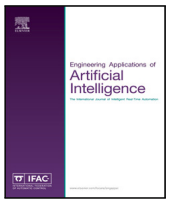
Engineering Applications of Artificial Intelligence 139 (2025) 109477

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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



A hybrid Convolutional Autoencoder training algorithm for unsupervised bearing health indicator construction

Ali Eftekhari Milani *, Donatella Zappalá, Simon J. Watson

TU Delft, Kluyverweg 1, Delft, 2629 HS, Netherlands

ARTICLE INFO

Keywords:
Particle Swarm Optimisation
Convolutional Autoencoder
Bearing
Health indicator
Prognostics
Remaining useful life prediction

ABSTRACT

Conventional Deep Learning (DL) methods for bearing health indicator (HI) adopt supervised approaches, requiring expert knowledge of the component degradation trend. Since bearings experience various failure modes, assuming a particular degradation trend for HI is suboptimal. Unsupervised DL methods are scarce in this domain. They generally maximise the HI monotonicity built in the middle layer of an Autoencoder (AE) trained to reconstruct the run-to-failure signals. The backpropagation (BP) training algorithm is unable to perform this maximisation since the monotonicity of HI subsections corresponding to input sample batches does not guarantee the monotonicity of the whole HI. Therefore, existing methods achieve this by searching AE hyperparameters so that its BP training to minimise the reconstruction error also leads to a highly monotonic HI in its middle layer. This is done using expensive search algorithms where the AE is trained numerous times using various hyperparameter settings, rendering them impractical for large datasets. To address this limitation, a small Convolutional Autoencoder (CAE) architecture and a hybrid training algorithm combining Particle Swarm Optimisation and BP are proposed in this work to enable simultaneous maximisation of the HI monotonicity and minimisation of the reconstruction error. As a result, the HI is built by training the CAE only once. The results from three case studies demonstrate this method's lower computational burden compared to other unsupervised DL methods. Furthermore, the CAE-based HIs outperform the indicators built by equivalent and significantly larger models trained with a BP-based supervised approach, leading to 85% lower remaining useful life prediction errors.

1. Introduction

Bearings are one of the most critical components in many machines. According to numerous studies about the distribution of failures in different wind turbine components, bearings are by far the most likely to fail among the sub-assemblies (Rezamand et al., 2020). In electric motors, depending on the type and size of the machine, bearing faults account for about 40% to about 90% of all failures (Bianchini et al., 2011). Therefore, the successful prediction of the remaining useful life (RUL) of bearings can lead to a considerable increase in system availability and a reduction in operations and maintenance (O&M) costs by preventing unplanned failures and affording adequate time to optimally plan maintenance interventions.

The construction of Health Indicators (HIs), which quantify the bearings' level of degradation through time, is a crucial step towards successful RUL prediction. Since bearing degradation is a highly monotonic process, resulting from irreversible changes starting at the micro-scale and propagating gradually until failure (Kipchirchir et al., 2023), monotonic trends in the constructed HIs represent a key requirement and play a crucial role in defining reliable prognostic approaches (Guo et al., 2018). The HI monotonic trend is commonly assessed by criteria such as monotonicity and trendability (Moradi et al., 2023).

During the last decade, due to the availability of industrial big data, numerous data-driven approaches have been successfully applied for bearing HI construction (Zhou et al., 2022). Traditional HI construction methods generally involve the manual extraction of features from raw signals in the time, frequency, and time–frequency domains (Rai and Upadhyay, 2016), followed by the application of feature selection and fusion. Statistical measures, such as Variance (Zhang et al., 2023), Root Mean Square (RMS) (Wang et al., 2023), Kurtosis, and Skewness (Ding et al., 2021; Wang et al., 2016) have been used for extracting degradation features from the time domain. Furthermore, various methods, such as Fourier and Wavelet transform (Yuan et al., 2017) and Empirical Model Decomposition (Lu et al., 2016), have been extensively applied for feature extraction from the frequency and time–frequency domains. Most of these approaches rely on complex and labour-intensive feature selection and extraction techniques and

* Corresponding author.
E-mail addresses: a.eftekharimilani@tudelft.nl (A.E. Milani), d.zappala@tudelft.nl (D. Zappalá), s.j.watson@tudelft.nl (S.J. Watson).

https://doi.org/10.1016/j.engappai.2024.109477
Received 3 April 2024; Received in revised form 12 September 2024; Accepted 12 October 2024
Available online 30 October 2024

Table 1
Deep Learning-based approaches for bearing HI construction validated on the PHM 2012 dataset.

