. Additionally, traditional encoders are not able to generate new artificial data, as they are not designed to sample from the latent space. To overcome these limitations, Kingma (2013) introduced the concept of the VAE incorporating principles of variational inference into the autoencoder framework. Unlike traditional autoencoders, where the encoder maps the input to a deterministic latent space, the VAE encode the data into a Gaussian probability distribution, characterized by the mean and variance. Latent variables sampled from this distribution are used by the decoder network for reconstruction. Let us assume VAE generates data x from underlying latent variables z through a generative process modelled by a conditional probability distribution $p_\theta(x \mid z)$, where θ represents the parameters of the model. The primary objective is to learn the distribution $p_\theta(x)$ by maximizing the likelihood of the observed data. This likelihood can be expressed as:

$$p_\theta(x) = \int p_\theta(x \mid z) p(z)\, dz, \tag{1}$$

where $p(z)$ is the prior distribution over the latent variables. However, directly optimizing this marginal likelihood is computationally intractable due to the integral over the latent variable z. To address this challenge, VAEs employ Variational Inference by introducing a variational distribution $q_\phi(z \mid x)$ to approximate the true posterior distribution $p_\theta(z \mid x)$ with ϕ being the parameters of the variational distribution. Instead of optimizing the intractable marginal likelihood directly, VAEs maximize the *Evidence Lower Bound* (ELBO) (Damm et al., 2023), which is given by:

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x \mid z)\right] - \mathrm{KL}(q_\phi(z \mid x) \parallel p(z)), \tag{2}$$

where $\mathrm{KL}(q_\phi(z \mid x) \parallel p(z))$ is the Kullback–Leibler (KL) divergence between the variational posterior $q_\phi(z \mid x)$ and the prior $p(z)$. The prior $p(z)$ is typically chosen to be a standard normal distribution $\mathcal{N}(0, I)$, where I is the identity matrix. The KL divergence term acts as a regulariser, encouraging the variational distribution to be close to the prior distribution. The expectation operator ensures that the latent variables z are responsible for reconstructing the data x. By optimizing the ELBO with respect to the parameters ϕ and θ, VAEs effectively learn a probabilistic mapping between the data and the latent space. Apart from their well-known capability in generating new samples, VAEs have numerous applications in supervised learning tasks (Ma et al., 2020). One of the key strengths of VAEs is to leverage latent distribution for supervised models. One such application is to use VAEs as a semi-supervised way to learn robust representations from both labelled and unlabelled data, which is further explained in the following subsection.

2.2. Semi-supervised variational autoencoder

The Semi-Supervised Variational Autoencoder (SSVAE) extends the traditional VAE framework by incorporating both labelled and unlabelled data during training (Kingma and Welling, 2014). In standard VAEs, the model focuses on learning the underlying distribution of the data through the conditional probability distribution $p_\theta(x \mid z)$ to generate data x from latent variables z. However, this method can limit its ability to leverage available label information for enhancing classification performance. SSVAE addresses this limitation by incorporating a supervised component that utilizes labelled data into the learning process. Specifically, SSVAE modifies the traditional ELBO to incorporate a supervised loss term, allowing the model not only to reconstruct the data but also to predict class labels for the input data. This is accomplished through a classifier that operates on the latent space to maximize the likelihood of the correct labels while simultaneously maintaining the generative capabilities of the VAE. The updated ELBO for the SSVAE can be expressed as follows:

$$\mathcal{L}(\theta, \phi) = \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x \mid z)\right] - \mathrm{KL}(q_\phi(z \mid x) \parallel p(z)) - \lambda \mathcal{L}_{\text{supervised}}, \tag{3}$$

where $\mathcal{L}_{\text{supervised}}$ is the supervised loss term that leverages the labelled data, and λ is a weighting factor that balances the contribution of the supervised loss. This approach not only improves latent representations but also enhances their performance on classification tasks. Thus, the SSVAE effectively bridges the gap between generative modelling and supervised learning, making it a powerful tool for classification tasks with limited labelled data.

2.3. Time-series variational semi-supervised learning

The proposed framework integrates the semi-supervised variational learning principles specifically tailored for time-series damage detection. The proposed framework comprises three neural networks: the Shared Encoder, the Decoder, and the Classifier as shown in Fig. 2. The shared encoder transforms the input time-series data into a latent distribution, preserving essential data features. The latent distribution is defined with two key components: the mean (Z_{mean}) and log variance (Z_{var}) from which random samples are drawn. The decoder reconstructs the original time-series input from the latent samples, capturing the underlying spatial-temporal patterns in the data. Finally, the classifier utilises the latent space information to predict class labels for the input time-series data. The proposed SSVAE utilises two distinct types

of loss functions: supervised loss and unsupervised loss for labelled and unlabelled data, respectively.

2.3.1. Unsupervised loss

The unsupervised loss is grounded in the variational inference framework, which is essential for training VAEs with unlabeled datasets. Given an input x, the encoder generates the parameters of the latent distribution z. This latent variable is sampled using the reparameterization trick, enabling gradient back-propagation through the stochastic gradient techniques:

$$z = \mu + \sigma \odot \epsilon, \tag{4}$$

where μ and σ are the mean and standard deviation outputs from the encoder, respectively, and ϵ is a noise sample drawn from a standard normal distribution $\mathcal{N}(0, 1)$. The generated sample z is further used by the decoder to reconstruct the input x'. The reconstruction loss demonstrates the ability of the decoder to reconstruct the original input from the latent representation. This loss is formulated as the Mean Squared Error (MSE) between the original input x and the reconstructed output x':

$$\mathcal{L}_{\text{recon}}(x, x') = \frac{1}{N} \sum_{i=1}^{N} \left(x_i - x_i'\right)^2, \tag{5}$$

where N represents the number of elements in the input x. This loss function encourages the model to minimize the squared differences between the original input and the reconstructed output, thus improving the reconstruction quality of the decoder. In addition to the reconstruction loss, the Kullback–Leibler (KL) divergence is incorporated as a regularization term for the latent distribution. The KL divergence encourages the learned distribution $q_\phi(z|x)$ to approximate a prior distribution $p(z)$, which is typically chosen to be a standard normal distribution.

$$\mathcal{L}_{\text{KL}} = \text{KL}(q_\phi(z|x)||p(z)). \tag{6}$$

Moreover, a marginalisation term is included to account for the predictive class probabilities derived from the latent representation. It serves as a regularizer, encouraging the model to learn meaningful latent representations that support classification by penalizing uncertain predictions. Overall, incorporating this term leads to a more structured latent space, improving both the generative and discriminative capabilities of the model. The marginalisation term can be expressed as:

$$\mathcal{L}_{\text{marg}} = -\sum_{i=1}^{C} p(y_i|z) \log p(y_i|z), \tag{7}$$

where $p(y_i \mid z)$ represents the probability of the i-th class predicted by the classifier based on the latent representation z, and C is the total number of classes. The total unsupervised loss for a single input can thus be expressed as:

$$\mathcal{L}_{\text{unsupervised}} = \mathcal{L}_{\text{recon}} + \beta \mathcal{L}_{\text{KL}} + \gamma \mathcal{L}_{\text{marg}}, \tag{8}$$

The hyperparameters β and γ control the balance between different terms in the unsupervised loss function. β adjusts the weight of the KL divergence term, influencing latent space to approximate prior distribution which is taken as a standard normal distribution, while γ controls the marginalization term, which encourages meaningful latent representations that support classification. Both hyperparameters are fixed within the range of 0 to 1, and their values determine the trade-off between reconstruction accuracy, latent space regularization, and classification performance during training.

2.3.2. Supervised loss

The supervised loss term is aimed at optimising the classification task using the labelled data. The classifier operates on the latent representation z and produces class probabilities for the corresponding input:

$$y_{\text{pred}} = q_\phi(y|z). \tag{9}$$

The supervised loss is formulated as the categorical cross-entropy between the true class labels and the predicted probabilities, which measures the performance of the classifier on the labelled dataset:

$$\mathcal{L}_{\text{supervised}}(y, y_{\text{pred}}) = -\sum_{i=1}^{N_s} \sum_{j=1}^{C} y_{ij} \log(q_\phi(y_j|z_i)) \tag{10}$$

where N_s represents the number of labelled samples in the dataset, C is the number of classes, and y_{ij} is the true label for the i^{th} sample corresponding to class j.