Paper	Input	Learning	Model	Degradation trend
Guo et al. (2017)	Hand-crafted features	Supervised	RNN	Linear
Yoo and Baek (2018)	Time–frequency domain	Supervised	CNN	Linear
Guo et al. (2018)	Time domain	Supervised	CNN	Linear
Ren et al. (2018)	Time–frequency domain	Supervised	CNN	Linear
Wu et al. (2019)	Various	Supervised	CNN	Various
Li et al. (2019)	Time–frequency domain	Supervised	CNN	Linear
She and Jia (2019)	Time domain	Supervised	CNN	Linear
Chen et al. (2021)	Time domain	Supervised	CAE	Quadratic
Jiang et al. (2022)	Time domain	Supervised	CNN-RNN	Quadratic
Wang et al. (2022)	Time domain features	Supervised	CNN	Linear
Zhuang et al. (2024)	Time and Time–frequency domain	Supervised	Attention	Linear

adopt conventional and shallow Machine Learning methods for feature fusion (Sim et al., 2023). As a consequence, they usually lack prognostic capability and adaptability (Yoo and Baek, 2018).

Recently, various Deep Learning (DL) methods have been developed for constructing HIs from sensor signals of degrading equipment. These methods learn features directly from signals in time (Wang et al., 2022), frequency (Wu et al., 2019), or time–frequency domain (Zhuang et al., 2024), obviating the need for lengthy and, in some cases, computationally expensive feature extraction and selection procedures. This new trend has been partly enabled by the increased availability of open-access datasets, allowing researchers to test and validate their methods. One example is the PHM 2012 bearing prognostic challenge dataset (Nectoux et al., 2012), which contains run-to-failure bearing vibration data related to 17 experiments performed under three distinct operational conditions and has been widely used as a benchmark for data-driven bearing prognostic methods. Table 1 summarises the key features of some of these methods in terms of the type of input data used, the learning approach, the DL model implemented, and the assumed degradation trend.

All these methods implement supervised learning for HI construction, requiring prior expert knowledge of the degradation trend. The majority of these approaches assume that bearings degrade linearly with time. This assumption oversimplifies the problem and does not capture the stochastic nature intrinsic to most complex degradation processes. Adopting supervised methods requires not only the assumption of a specific HI regime, such as linear or exponential, but also the knowledge of the parameters describing it. This introduces strong constraints and limits the indicator generalisation ability and adaptability to describe various bearing failure modes. Since degradation can develop with different trends and speeds under various working conditions, assuming the correct behaviour without in-depth information on the true failure mechanism is not trivial and often requires expert domain knowledge.

These limitations can be overcome by adopting an unsupervised approach for HI construction, which does not require prior knowledge of the degradation mechanism (Wen et al., 2024). However, despite some applications in fault detection (Cha and Wang, 2017) and diagnosis (Tao et al., 2023), very few unsupervised DL methods have been proposed in the literature for bearing HI construction. Unsupervised methods based on Gaussian mixtures and on manual selection and fusion of time domain features are proposed in Wen et al. (2024) and Duong et al. (2018), respectively. This work focuses on DL-based unsupervised methods due to the proven superiority of these approaches in feature extraction and fusion. These methods generally adopt an Autoencoder (AE) (Wang and Cha, 2020), which reconstructs the input signals in its output, passing them through a bottleneck in its middle layer where the HI is built. Since degradation is generally an irreversible process, the closer the HI is to a monotonic function, the closer it describes the degradation trend. Therefore, the objectives are the minimisation of the AE reconstruction error and the maximisation of the HI monotonicity. Guo et al. (2022) proposes a multi-scale CAE, an unsupervised framework, to extract several features from signals in the time domain. They are then weighted and fused to construct the HI. The CAE hyperparameters are set using a Genetic Algorithm (GA), and the feature weights are obtained by a quadratic programming-based optimisation of a trendability metric. The GA-based setting of the hyperparameters entails a heavy computational burden since several CAEs are trained by backpropagation (BP) until convergence at each algorithm generation. In addition, the extracted features are fused manually, forgoing one of the most valuable benefits of DL-based HI construction methods, *i.e.*, automatic feature extraction and fusion. In Yang et al. (2018), an unsupervised method based on a Deep Sparse Autoencoder (SAE) is proposed. Maximising the monotonicity of small portions of the HI corresponding to batches of SAE input samples does not guarantee the monotonicity of the whole HI. Therefore, it is impossible to train the SAE using BP to maximise the HI monotonicity along with minimising the reconstruction error. Therefore, in Yang et al. (2018), the SAE architecture and the hyperparameters are searched by trial and error so that the SAE builds a highly monotonic HI when trained by BP to minimise the reconstruction error. In a follow-up paper (Yang et al., 2022), a random search algorithm is developed to set the SAE hyperparameters. Although it has the benefit of removing the need for trial and error, this approach leads to a heavy computational burden. To overcome this problem, in Milani et al. (2020), the SAE architecture and hyperparameters are set using a multi-objective co-operative co-evolutionary optimisation algorithm. Despite being more time-efficient compared to trial and error and random search, this method is still computationally expensive since several SAEs are trained by BP until convergence at each generation of the co-evolutionary algorithm. Therefore, it is not feasible in real-world applications with large signal datasets.