The effectiveness of the proposed framework is derived from a simultaneous training procedure that is conceptually based on the M2 model outlined in the study by Kingma and Welling (2014). The training procedure tailored for time-series data is provided in Algorithm 1. The training starts with the unsupervised part, where batches of unlabelled time-series data are fed into the shared encoder to create latent representations. These representations are then used by the decoder for reconstruction. The quality of the compression reconstructions is measured with unsupervised loss in Eq. (8) and weights are updated accordingly (θ_E and θ_D). At the same time, the supervised training phase processes batches of labelled data through the encoder and classifier loop. The predictions of the classifier are compared with the true labels using a classification loss in Eq. (10), which updates the shared encoder and classifier weights (θ_E and θ_C). The training process alternates between updating the weights of the shared encoder and decoder for the unsupervised loss, and the encoder and classifier for the supervised loss, allowing the weights to adapt effectively to both the generation and classification tasks.

Algorithm 1 Proposed TSVSSL algorithm.

Initialization:
Define the shared Encoder model E with encoder weights θ_E
Define the Decoder model D with decoder weights θ_D
Define the Classifier model C with classifier weights θ_C
Define the Adam optimisers: optimizer_vae for the E and D, and optimizer_classifier for E and C.
Training Loop:
for each epoch e from 1 to epochs **do**
  for each batch b from 1 to num_batches **do**
    Unsupervised Learning:
    Sample a batch of unsupervised data: x_{unsup}
    Compute the unsupervised loss $\mathcal{L}_{\text{unsupervised}}$ using E and D
    Update the Encoder and Decoder weights: $\theta_E, \theta_D \leftarrow \theta_E, \theta_D -$ optimizer_vae $\cdot \nabla_{\theta_E, \theta_D} \mathcal{L}_{\text{unsupervised}}$
    Supervised Learning:
    Sample a batch of supervised data: $x_{\text{sup}}, y_{\text{sup}}$
    Compute the supervised loss $\mathcal{L}_{\text{supervised}}$ using E and C
    Update the Encoder and Classifier weights: $\theta_E, \theta_C \leftarrow \theta_E, \theta_C -$ optimizer_classifier $\cdot \nabla_{\theta_E, \theta_C} \mathcal{L}_{\text{supervised}}$
  end for
end for

3. Numerical experimentation

Operational data on offshore wind turbines under varying environmental conditions is often scarce and rarely accessible in the public domain. Due to constraints like maintaining competitive advantage, regional regulations, and the need for data confidentiality, it is often difficult to access real-world data. In the absence of actual data collected from real FOWTs, the open-source simulation tool OpenFAST by the National Renewable Energy Laboratory (NREL), is employed to synthetically generate data representing both the damaged and healthy states. The simulated data of the semi-submersible floating system that is developed as part of the DeepCWind project (Jonkman et al., 2009) as shown in Fig. 1. The DeepCWind design includes a central column, three

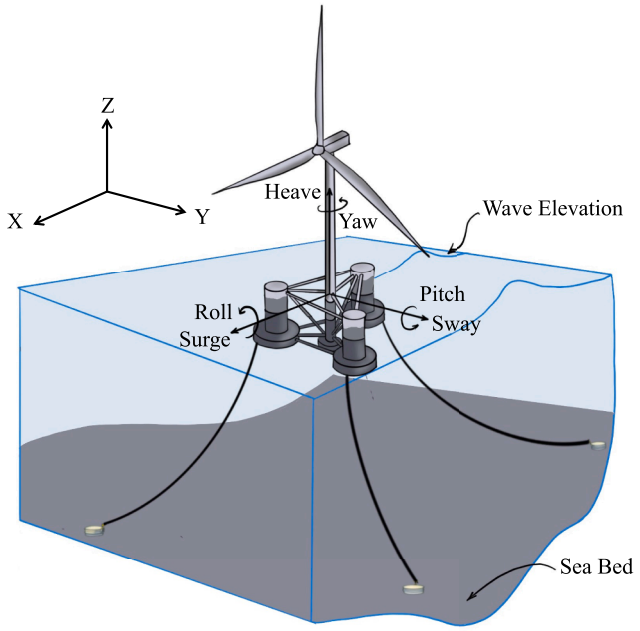


Fig. 1. A schematic representation of FOWT with semi-submersible DeepCWind platform.

peripheral offset columns, and an array of slender braces to enhance the structural integrity of the floating structure. Additionally, a set of three multi-point catenary chain moorings is symmetrically connected to the platform and anchored to the seabed to prevent displacement beyond the design limit. The primary focus of this study is to assess the damage to these mooring systems using simulated time series displacement responses of the platform's degrees of freedom (translational and rotational) under various environmental conditions. The simulation of the platform response is carried out using NREL's (OpenFAST) open-sourced wind turbine simulation tool, OpenFAST. It is highly regarded in this field as one of the key simulation environments for modelling the complex behaviour of semi-submersible FOWT. The modular architecture of this software facilitates the integration of various physics modules, including aerodynamics (AeroDyn), hydrodynamics (HydroDyn), and mooring dynamics (MoorDyn), to capture the complex interactions among different components of the wind turbine system, such as the rotor, tower, floating platform, and mooring systems.

To accurately reflect the different degradation modes of the mooring system under real-world conditions, the response of the floating platform is simulated over a broad spectrum of environmental conditions. In general, environmental conditions are characterised by wave height, wind speed, wave period, and current speed. The response of the platform for both damaged and healthy scenarios is generated by randomly selecting environmental conditions from the range specified in Table 1. These ranges of environmental conditions are not specific to any particular site; instead, they cover the full operational range found at a generic location.

Table 1
Simulation setup and environmental conditions.

Property	Value	Units
Simulation time	2000	s
Sampling time step	0.20	s
Transient time	500	s
Significant height, H_S	1–10	m
Peak period, T_P	7–15	s
Wind Velocity, V	1–15	m/s
Current Speed, C	0.5–1.5	m/s

The research focuses on two primary types of damage commonly observed in mooring lines of offshore structures: biofouling and anchor sliding. Biofouling is the accumulation of seaweeds, microorganisms, plants, algae, and other sea organisms, such as barnacles and mussels, on the mooring line, making it heavy and changing its structural characteristics. Similarly, anchor sliding problems arise when anchors lose their holding capacity due to factors such as seabed conditions, extreme weather events, or insufficient embedment. Thankfully, the modular nature of OpenFAST allows for the implementation of custom modifications in the MoorDyn module to simulate these damage scenarios in the mooring system. To simulate biofouling damage, the mass per unit length of the mooring lines are enhanced without altering the stiffness properties Sharma et al. (2023). While actual biofouling can lead to a wide range of mass increments, from minor (5 %) to severe (> 50 %) Decurey et al. (2020), the authors have chosen a minimal range (5–10 %) to effectively demonstrate the algorithm's capability in detecting subtle biofouling effects. Identifying damage at this early stage is vital for maintenance purposes, enabling timely intervention to prevent further damage. The damage percentages are uniformly distributed across the dataset, with each sample corresponding to a fixed damage level. Moreover, biofouling damage is applied uniformly across all mooring lines, as they are subjected to the same underwater conditions. A similar approach is adopted for anchorage failure, where the anchor point of a single mooring line along the wave direction is shifted within a range of ±5–10 m.

Typically, each simulated event contains six time-history responses corresponding to six degrees of freedom (*dofs*) of the platform: surge, sway, heave, roll, pitch, and yaw. In structural analysis, understanding the sensitivity of *dofs* to structural anomalies is crucial. A recent study conducted by Sharma and Nava (2024) on the same setup demonstrated that the sensitivity of mooring damage in sway, roll, and yaw *dofs* is minimal. Based on these observations, this study focuses on three primary *dofs* namely surge, heave, and pitch, which demonstrate higher sensitivity to mooring system damage due to the structural symmetry of the system and the direction of excitation. Thus, in each metocean scenario, single-point response measurements contain three *dofs* for 2000 s duration at 5 Hz sampling frequency. From this sample, the last 1600 time steps from each of the three *dofs* are selected, after excluding the first 500 s corresponding to the unstable phase. This ensures that any remaining effects from the unstable phase are removed, allowing the analysis to focus on the stable portion of the simulation. To preprocess the data, the mean of the respective sample is subtracted to centre the data around zero, and then normalisation is applied by scaling the data with respect to maximum absolute value within that channel. The training dataset consists of 12,000 samples representing different system states under a range of environmental conditions given in Table 1. Importantly, the set of environmental conditions remains constant, with 4000 unique combinations applied across all health states viz., healthy (i.e., undamaged), biofouling, and anchor slippage. This approach is designed to isolate the effects of damage on the response rather than external metocean factors. For testing, a separate dataset is created with 300 samples, including 100 from each health state. Table 2 provides a detailed breakdown of both the training and testing datasets, including subsets of labelled data corresponding to 5 %, 10 %, 20 %, and 30 % of the total samples. Following the earlier-detailed method in this manuscript, a TSVSSL approach is used to distinguish the features from the data. Since the data includes multivariate time-series responses of the platform, special attention has been paid to the architecture of neural network components to capture the inherent damage characteristics, as illustrated in the following subsection.

3.1. Architecture

The architectural designs used here are specifically designed to handle the temporal dependencies inherent in time-series data, offering enhanced interpretability and reduced training times. In this proposed

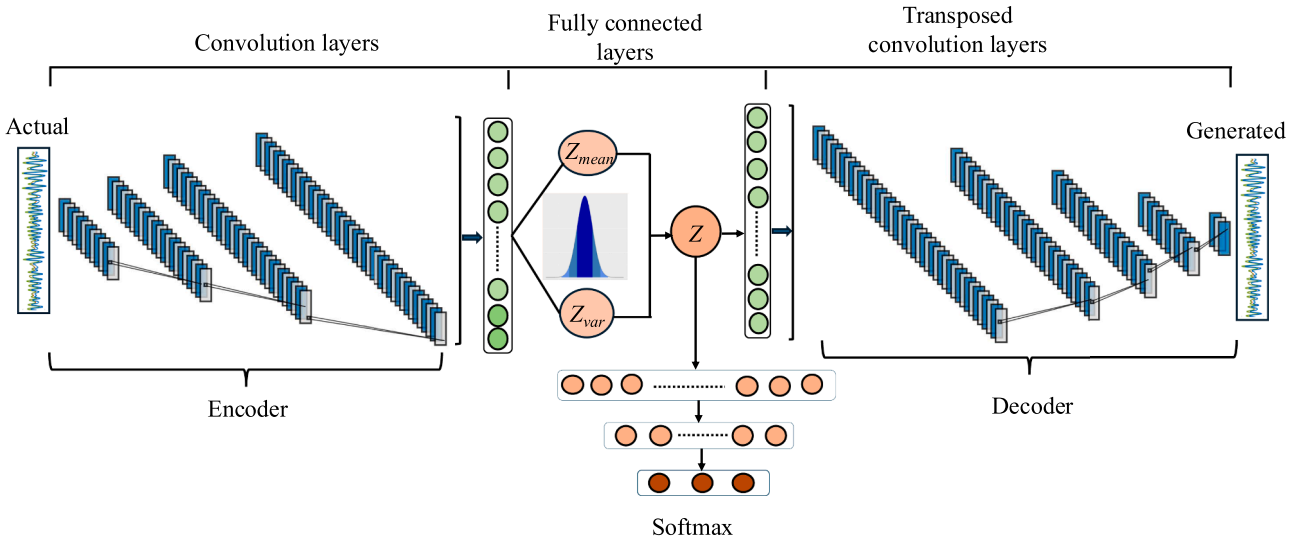


Fig. 2. Schematic representation of the proposed TSVSSL framework.

Table 2
Training and testing datasets.

Dataset	State	Total samples	% Labelled data			
			5%	10%	20%	30%
Training (12,000)	Biolfouling	4000	200	400	800	1200
	Anchor Slippage	4000	200	400	800	1200
	healthy	4000	200	400	800	1200
Testing (300)	Biolfouling	100	–	–	–	–
	Anchor Slippage	100	–	–	–	–
	healthy	100	–	–	–	–

study, the encoder-decoder architecture uses convolution operations for identifying underlying features along with multi-layer perceptron (MLP) for classification. An overview of the proposed architecture is presented in pictorial form in Fig. 2.

The architecture details of the shared encoder are presented in Table 3. The encoder takes an input tensor of three-channel time-series data corresponding to three degrees of freedom as outlined in the previous subsection. It is followed by several convolution layers (Conv1D) with a ReLU activation function that progressively reduces the spatial dimensions while increasing the depth of the feature maps. Each convolution layer is followed by a dropout layer to prevent over-fitting during the classification task. After the convolution layers, the feature maps are flattened into a one-dimensional vector, which is then processed by two dense layers to estimate the mean (*z_mean*) and log variance (*z_var*) of the latent space distribution. Finally, a lambda layer samples the latent vector z from this distribution using the reparameterisation technique.

The architecture of the decoder is presented in Table 4, which mirrors the structure of the encoder but operates in reverse order. It begins with a sample from latent distribution, which is processed through several transposed convolution layers (Conv1DTranspose) that progressively up-sample the data, restoring the original spatial dimensions while decreasing the depth of the feature maps. Unlike the encoder, no dropout is applied in the decoder because primary role of the decoder is reconstruction rather than classification. The final layer of the decoder employs the *tanh* activation function, reconstructing the original input shape in a normalised form.

The architecture of the classifier is outlined in Table 5, where the input to the model is the sample from latent space distribution generated by the shared encoder. This input passes through two dense layers utilising the ReLU activation function to introduce non-linearity. Each dense layer is followed by a dropout layer to reduce overfitting during training. Finally, a softmax-activated output layer produces the probability distribution for multi-class damage assessment tasks.

4. Investigation framework

To provide a comprehensive evaluation of the TSVSSL algorithm, its performance is benchmarked against several prominent deep-learning models. This comparison is conducted to analyze the relative strengths and limitations of TSVSSL across diverse scenarios, ensuring a robust

Table 3
Architecture details of the shared encoder.

Layer	Name	Kernel (Filter)	Activation	Shape	Remarks
Input	–	–	–	$(N, 1600, 3)$	3-channel timeseries
Conv1D	Conv1	3 (32)	ReLU	$(N, 800, 32)$	Stride = (2), Padding = 'same'
Dropout	Drop1	–	–	$(N, 800, 32)$	Dropout rate = 0.25
Conv1D	Conv2	3 (64)	ReLU	$(N, 400, 64)$	Stride = (2), Padding = 'same'
Dropout	Drop2	–	–	$(N, 400, 64)$	Dropout rate = 0.25
Conv1D	Conv3	3 (128)	ReLU	$(N, 200, 128)$	Stride = (2), Padding = 'same'
Dropout	Drop3	–	–	$(N, 200, 128)$	Dropout rate = 0.25
Conv1D	Conv4	3 (256)	ReLU	$(N, 100, 256)$	Stride = (2), Padding = 'same'
Dropout	Drop4	–	–	$(N, 100, 256)$	Dropout rate = 0.25
Flatten	Flatten_1	–	–	$(N, 25600)$	Flattening to 1D
Dense	z_mean	–	ReLU	$(N, 128)$	Latent mean
Dense	z_var	–	ReLU	$(N, 128)$	Latent log variance
Lambda	z	–	–	$(N, 128)$	Sampling from latent space

Table 4
Architecture details of the decoder.

Layer	Name	Kernel (Filter)	Activation	Shape	Remarks
Input	–	–	–	$(N, 128)$	Latent space input
Dense	Dense_1	–	ReLU	$(N, 3200)$	3200 neurons
Reshape	Reshape_1	–	–	$(N, 100, 32)$	Reshape to 3D tensor
Conv1DTranspose	Deconv1	3 (256)	ReLU	$(N, 100, 256)$	Stride = (2), Padding = 'same'
Conv1DTranspose	Deconv2	3 (128)	ReLU	$(N, 200, 128)$	Stride = (2), Padding = 'same'
Conv1DTranspose	Deconv3	3 (64)	ReLU	$(N, 400, 64)$	Stride = (2), Padding = 'same'
Conv1DTranspose	Deconv4	3 (32)	ReLU	$(N, 800, 32)$	Stride = (2), Padding = 'same'
Conv1DTranspose	Deconv5	3 (3)	Tanh	$(N, 1600, 3)$	Stride = (2), Padding = 'same'

Table 5
Architecture details of the classifier.