To overcome these limitations and enable an efficient implementation of unsupervised DL for bearing HI construction, this work proposes a CAE that is trained using a novel hybrid algorithm. This algorithm combines Particle Swarm Optimisation (PSO) (Poli et al., 2007) and BP to simultaneously maximise the HI monotonicity in its middle layer and minimise the reconstruction error. CAE is adopted instead of AE because convolutional layers have been shown to be more effective in feature extraction compared to fully connected layers (Guo et al., 2018), both when the features have a meaningful ordering, like vibration amplitudes at different frequencies, and when they do not, like its application to wind turbine data (Kong et al., 2020; Xiang et al., 2021). Furthermore, PSO is selected because, other than being a good alternative to BP training (Suresh et al., 2015), it does not require a sequential input of small batches of samples. Therefore, it can be used to train the CAE to maximise the HI monotonicity. However, despite its superiority in traversing the search space towards the global optimum, PSO under-performs BP in local searches (Ismail et al., 2013), leading to slower convergence. This necessitates the adoption of a hybrid approach, combining PSO and BP. In the existing approaches (Ismail et al., 2013; Zhang et al., 2007; Yaghini et al., 2013), the networks are initially trained by PSO, and at the final stages of the training, BP is adopted to continue the training until convergence is attained. Therefore, since BP cannot be used to maximise HI monotonicity, it

cannot be used to train a CAE for this objective. For this reason, this work proposes a new training algorithm, where in each generation PSO and BP are implemented in tandem to update CAE weights and achieve an efficient optimisation of both objectives.

The effectiveness of the proposed method is demonstrated in three case studies: case study 1 using the synthetic dataset from Yang et al. (2018), Milani et al. (2020), and case studies 2 and 3 using the IEEE PHM 2012 (Nectoux et al., 2012) and the XJTU-SY (Wang et al., 2020) bearing prognostic datasets respectively, which have been used by many researchers for the evaluation of bearing HI construction methods. The key contributions of this work are:

- a small but effective CAE architecture for unsupervised HI construction that outperforms established models with similar sizes;
- a novel hybrid CAE training algorithm based on PSO and BP, which, unlike previous approaches proposed in the literature, enables the simultaneous maximisation of the HI monotonicity and minimisation of the reconstruction error;
- a reduction of computational burden for HI construction compared to the existing unsupervised DL methods, which allows its applicability to large datasets;
- the construction of HIs that follow the true component degradation trend, leading to their superior performance in terms of the 3 metrics of monotonicity, failure prediction accuracy, and remaining useful life prediction compared to those built by similar and larger models trained with a supervised approach.

The rest of this paper is organised as follows. The proposed CAE architecture, HI construction and hybrid training algorithm are presented in Section 2. The datasets, experiments, and results for case studies 1 through 3 are detailed in Sections 3 through 5. Conclusions are drawn, and future research steps are discussed in Section 6.

2. Methodology

This section describes the proposed CAE architecture, the CAE-based HI construction, and the developed hybrid CAE training algorithm.

2.1. CAE architecture and HI construction

An AE is a neural network trained to deconstruct and reconstruct the input signals, passing them through a bottleneck in its inner layer. The general architecture of an AE is shown in Fig. 1. It consists of two main modules: the Encoder (E) and the Decoder (D) (Hinton and Salakhutdinov, 2006). The Encoder maps the high-dimensional input signals $\vec{\mathbf{x}} \subset \mathbb{R}^F$ to a low-dimensional latent representation $\vec{\mathbf{z}} \subset \mathbb{R}^G$, where $F \gg G$, and the Decoder reconstructs the original signals and outputs $\hat{\vec{\mathbf{x}}} \subset \mathbb{R}^F$ by mapping the latent low-dimensional representation output from the Encoder to the original high-dimensional space. Formally:

$$\hat{\vec{\mathbf{x}}} = D\left(E\left(\vec{\mathbf{x}}\right)\right) \approx \vec{\mathbf{x}} \tag{1}$$

An AE learns to maintain only the information required for reconstructing the signals in the inner layer, yielding a condensed latent space representation of the input signals. In its simplest form, an AE is a feedforward neural network built up of three fully connected layers with F, G, and F number of neurons.

Fully connected layers introduce redundancy in the parameters by forcing each feature to be global and span the entirety of the input feature space. This work adopts a Convolutional AE (CAE) consisting of convolutional rather than fully connected layers. The weights of the convolutional layers are shared among all the locations in the input, allowing the discovery of localised features (Masci et al., 2011). As a result, convolutional layers can extract robust and well-performing features with significantly fewer parameters than fully connected layers.

A typical bearing sensor signal dataset is organised in a two-dimensional matrix with dimensions equal to the number of signals

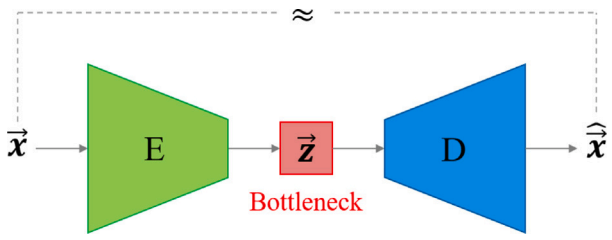


Fig. 1. Typical architecture of an AE: E is the Encoder, and D is the Decoder.