Layer	Name	Activation	Shape	Remarks
Input	–	$(N, 128)$	Latent sample from encoder	
Dense	Dense_1	ReLU	$(N, 128)$	128 neurons
Dropout	Drop1	–	$(N, 128)$	Dropout rate = 0.25
Dense	Dense_2	ReLU	$(N, 64)$	64 neurons
Dropout	Drop2	–	$(N, 64)$	Dropout rate = 0.25
Dense	Output_Layer	Softmax	$(N, 3)$	-

assessment of its overall effectiveness. The models chosen for benchmarking include:

- **Deep Convolutional Neural Network (DCNN):** To rigorously evaluate the proposed TSVSSL algorithm, the DCNN classifier is used as a primary benchmark due to its extensive application and proven effectiveness in classification tasks. Notably, the DCNN classifier is designed with the same architecture as the discriminator model in the TSVSSL framework, as outlined in Table 3. One limitation of this framework is that it relies solely on labelled data, which restricts its ability to leverage unlabelled data.
- **Autoencoder (AE):** Autoencoder architecture is chosen due to its capability in unsupervised learning and reconstruction tasks. To address the challenge of limited labelled data for the classification task, the framework first trains a simple encoder-decoder model using the available unlabelled data. Once this unsupervised training is complete, the learned latent features are extracted and used for damage classification. Next, the classifier is trained in a supervised manner using the labelled samples, leveraging the latent representations learned from the larger pool of unlabeled data. The AE has the same architecture as the encoder-decoder model used in the TSVSSL framework, with the key distinction that, unlike the VAE, which models a latent distribution, the AE employs a deterministic latent vector. Additionally, the classifier architecture mirrors that of the proposed TSVSSL framework to ensure consistency.
- **Variational Autoencoder (VAE):** The Variational Autoencoder is implemented in a semi-supervised setting within a transfer learning framework, drawing on the principles of the M1 model presented in the study by Kingma and Welling (2014). The VAE follows a two-step training process. Initially, the VAE is trained on the available unlabelled data using an encoder-decoder architecture. Once this unsupervised training phase is complete, the learned latent distribution is utilised for damage classification. Subsequently, the classifier is trained in a supervised manner using the labelled samples, allowing the model to effectively leverage the latent representations obtained from the unlabelled data. The architecture is identical to that of the proposed TSVSSL algorithm, with the only difference being in the training procedure. In the TSVSSL framework, the encoder-decoder and classifier are trained simultaneously, whereas in the VAE approach, the training occurs in two distinct phases.

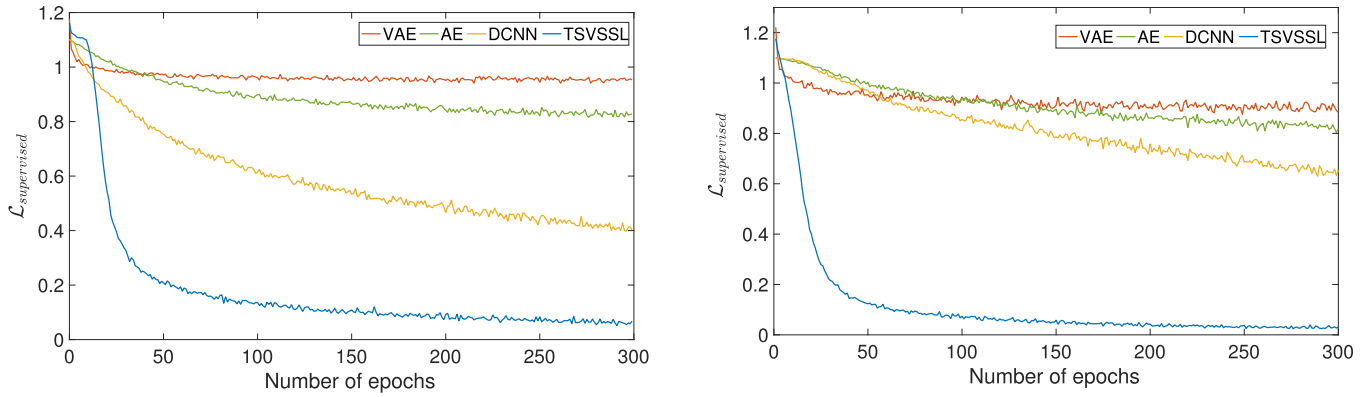
The selection of these three algorithms for comparison with the proposed TSVSSL algorithm is grounded in the principle of architectural consistency. Both the AE and VAE share similar architectural frameworks with the TSVSSL model, particularly in their encoder-decoder configurations, facilitating a direct and meaningful comparison using only the labelled data. The DCNN is designed with the same architecture as the encoder model, providing a common basis for evaluating damage classification performance. This architectural alignment ensures that differences in performance can be attributed to the respective training methodologies rather than variations in the structure of the model. By maintaining this consistency, the comparative analysis can effectively isolate the impact of the proposed TSVSSL training procedure, allowing for a clearer understanding of its advantages and capabilities in different learning contexts.

To assess the performance of the proposed TSVSSL algorithm, experiments are carried out across four distinct scenarios involving 5 %, 10 %, 20 %, and 30 % labelled data. This evaluation aims to shed light on the effectiveness of the TSVSSL algorithm with varying levels of labelled information. For each scenario, the models are trained for a total of 300 epochs and progress is monitored throughout the training process. The Adam optimizer is utilised, starting with an initial learning rate of 2×10^{-4}. All experiments are conducted on the TensorFlow platform using the Donostia International Physics Center's supercomputing facility, which features NVIDIA Tesla P40 GPUs. This robust computational environment enables the efficient management of the substantial training demands associated with the experiments.

5. Results and discussions

A comparative analysis is performed systematically to assess the performance of the proposed TSVSSL algorithm involving several prominent deep learning models, as detailed in the preceding section. The evaluation utilised a variety of performance metrics, including accuracy, precision, recall, and F1 score, to quantify the effectiveness of each model in detecting and classifying damage across various classes in the test dataset.

Fig. 3 shows the supervised loss for all four models using 30 % labelled and 10 % labelled data, which reflects the performance of the models in the classification task. TSVSSL model exhibits a rapid decrease in loss, stabilising after approximately 50 epochs in both cases. The DCNN model also shows a significant decrease in loss; however, it plateaus at a higher value than the lowest loss achieved by the TSVSSL model. In contrast, both the AE and VAE models initially decrease in loss but stabilise at much higher values. TSVSSL model outperforms the other models in minimising loss, with the DCNN following behind. Similarly, the performance metrics for the AE and VAE models demonstrate significantly poorer performance, as summarized in Table 6. The performance metrics for various models are compared with different percentages of labelled data. Across all levels of labelled data (5 %, 10 %, 20 %, and 30 %), TSVSSL consistently outperforms the other models, achieving the highest accuracy, precision, recall, and F_1 scores. With 5 % labelled data, TSVSSL achieves an accuracy of 54.7 %, significantly outperforming DCNN (49.7 %), AE (46.0 %), and VAE (38.7 %). As the amount of labelled data increases, the performance gap widens even more, with TSVSSL reaching 92.0 % accuracy using 30 % labelled data, while the next best model, DCNN, achieves only 77.7 %. Specifically, the performance of VAE is significantly lower than the other models, with



(a) With 30% labelled Data (b) With 10% labelled Data

Fig. 3. Comparison of absolute loss curve fir different approaches. (a) With 30 % labelled Data. (b) With 10 % labelled Data.

Table 6
Performance metrics for various models with different percentages of labelled data.

% Labelled data	Model	Accuracy	Precision	Recall	F_1
5 %	AE	46.0	46.0	46.0	45.8
	DCNN	49.7	51.1	49.7	49.1
	VAE	38.7	26.2	38.7	30.6
	TSVSSL	54.7	56.5	54.7	55.0
10 %	AE	52.7	52.7	52.7	52.7
	DCNN	65.3	64.6	65.3	64.8
	VAE	34.0	40.8	34.0	29.4
	TSVSSL	74.7	76.8	74.7	74.9
20 %	AE	61.0	61.0	61.0	60.9
	DCNN	77.0	76.9	77.0	76.9
	VAE	41.3	55.3	41.3	38.2
	TSVSSL	81.7	81.6	81.7	81.6
30 %	AE	62.7	62.1	62.7	62.1
	DCNN	77.7	78.3	77.7	77.7
	VAE	43.7	59.0	43.7	39.0
	TSVSSL	92.0	92.1	92.0	92.0

minimal improvement even as the amount of labelled data increases. Notably, the performance metrics followed the same trend depicted by the supervised loss, further emphasizing the limitations of the VAE and AE models.

Furthermore, the confusion matrix is used to get an insight into the performance of these models across different classes. Fig. 4 presents a comparison of all four models on the testing data, trained with 30 % labelled data. The results show that the TSVSSL algorithm is efficient at learning damage patterns in the data, providing more reliable predictions. However, all models performed well in detecting anchor sliding, suggesting distinct and unambiguous characteristics that are relatively simple for the models to recognise. In contrast, identifying biofouling damage is challenging and often leads to misclassification. However, the proposed TSVSSL model excels in this task compared to the other models with fewer misclassification instances.

Although it is evident from the previous results that TSVSSL outperforms other models, the structure of the latent space remains obscure. To address this, t-distributed stochastic neighbour embedding (t-SNE) (Van der Maaten and Hinton, 2008) is utilised as it is a powerful dimensionality reduction technique that effectively visualises high-dimensional data by projecting it into a lower-dimensional space, facilitating a comprehensive understanding of the characteristics of latent space formed by the models. Fig. 5 shows the reduced 2D latent space representation generated by all four models trained with 30 % labelled data where t-SNE highlights the clustering of high-dimensional features within the models' latent space. The AE and VAE models show significant overlap between classes with poor class separation, which translates into poor performance performance in distinguishing between

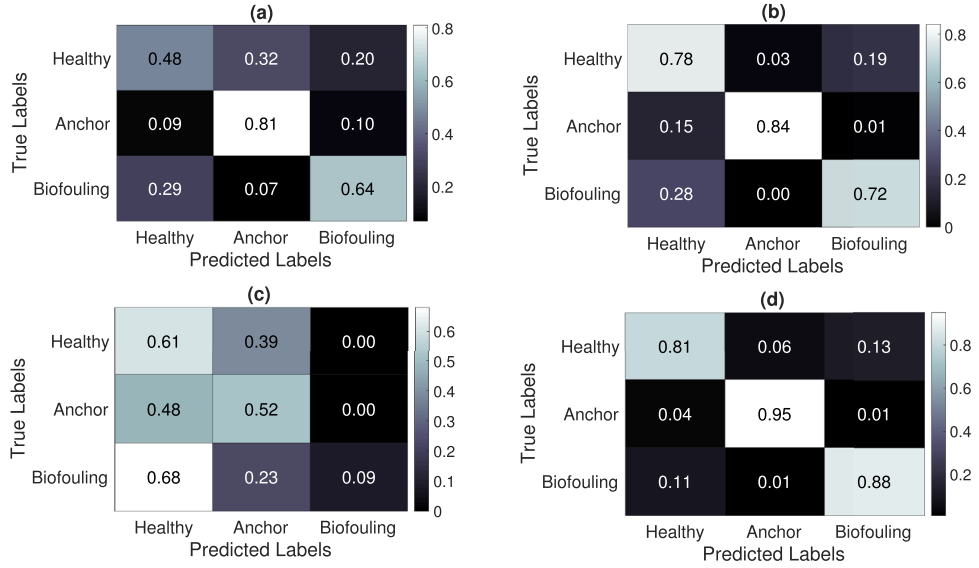


Fig. 4. Normalised confusion matrices for the different models for 30 % labelled data (a) AE (b) DCNN (c) VAE and (d) TSVSSL.

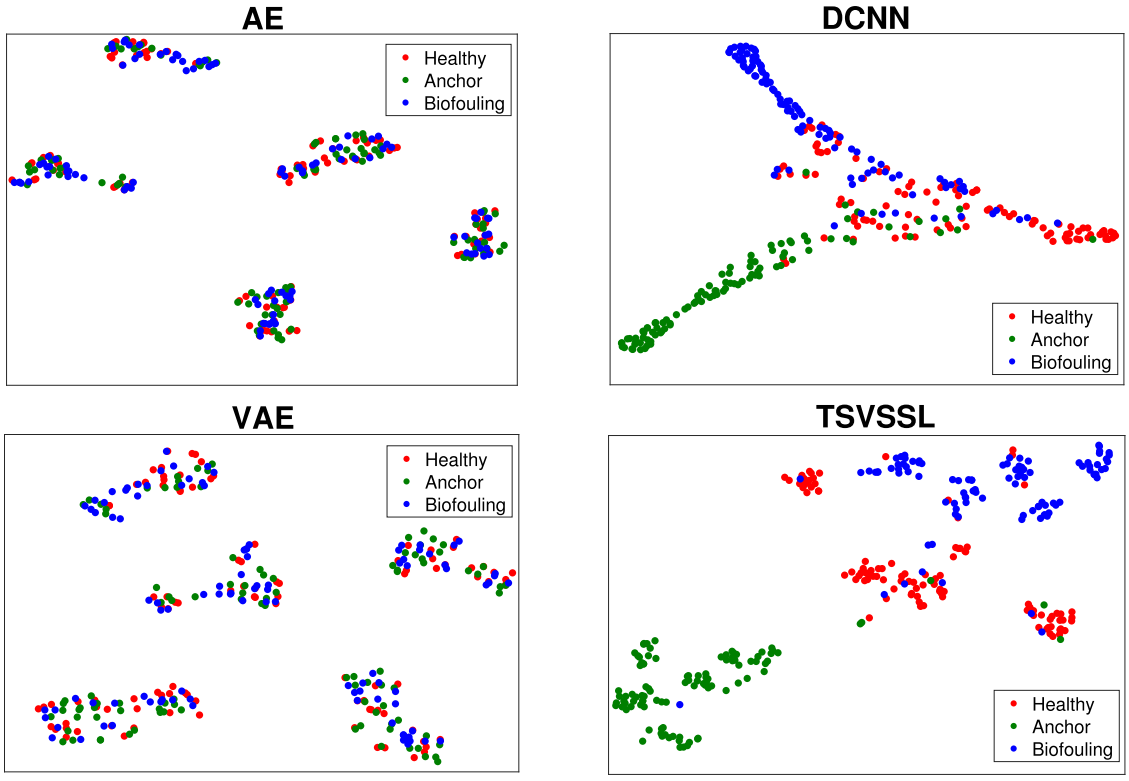


Fig. 5. t-SNE plot for the testing data using the models trained on 30 % labelled data.

damage types. The DCNN shows improved separation, particularly between the anchor sliding and healthy state. However, it experiences some mixing with the biofouling state, which limits its overall effectiveness. In contrast, the TSVSSL model shows clear and distinct clusters for each class, with minimal overlap, leading to better latent feature extraction and, consequently, superior classification accuracy. A deeper understanding of such behaviour of latent space can be analysed through activation magnitude analysis. For this analysis, the common layer "Conv4" (see Table 3) present in all models is used as a representative example. Fig. 6 displays the distribution of neuron activations in this layer across four models for a sample obtained from testing data. The x-axis represents the activation strength, while the y-axis shows the frequency of each activation value. TSVSSL shows a broad and symmetric activation distribution capturing a wide range of features. In contrast, the VAE

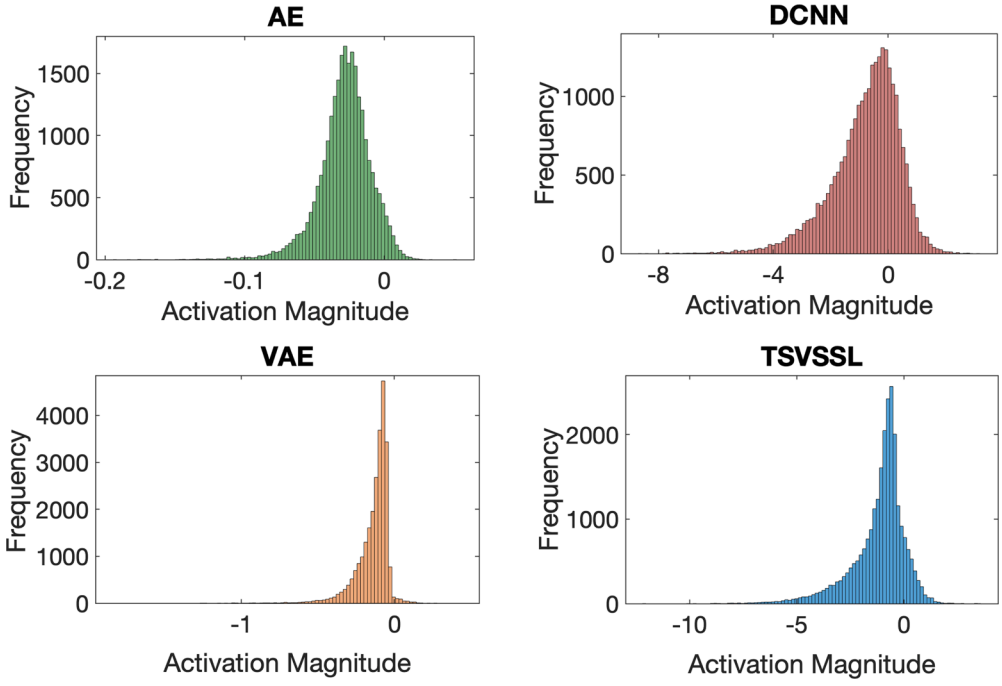


Fig. 6. The activation magnitude for "Conv4" layer across all four models.

and AE exhibit narrow distributions with low variance, suggesting weak feature extraction, which likely contributes to their poor class separation. Although DCNN has a wider activation distribution compared to AE and VAE, it still lags behind TSVSSL in capturing diverse features. This is because the performance of DCNN is constrained by the limited availability of labelled data, while TSVSSL benefits from both labelled and unlabelled data.