(feature domain) and the number of samples (time domain). Depending on the sensor type, the feature domain can consist of vibration data in the time or frequency domain or other data sources, such as temperature at different bearing locations. When dealing with bearing vibration data, several studies proposed in the literature for bearing HI construction, including (Yoo and Baek, 2018; Ren et al., 2018; Li et al., 2019), show that extracting features from the time–frequency domain rather than only from the time or the frequency domain leads to improved HI construction. This is expected to hold when the feature domain consists of data sources other than vibration data. Therefore, the CAE architecture proposed in this study consists of 2D convolutional layers using time-feature input matrices instead of 1D convolutional layers relying only on time or feature domain inputs.

At each time instance t of a bearing run-to-failure dataset, this value and the previous $W-1$ instances $(t-W+1,\ldots,t-1,t)$ are used to create a sampled subset of the original dataset. In this manner, for a given $T \times F$ dataset, where T and F are the sizes of the time and feature domain, respectively, a new dataset containing $T-W+1$ samples is built. Each sample of size $W \times F$ is then used as input to the CAE. The data preparation process and the proposed CAE architecture are shown in Fig. 2. Two convolutional layers are used in the Encoder with ReLU activation functions. The first layer performs convolution only in the time domain and uses eight filters of shape $W \times 1$, transforming the input matrix to 8 matrices of shape $1 \times F$, *i.e.*, the shape of $1 \times F \times 8$. The second layer performs convolution in the feature domain and uses a filter of shape $1 \times F$. Considering that the first layer's output has 8 channels, the dot product between the filter with the shape of $1 \times F \times 8$ and the output of the first layer yields a scalar value, which becomes the HI. The Decoder has a symmetric architecture and transforms the output of the Encoder to the original $W \times F$ shape by upsampling and deconvolution in the feature and time domain, respectively. The scalar HI is transformed to $1 \times F$ shape by repeating it F times using an upsampling layer. Subsequently, a transposed convolution layer with 8 filters of shape $1 \times F$ is applied, transforming it to $1 \times F \times 8$ shape. The upsampling and transposed convolution are repeated in the time domain, where the output of the previous layer is upsampled to the shape $W \times F \times 8$ by repeating it W times, followed by a transposed convolution layer with one filter of shape $W \times 1$ which restores the original sample shape of $W \times F$.

2.2. Proposed hybrid training algorithm

2.2.1. Training loss function

Unsupervised CAE-based bearing HI construction includes training the CAE to reconstruct the bearing signals so that their latent space representation in the middle layer (the HI) is as close as possible to a monotonic function. The objectives are to train the CAE so that:

1. The output of the Encoder is highly monotonic. This means that the CAE has learned to amplify the monotonic degradation factor of the input signals and dampen the other non-monotonic components;

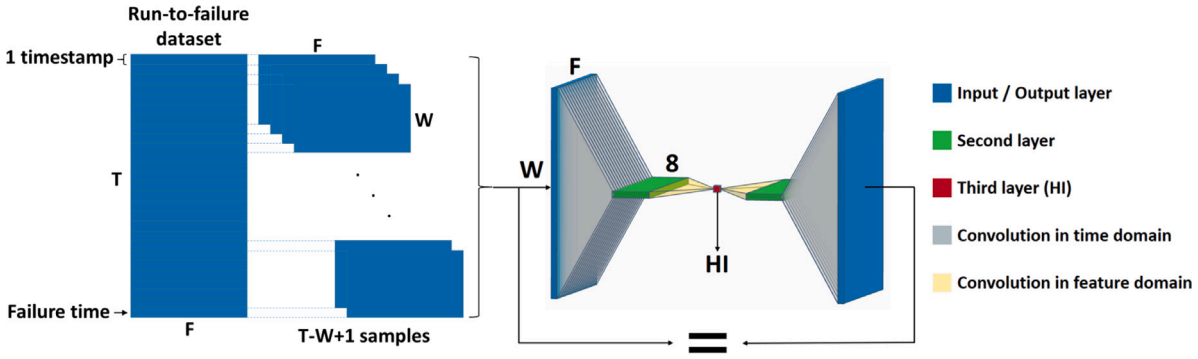


Fig. 2. CAE architecture.

2. The Decoder can successfully reconstruct the bearing signals from their condensed representation in the middle layer, ensuring that important information is not lost in the encoder.