The performance observed in the previous discussion is attributed to the fundamental differences in the utilisation of labelled and unlabelled data by each model. While DCNN is found effective for classification tasks, it is limited by its dependence on labelled data affecting its performance in situations where labelled data is scarce. Although the AE model is designed to exploit the larger pool of unlabelled, the deterministic latent space limits its ability for supervised damage detection tasks. On the other hand, TSVSSL and VAE share the same architecture but differ in their training procedures. VAE operate in a two-step process, where latent features are learned first, followed by supervised classification. The first phase focuses solely on the reconstruction of the data distribution across the entire dataset, which limits its ability to effectively decouple the features for the classification phase in the second step. In contrast, TSVSSL benefits from the joint training of the encoder-decoder and classifier, and this simultaneous training contributes to its superior performance, as it gives equal attention to both reconstruction and classification.

6. Data augmentation

A rejection sampling technique is proposed for generating class-specific samples for downstream applications. For this purpose, the trained decoder and classifier network of the TSVSSL algorithm are utilised in a conditional framework to generate artificial data. The proposed technique begins by generating latent vector from a multivariate standard normal distribution $\mathcal{N}(0, I)$. These latent vectors are then decoded using a pre-trained decoder to produce candidate sample time series corresponding to the three *dofs* of the platform. However, to accept a sample as valid artificial data, the candidate must meet two criteria: the predicted class from the classifier must align with the target class, and the associated class probability must exceed a predefined acceptance threshold. The threshold parameter (τ) in the rejection sampling technique plays a crucial role in determining the confidence level required for accepting generated samples as valid class-specific data. In this study, the threshold parameter (τ) was set to 0.95, corresponding to a 95 % confidence threshold for the predicted class probability. This value was empirically determined through preliminary experiments to

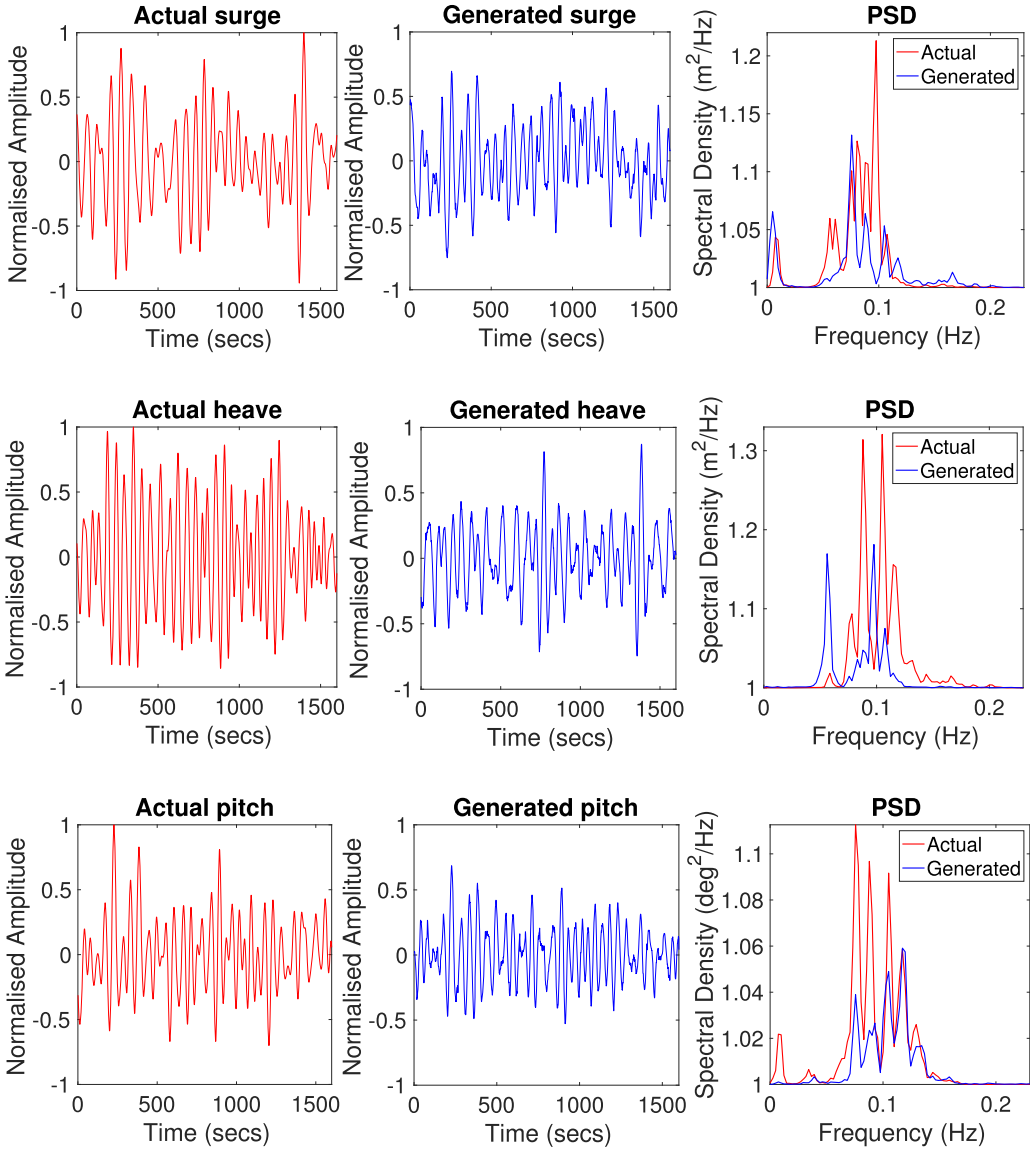


Fig. 7. Comparison of the actual and artificial time-series for Surge, heave and pitch, respectively, using the rejection sampling technique.

Algorithm 2 Rejection sampling algorithm.

Initialization:
Define the target class c_{target}
Define the acceptance threshold τ
Load pre-trained Decoder model D with decoder weights θ_D
Load pre-trained Classifier model C with classifier weights θ_C
Define the number of desired samples n_{samples}
Initialise list of accepted samples accepted_samples ← []
Sampling Loop:
while length of accepted_samples < n_{samples} **do**
  Sample latent vectors z from $\mathcal{N}(0, I)$: $z \sim \mathcal{N}(0, I)$
  Decode the latent vectors to generate time-series: $\hat{x} \leftarrow D(z)$
  Classify the generated time-series: $\hat{y} \leftarrow C(\hat{x})$
  Compute predicted class probabilities: $p_{\text{pred}} \leftarrow \text{softmax}(\hat{y})$
  for each generated sample i in the batch **do**
    if $\arg\max(p_{\text{pred}}[i]) = c_{\text{target}}$ **and** $\max(p_{\text{pred}}[i]) > \tau$ **then**
      Append $\hat{x}_i$ to accepted_samples
    end if
  end for
end while

achieve a balance between the quality and quantity of the generated samples. An extremely high (τ) ensures that only high-confidence samples are accepted, leading to reduce the overall sample acceptance rate. Conversely, lowering (τ) increases the number of accepted samples but risks compromising data quality by including low-confidence samples. The selection of $(\tau) = 0.95$ reflects an optimal trade-off observed during experimentation, ensuring that the generated artificial data aligns closely with the intended class characteristics while maintaining efficiency. Algorithm 2 outlines the rejection sampling process for generating damage-specific samples using sampled latent vectors from a multivariate normal distribution.

Fig. 7 presents a comparison between the actual and reconstructed motions for surge, heave, and pitch using the proposed rejection sampling technique. The plots correspond to three time series: surge for the healthy state, heave for the biofouling case and pitch for anchor sliding. While a direct time-domain comparison of these time series is not feasible, their comparison in the frequency domain provides valuable insights. The power spectral density (PSD) using the Welch method for both the normalized real and reconstructed signals are compared. The satisfactory alignment of the PSD between actual and generated signals implies that the generator is capturing the essential features of signals. Although some minor differences exist in frequency content, the generator demonstrates significant potential in reflecting the fundamental frequency features of the original motions.

Although the frequency domain comparison shows a good resemblance, quantifying this similarity poses challenges. Therefore, the proposed Maximum Mean Discrepancy (MMD) is used to compare the data distributions using actual and generated samples as suggested in Gretton et al. (2012). Fig. 8 presents the MMD values for 100 generated samples of different classes using rejection sampling. The analysis covers three distinct classes: healthy, biofouling and anchor displacement, highlighting the distributional differences between the generated and actual data. The healthy and anchor states exhibit nearly identical MMD scores of 0.0209 and 0.0208, respectively, both of which are minimal values, indicating that the generated samples for these health states closely resemble the actual data distribution, while the biofouling state shows a slightly higher MMD value of 0.0218. Overall, the generative model demonstrates strong performance across all health states, with MMD values indicating a satisfactory match between the generated and actual data.