In the literature, the monotonic trend of a HI is predominantly measured through a combination of the Monotonicity, Mon, and Trendability, Tre, metrics (Duong et al., 2018; She and Jia, 2019; Wang et al., 2022):

$$Mon(Z) = \frac{1}{n-1}\left|\sum_{i=1}^{n-1} sign(z_{i+1} - z_i)\right| \tag{2}$$

$$Tre(Z) = Corr(Z, T) \tag{3}$$

where $Z = z_1, z_2, \ldots, z_n$ is the considered HI sequence, T is the time vector, and $Corr$ is the Pearson correlation coefficient. However, Mon is very sensitive to noise, and its utilisation requires denoising the HI. Furthermore, combining two distinct metrics leads to added complexity. Therefore, in this work, the monotonic trend of the constructed HI is measured by the MK monotonicity metric, τ_{MK}, proposed in Pohlert (2015), and the two metrics Mon and Tre are only used for comparison with the literature.

$$\tau_{\text{MK}} = \frac{S}{P} \tag{4}$$

where S and P are defined as:

$$S = \sum_{k=1}^{n-1}\sum_{j=k+1}^{n} \text{sign}\left(z_j - z_k\right) \tag{5}$$

$$P = \frac{1}{2}n(n-1) \tag{6}$$

In Pohlert (2015), a similar metric called ρ_{MK} is proposed:

$$\rho_{\text{MK}} = \begin{cases} \frac{S-1}{\sigma} & \text{if} \quad S > 0 \\ 0 & \text{if} \quad S = 0 \\ \frac{S+1}{\sigma} & \text{if} \quad S < 0 \end{cases} \tag{7}$$

where $\sigma = \sqrt{\frac{n(n-1)(2n+5)}{18}}$. However, unlike ρ_{MK}, τ_{MK} is bounded between -1 and 1 and thus is suitable as a training loss function. For both of these metrics, a higher absolute value signifies a higher monotonicity and values close to zero indicate the lack of a monotonic trend. Positive values indicate an increasing trend and negative values indicate a decreasing trend. Therefore, the first training objective can be achieved by maximising $\tau_{\text{MK}}(HI)$, while the second objective is attained by minimising the reconstruction error between the CAE input and output measured using the Mean Squared Error (MSE):

$$MSE(\vec{\mathbf{x}}, \hat{\vec{\mathbf{x}}}) = \frac{1}{WF}\sum_{i=1}^{W}\sum_{j=1}^{F}\left(x_{i,j} - \hat{x}_{i,j}\right)^2 \tag{8}$$

with W representing the size of the time window of each time series sample, F the number of signals or the size of the feature domain, $\vec{\mathbf{x}} = x_{i,j}$, $i = 1, \ldots, W$, $j = 1, \ldots, F$ the elements of the input time series samples, and $\hat{\vec{\mathbf{x}}} = \hat{x}_{i,j}$, $i = 1, \ldots, W$, $j = 1, \ldots, F$ the elements of the reconstructed time series samples. The training fitness function, f, adopted in this study to achieve both training objectives is:

$$f = f_M - f_R \tag{9}$$

where the first term $f_M = \tau_{\text{MK}}(HI)$ is the monotonicity of the HI measured by the τ score of the MK monotonicity test, defined in Eq. (4), and $f_R = MSE(\vec{\mathbf{x}}, \hat{\vec{\mathbf{x}}})$ is the reconstruction error measured by the mean squared error between the input and the output of the CAE, defined in Eq. (8). To obtain comparable HIs for the different tests, the terms $-|HI(0)|$ and $-|HI(end) - 1|$, where end is the final time stamp of the HI, are added to the training cost function in Eq. (9) to obtain f' in Eq. (10):

$$f' = f - |HI(0)| - |HI(end) - 1| \tag{10}$$

In this way, as f' is maximised, the value $HI(0) = 0$ is always associated with the initial bearing state, and the value $HI(end) = 1$ is always associated with its failure time. This is conducive to an effective failure threshold selection.

2.2.2. PSO training algorithm

The second training objective can be achieved by training the CAE using BP. However, this training algorithm does not allow the maximisation of HI monotonicity since only a small portion of the run-to-failure sequence is input to the CAE at each BP iteration, while the monotonicity is measured for the whole HI sequence. Maximising the monotonicity of different HI portions does not guarantee a monotonic HI.

PSO is shown to be a good alternative to the BP training algorithm (Suresh et al., 2015), and it does not require a sequential input of small batches of samples. Therefore, it allows the optimisation of both objectives. In the PSO algorithm, a swarm of particles, which are points in a K-dimensional search space (*i.e.*, the space of the CAE weight parameters), is maintained and evolved for Γ generations. The number of CAE weight parameters, K, depends on the CAE architecture and input and output shape. Each particle in this space represents one set of CAE weights and evaluates the fitness function at its current location. It then determines its movement through the search space according to a combination of its current and best locations and the global best location with some random perturbations. Each particle in the swarm includes three vectors: the current position $\vec{y}_i$, the previous best position $\vec{p}_i$, and the velocity $\vec{v}_i$. The particle's location and velocity are updated according to:

$$\begin{cases} \vec{v}_i \leftarrow \vec{v}_i + \vec{U}\left(0, \phi_1\right) \otimes \left(\vec{p}_i - \vec{y}_i\right) + \vec{U}\left(0, \phi_2\right) \otimes \left(\vec{p}_g - \vec{y}_i\right) \\ \vec{y}_i \leftarrow \vec{y}_i + \vec{v}_i \end{cases} \tag{11}$$

where each $\vec{U}\left(0, \phi_i\right), i = 1, 2$, is a vector of random numbers uniformly distributed in $\left[0, \phi_i\right]$ which is randomly generated at each iteration and for each particle, $\otimes$ is component-wise multiplication, and $\vec{p}_g$ is the global best location. Each component of $\vec{v}_i$ is kept within the range $[-V_{max}, V_{max}]$.

Algorithm 1: Hybrid CAE training algorithm

```
Initialize Swarm
for Γ_max generations do
    for Particle ⊂ Swarm do
        Decode particle to CAE weight matrix and evaluate its
          fitness function;
        Update ParticleBest;
        Update GlobalBest;
    end
    for Particle ⊂ Swarm do
        Update particle velocity and position;
        Decode particle to CAE weight matrix and apply one BP
          iteration to the Decoder;
        Update Decoder weights in the particle;
    end
    Compute f_μ and f_σ;
    Compute f_μ,10;
    Compute δ_μ as the increment of f_μ,10;
    if f_σ < ε_σ and δ_μ < ε_μ then
        Break;
    end
end
```

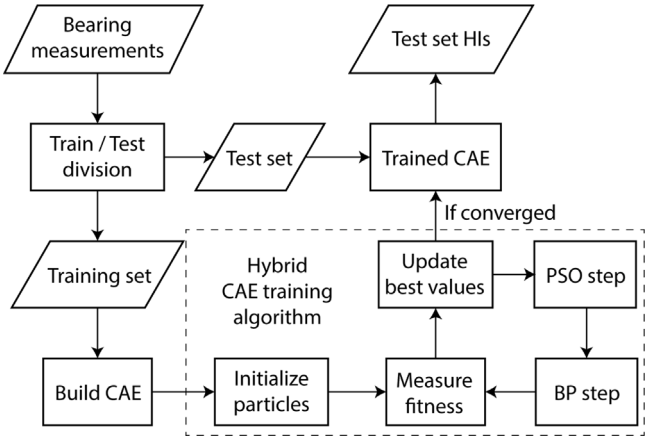


Fig. 3. Flowchart of the proposed method.

2.2.3. Hybrid training algorithm

PSO is able to perform an effective global search but under-performs BP while converging to the optimum (Ismail et al., 2013). To overcome this limitation, in this work, a novel hybrid CAE training algorithm (Algorithm 1) is developed where, at each generation, the particle velocity and position update in Eq. (11) to maximise the fitness function f (Eq. (9)) is followed by one iteration of BP applied to the Decoder weights to minimise the reconstruction error f_R (Eq. (9)). The coordinate vector of each particle in the search space is set as the CAE weight matrix, the Encoder weights are frozen, and one BP iteration is performed with a randomly selected batch of input signals, considering only the reconstruction error as the loss function. The implementation of one PSO and one BP iteration in tandem in each generation enables efficient and stable CAE training to achieve both objectives. The convergence criteria used in this work are if the swarm fitness standard deviation (f_σ) is less than ϵ_σ and if the mean swarm fitness (f_μ) averaged in the last 10 generations ($f_{\mu,10}$) improves less than ϵ_μ. ϵ_μ is set to a vary small value and ϵ_σ is selected experimentally. Once convergence is reached, the particle with the highest fitness value is selected as the best solution, yielding the trained CAE. The flowchart in Fig. 3 shows the data flow from measurement to HI construction, and Fig. 4 describes the proposed hybrid training algorithm in detail.

The effectiveness of the proposed method has been demonstrated in the two case studies presented in Sections 3 and 4.

3. Case study 1

3.1. Synthetic dataset

The synthetic dataset used in this case study has been developed in Yang et al. (2018). It includes 10 run-to-failure simulated sensor signals ($F = 10$) shown in Fig. 5(right) and obtained by non-linearly combining three factors representing the degradation trend $D(t)$, the environmental and operational behaviour $C(t)$, and the process noise $w(t)$, using trigonometric functions such as $s_1(t) = sin(0.5D(t) + 2C(t))$. The three factors are shown in Fig. 5(left) and are defined as:

$$D(t) = \alpha t \tag{12}$$

$$C(t) = \sin\left(\frac{1}{25}\pi t\right) + w_c \tag{13}$$

$$w(t) \sim N(\mu = 0, \sigma = 0.1) \tag{14}$$

where $\alpha = 0.025$ is the slope of the linear degradation factor, $t = 1, 2, \ldots, T$ is the time vector given in arbitrary time units ($T = 200$), and w_c represents a noise term inherent to the stochastic environmental and operational conditions and follows a normal distribution (N) with mean of 0 and standard deviation of 0.2:

$$w_c \sim N(\mu = 0, \sigma = 0.2) \tag{15}$$

The process noise, $w(t)$, which represents the uncertainty introduced by sensor inaccuracies, external disturbances, and other factors, follows a normal distribution with a mean of 0 and a standard deviation of 0.1.