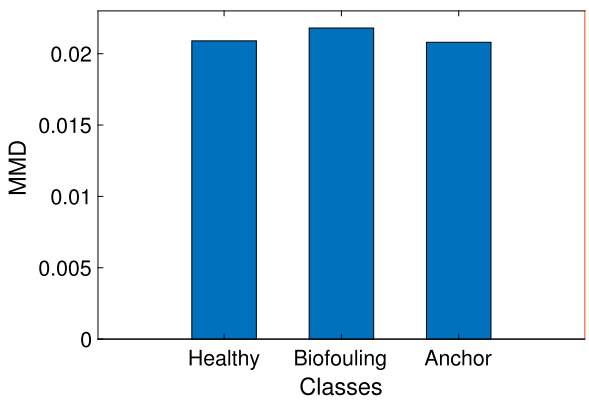


Fig. 8. Comparison of MMD values for different classes with TSVSSL.

7. Conclusions

This paper introduces a novel semi-supervised learning based on the VAE to address the challenges posed by limited labelled data for mooring system damage detection. The proposed TSVSSL framework advances the SSVAE approach by tailoring it specifically for time-series data. TSVSSL incorporates a shared encoder, decoder, and classifier network, which enables the simultaneous extraction of spatial-temporal features, data reconstruction, and class label prediction. Through unique strategy training adopted for the encoder-decoder and classifier, the framework produces a well clustered latent space that supports classification and synthetic data generation. This structured latent space is facilitated by the innovative introduction of a marginalization term in the unsupervised loss, which uses predictive class probabilities to regularise the latent space. Additionally, the framework includes a rejection sampling technique for generating artificial time-series data by combining a trained decoder with a classifier to ensure that generated data meets specific class criteria. These advancements significantly enhance the ability to detect damage and address class imbalances in scenarios like offshore mooring systems with limited labelled data. The key concluding points of this study are presented below:

- The TSVSSL algorithm consistently outperforms other deep learning models, exhibiting superior accuracy and robustness under various levels of labelled data. Its strong performance highlights the effectiveness of the proposed simultaneous training approach to utilise both labelled and unlabelled data efficiently.
- The comparison of latent space representations through the t-SNE underscores the superior performance of the TSVSSL algorithm, which exhibits well-clustered damage states in contrast to the other models. This structured latent space facilitates effective damage detection and data generation.
- The normalised confusion matrices for the different models reveal that detecting anchor damage is less complex than identifying other damage states. In particular, the TSVSSL algorithm shows the lowest misclassification rate highlighting effective feature retention.
- The proposed rejection sampling technique for artificial data generation demonstrates that the reconstructed time-series aligns with the actual one with respect to frequency characteristics. In addition, the MMD measures confirm the minimal differences between the actual and generated data distributions.

Overall, the proposed TSVSSL algorithm exhibits a strong potential for damage detection and the generation of artificial data with limited labelled data. The results underscore the capability to achieve reasonable accuracy (>81 %) with as little as 20 % labelled data, providing a practical threshold for applications with limited labelled datasets for this problem. The proportion of labelled data for the same level of accuracy may vary depending on the specific problem being addressed and the extent or type of damage under consideration, as different damage scenarios may exhibit varying levels of sensitivity to labelled data availability. Further, these results are limited to the numerical

experiment conducted on the 5 MW FOWT, where the wave angle, wind propagation, and current direction are all aligned. In this setup, the natural frequencies along surge, heave, and pitch change due to the damage. Further studies are needed to investigate the impact of misalignment between the wave angle, wind propagation, and current direction on the natural frequencies of the degrees of freedom (DOFs). Furthermore, it is crucial to test its performance with real datasets, as the analysis carried out in this study has only used simulated data. Hence, validating these findings with real-world data will be the future work to further confirm their effectiveness and applicability in practical settings. One of practical applications of the TSVSSL framework can be its integration into real-time monitoring of mooring systems for offshore platforms. Using onboard sensors such as accelerometers and gyroscopes, the framework can continuously process time-series motion data to dynamically assess the health of mooring systems. The encoder extracts latent features from the continuous feed of the real time data, the classifier identifies any deviations in latent space to signify any potential damage. Nevertheless, the positive outcomes of this study suggest that the proposed technique has significant potential for applications in structural health monitoring using time-series data.

Data availability

Data and codes are available upon request.

CRediT authorship contribution statement

Pranjal Tamuly: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **Smriti Sharma:** Writing – original draft, Validation, Software, Data curation; **Vincenzo Nava:** Writing – review & editing, Writing – original draft, Supervision, Resources, Project administration, Funding acquisition.

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.

Acknowledgements

This research is supported by funding from the Spanish Ministry of Economic Affairs and Digital Transformation as part of the Recovery, Transformation, and Resilience Plan, specifically through the R&D Missions within the Artificial Intelligence 2021 Programme. The financial support is provided under the IA4TES project (Artificial Intelligence for Sustainable Energy Transition) with reference number MIA.2021.M04.008. The authors also acknowledge funding from the ELKARTEK project RUL-ET by the Basque Government (KK-2024/00086); the "BCAM Severo Ochoa" excellence accreditation (CEX2021-001142-S / MICIN / AEI / 10.13039/501100011033); and the Basque Government's BERC 2022–2025 program. The authors further extend their gratitude to the Donostia International Physics Center (DIPC) for their collaboration and support in providing high-performance computing (HPC) resources.

References

Al-Jabri, K.S., Al-Kuwari, A.H., Swamidas, A.S.J., 2006. Damage detection in offshore structures using vibration response. J. Offshore Mech. Arct. Eng. 128 (1), 61–67.

Brindley, W., Comley, A.P., 2014. North Sea mooring systems: how reliable are they? In: International Conference on Offshore Mechanics and Arctic Engineering. Vol. 45370. American Society of Mechanical Engineers, p. V01AT01A025.

Chen, W., Li, M., Zhang, L., 2023a. Study on mooring design of 15 MW floating wind turbines in south china sea. J. Mar. Sci. Eng. 12 (1), 33. https://doi.org/10.3390/jmse12010033

Chen, W., Zhang, L., Wang, H., 2023b. Investigation on local mooring stresses of floating offshore wind turbines considering mooring chain geometrical and material nonlinearity. Ocean Eng. 162, 301–315. https://doi.org/10.1016/j.oceaneng.2023.105123

Chen, W., Zhang, L., Wang, H., 2023c. A simulation technique for monitoring the real-time stress responses of various mooring configurations for offshore floating wind turbines. Ocean Eng. 278, 114366. https://doi.org/10.1016/j.oceaneng.2023.114366

Chen, X., Wang, Z., Zhang, Z., Jia, L., Qin, Y., 2018. A semi-supervised approach to bearing fault diagnosis under variable conditions towards imbalanced unlabeled data. Sensors 18 (7), 2097.

Coulling, A.J., Goupee, A.J., Robertson, A.N., Jonkman, J.M., Dagher, H.J., 2013. Validation of a FAST semi-submersible floating wind turbine numerical model with deepcwind test data. J. Renew. Sustain. Energy 5 (2), 023116.

D'Agostino, P., Borri, C., Benassai, M., 2019. Strength and durability of synthetic mooring ropes. Mar. Struct. 66, 102653. https://doi.org/10.1016/j.marstruc.2019.102653

Damm, S., Forster, D., Velychko, D., Dai, Z., Fischer, A., Lücke, J., 2023. The ELBO of variational autoencoders converges to a sum of three entropies. 2010.14860. https://arxiv.org/abs/2010.14860.

Decurey, B., Schoefs, F., Barillé, A.-L., Soulard, T., 2020. Model of bio-colonisation on mooring lines: updating strategy based on a static qualifying sea state for floating wind turbines. J. Mar. Sci. Eng. 8 (2), 108.

Duarte, T., Almeida, S., Estefen, S., 2018. Maintenance of offshore wind turbines: issues, trends, and approaches. Renew. Sustain. Energy Rev. 89, 379–390. https://doi.org/10.1016/j.rser.2018.03.024

Engelmann, J., Lessmann, S., 2021. Conditional wasserstein GAN-based oversampling of tabular data for imbalanced learning. Expert Syst. Appl. 174, 114582.