In this case study, the time window for each CAE input, W, is experimentally set to 10. Hence, the CAE input shape is $W \times F = 10 \times 10$. Considering the input shape of the CAE in this case study, the weight matrix includes 338 parameters. Therefore, the PSO search space includes $K = 338$ dimensions.

3.2. Model selection and HI construction

This section compares first the proposed hybrid CAE training algorithm with the PSO and then the proposed CAE architecture with three alternatives with similar model sizes in terms of convergence speed and obtained global optimum. Finally, the CAE trained with the proposed hybrid algorithm is compared to the existing unsupervised HI construction approaches in terms of their ability to construct monotonic HIs and their computational burden during training.

The CAE is trained with the synthetic dataset proposed in Yang et al. (2018), normalised between 0 and 1. Since maintaining small weight values is conducive to successfully training deep NNs and preventing overfitting (Srivastava et al., 2014), the particles are constrained to the range $[-0.1, 0.1]$ at each dimension, implying that each weight parameter is kept within this range. Further, based on a trade-off between convergence speed and stability, V_{max} is set to 0.005, and ϕ_1 and ϕ_2 are set to their default value of 2. A swarm of 20 particles is randomly initialised in the 338-dimensional search space, and both PSO and hybrid (Algorithm 1) training algorithms are implemented. Setting the number of particles to 20 leads to an effective search, maintaining the diversity while keeping the computational burden at an acceptable level. Convergence is considered attained once the standard deviation of particle fitness values falls below $\epsilon_\sigma = 0.01$ and there is less than $\epsilon_\mu = 0.0001$ improvement in the mean swarm fitness averaged over 10 consecutive generations. The fitness value (f) corresponding to the best particle at each generation is plotted in Fig. 6. As a more robust comparison, considering their stochastic nature, the two algorithms are run 10 times with similar initial swarms, and the mean and standard deviations (stds) of the fitness, monotonicity score, and reconstruction error (f, f_M, and f_R in Eq. (9)) corresponding to the best particle in

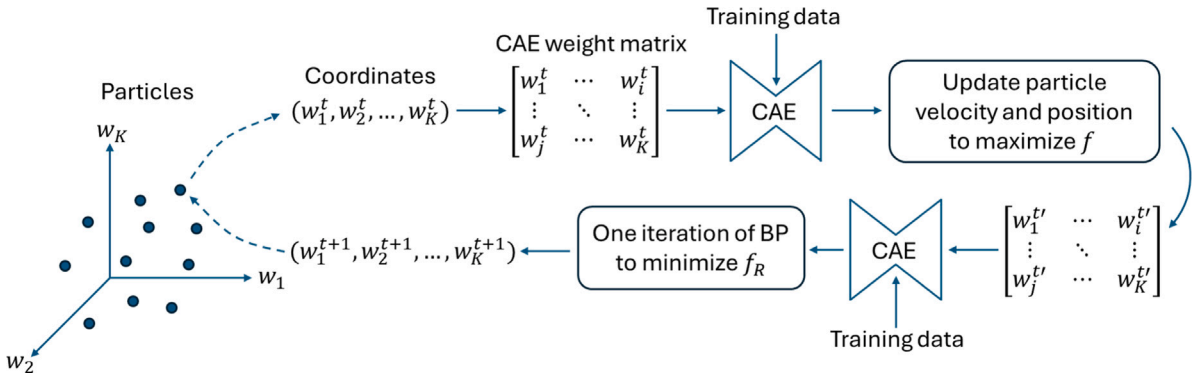


Fig. 4. Hybrid CAE training algorithm.

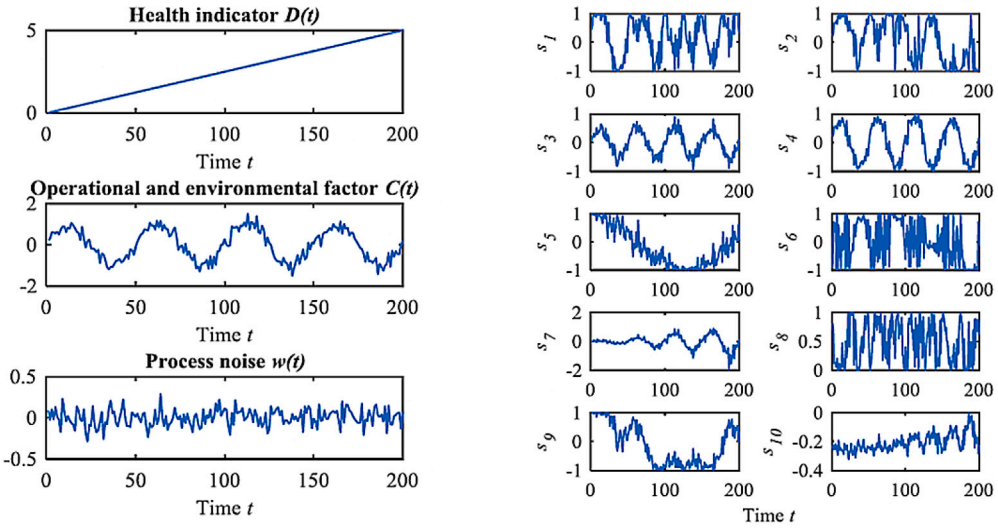


Fig. 5. (Left) Three factors contributing to the generation of the synthetic dataset. (Right) 10 simulated sensor signals (Yang et al., 2018).