Farrar, C.R., Worden, K., 2012. Structural Health Monitoring: A Machine Learning Perspective. Wiley.

Gao, Y., Kong, B., Mosalam, K.M., 2019. Deep leaf-bootstrapping generative adversarial network for structural image data augmentation. Computer-Aided Civ. Infrastruct. Eng. 34 (9), 755–773.

Gao, Y., Zhai, P., Mosalam, K.M., 2021. Balanced semisupervised generative adversarial network for damage assessment from low-data imbalanced-class regime. Computer-Aided Civ. Infrastruct. Eng. 36 (9), 1094–1113.

Gao, Z., Zhang, Z., Moan, T., 2010. Fatigue analysis of mooring chains in offshore structures. J. Offshore Mech. Arct. Eng. 132 (2), 021401. https://doi.org/10.1115/1.4001522

Gorostidi, N., Pardo, D., Nava, V., 2023. Diagnosis of the health status of mooring systems for floating offshore wind turbines using autoencoders. Ocean Eng. 287, 115862.

Gretton, A., Borgwardt, K.M., Rasch, M.J., Schölkopf, B., Smola, A., 2012. A kernel two-sample test. J. Mach. Learn. Res. 13 (1), 723–773.

Harvey, W., Naderiparizi, S., Wood, F., 2021. Conditional image generation by conditioning variational auto-encoders. arXiv preprint arXiv:2102.12037

International Renewable Energy Agency (IRENA), 2024. Renewable Energy Statistics 2024. International Renewable Energy Agency, Abu Dhabi.

Jahangiri, V., Mirab, H., Fathi, R., Ettefagh, M.M., 2016. Tlp structural health monitoring based on vibration signal of energy harvesting system. Latin Am. J. Solids Struct. 13, 897–915.

Jonkman, J., Butterfield, S., Musial, W., Scott, G., 2009. Definition of a 5-MW Reference Wind Turbine for Offshore System Development. Technical Report. National Renewable Energy Laboratory. United States. https://doi.org/10.2172/947422

Kingma, D.P., 2013. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114

Kingma, D.P., Welling, M., 2014. Auto-encoding variational bayes. In: International Conference on Learning Representations (ICLR).

Kingma, D.P., Welling, M., 2022. Auto-encoding variational bayes. 1312.6114. https://arxiv.org/abs/1312.6114.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521 (7553), 436–444.

Lei, Y., Jia, F., Lin, J., Xing, S., Ding, S.X., 2016. An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data. IEEE Trans. Ind. Electron. 63 (5), 3137–3147.

Liu, C., Gryllias, K., 2020. A semi-supervised support vector data description-based fault detection method for rolling element bearings based on cyclic spectral analysis. Mech. Syst. Signal Process. 140, 106682.

Ma, K.-t., Shu, H., Smedley, P., L'Hostis, D., Duggal, A., 2013. A historical review on integrity issues of permanent mooring systems. In: Offshore Technology Conference. OnePetro.

Ma, X., Lin, Y., Nie, Z., Ma, H., 2020. Structural damage identification based on unsupervised feature-extraction via variational auto-encoder. Measurement 160, 107811.

Van der Maaten, L., Hinton, G., 2008. Visualizing data using t-SNE. J. Mach. Learn. Res. 9 (11), 2579–2605.

Medina-Manuel, A., Molina Sánchez, R., Souto-Iglesias, A., 2024. Ai-driven model prediction of motions and mooring loads of a spar floating wind turbine in waves and wind. J. Mar. Sci. Eng. 12 (9), 1464. https://doi.org/10.3390/jmse12091464

Myhr, A., Bjerkseter, C., Ågotnes, A., Nygaard, T.A., 2014. Levelised cost of energy for offshore floating wind turbines in a life cycle perspective. Renew. Energy 66, 714–728. https://doi.org/10.1016/j.renene.2014.01.017

Narasimhan, S., Nagarajaiah, S., 2018. Machine learning algorithms for structural health monitoring of civil infrastructure. Curr. Opin. Civ. Eng. 1, 38–45.

Nguyen, P.V., Soares, C.G., 2020. Long-term fatigue analysis of mooring lines using a time-domain approach. Ocean Eng. 211, 107552. https://doi.org/10.1016/j.oceaneng.2020.107552

Pan, T., Chen, J., Xie, J., Chang, Y., Zhou, Z., 2020. Intelligent fault identification for industrial automation system via multi-scale convolutional generative adversarial network with partially labeled samples. ISA Trans. 101, 379–389.

Rafiei, M.H., Adeli, H., 2018. A novel machine learning model for estimation of critical response parameters in large-scale structures subjected to earthquake. Eng. Appl. Artif. Intell. 73, 24–36.

Razavi-Far, R., Hallaji, E., Farajzadeh-Zanjani, M., Saif, M., 2018. A semi-supervised diagnostic framework based on the surface estimation of faulty distributions. IEEE Trans. Ind. Inf. 15 (3), 1277–1286.

Santos, A. B.M., Rosas, H.C., 2021. Analysis of mooring line failures in offshore platforms. Appl. Ocean Res. 111, 102651. https://doi.org/10.1016/j.apor.2021.102651

Schmidhuber, J., 2015. Deep learning in neural networks: an overview. Neural Netw. 61, 85–117.

Sharma, S., Nava, V., 2024. Condition monitoring of mooring systems for floating offshore wind turbines using convolutional neural network framework coupled with autoregressive coefficients. Ocean Eng. 302, 117650.

Sharma, S., Nava, V., Gorostidi, N., et al., 2023. Monitoring mooring (monimoor) lines of floating structures using deep learning-based approaches. In: Proceedings of the 9th ECCOMAS Thematic Conference on Computational Methods in Structural Dynamics and Earthquake Engineering (COMPDYN 2023), Athens, Greece, pp. 158–171.

Sharma, S., Subhamoy, S., 2020. One-dimensional convolutional neural network-based damage detection in structural joints. J. Civ. Struct. Health Monit. 10 (5), 1057–1072.

Sharma, S., Subhamoy, S., 2021. Bridge damage detection in presence of varying temperature using two-step neural network approach. J. Bridge Eng. 26 (6), 04021027.

Sidarta, D.E., Auburtin, E., Ledoux, A., Lim, H.-J., Leridon, A., Tcherniguin, N., 2023. Mooring line failure detection using artificial neural networks: an application to field data including artificial failure cases. In: Offshore Technology Conference. OTC, p. D031S032R007.

Soliman, Y.M.A., El-Shanawany, M.M., 2015. Probabilistic modeling of fatigue damage in mooring lines. Eng. Struct. 85, 39–47. https://doi.org/10.1016/j.engstruct.2014.11.010

Stambaugh, I.M., Inalpolat, M., 2016. Damage detection in offshore structures through acoustic emission monitoring and signal processing. In: Proceedings of the ASME 2016 35th International Conference on Ocean, Offshore and Arctic Engineering. American Society of Mechanical Engineers (ASME), p. V002T08A039.

Sun, Y., Li, Z., Wang, Y., 2018. An overview of damage detection techniques for structures. J. Civ. Struct. Health Monit. 8 (4), 649–670.

Tygesen, U.T., Jepsen, M.S., Vestermark, J., Dollerup, N., Pedersen, A., 2018. The true digital twin concept for fatigue re-assessment of marine structures. In: International Conference on Offshore Mechanics and Arctic Engineering. Vol. 51203. American Society of Mechanical Engineers, p. V001T01A021.

Verstraete, D.B., Droguett, E.L., Meruane, V., Modarres, M., Ferrada, A., 2020. Deep semi-supervised generative adversarial fault diagnostics of rolling element bearings. Struct. Health Monit. 19 (2), 390–411.

Wang, M.L., Zhao, Y., Haritos, N., 2001. Structural monitoring of offshore platforms. J. Offshore Mech. Arct. Eng. 123 (3), 151–156.

Yao, H., Wu, C., Zhang, Y., 2021. Deep learning for structural health monitoring: a review. Mech. Syst. Signal Process. 148, 107130.

Yuen, K.-V., Kuok, S.-C., Kuok, K.-K., 2016. Inverse problems in structural health monitoring of civil engineering structures: a review. Struct. Control Health Monit. 23 (3), 409–422.

Zhang, H., Yang, Y., Wang, X., 2020. Deep learning for structural health monitoring: a review. Struct. Health Monit. 19 (4), 1059–1079.

Zhao, M., Li, B., Qi, J., Ding, Y., 2017. Semi-supervised classification for rolling fault diagnosis via robust sparse and low-rank model. In: 2017 IEEE 15th International Conference on Industrial Informatics (INDIN). IEEE, pp. 1062–1067.