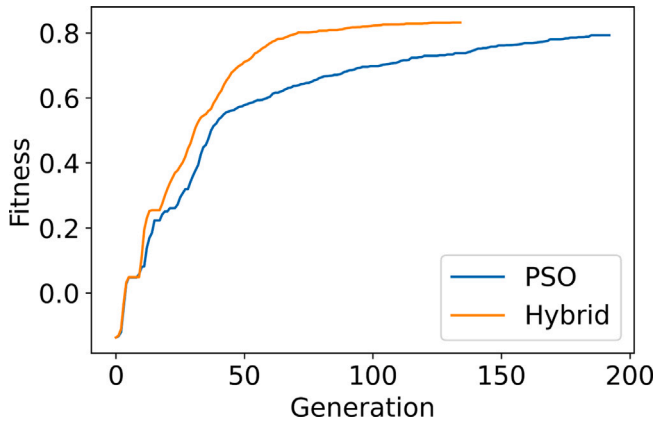


Fig. 6. Fitness (f) of the best particle in each generation until convergence.

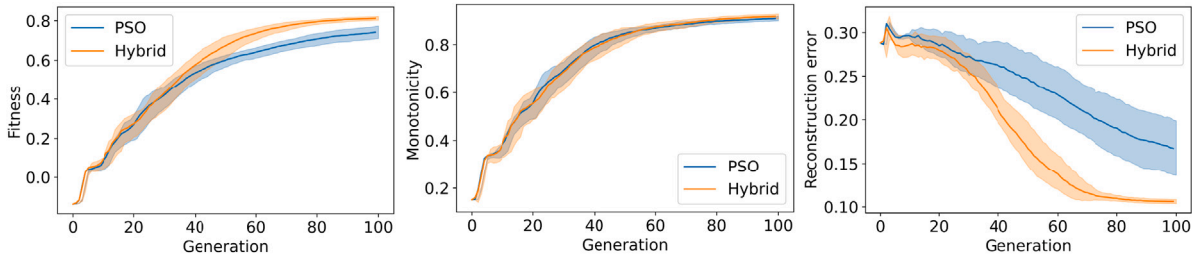
each generation are plotted in Fig. 7. The hybrid algorithm converges faster and to a higher f. The maximisation of f_M is similar in the two algorithms, and the improved f maximisation is mostly due to the significantly faster minimisation of f_R achieved by the hybrid algorithm.

The proposed CAE architecture is compared with three alternative architectures with the commonly used kernel sizes of 3×3, 5×5, and 7×7:

- Alternative CAE 1 — layer1: 3×3, layer2: 3×3
- Alternative CAE 2 — layer1: 5×5, layer2: 5×5
- Alternative CAE 3 — layer1: 7×7, layer2: 3×3

followed by Global Average Pooling (Lin et al., 2013), and a symmetric decoder. The remaining hyperparameters are similar to the proposed CAE architecture. The CAEs are trained by the hybrid algorithm until convergence, and the fitness value corresponding to the best particle in each generation is plotted in Fig. 8 for the four architectures. The proposed architecture outperforms in terms of both the speed of convergence and the attained optimum.

When adopting the proposed CAE architecture and the hybrid training algorithm, convergence is reached within 135 generations with runtime ≈ 6 min (CPU: 11th Gen Intel(R) Core(TM) i7-1185G7 — RAM: 16.0 GB) when run sequentially. However, running the hybrid training algorithm in parallel for the 20 particles leads to a runtime of ≈ 20 s. The constructed HI is shown in Fig. 9 and has monotonicity scores of $\rho_{\text{MK}} = 19.2417$ and $\tau_{\text{MK}} = 0.9367$. This compares favourably with the methods proposed in Yang et al. (2018) and Milani et al. (2020), which yield less monotonic HIs, with lower ρ_{MK} scores of 15.7612 and 15.5647, respectively, and considerably higher runtime. For example, the co-evolutionary optimisation framework proposed in Milani et al. (2020) has a runtime of more than 10 h on similar hardware. Therefore, it is around 100 times more computationally expensive.



(a) Fitness (f) (b) Monotonicity score (f_M) (c) Reconstruction error (f_R)

Fig. 7. *mean* $\pm$ *std* of fitness, monotonicity score, and reconstruction error of the best particle in each generation.

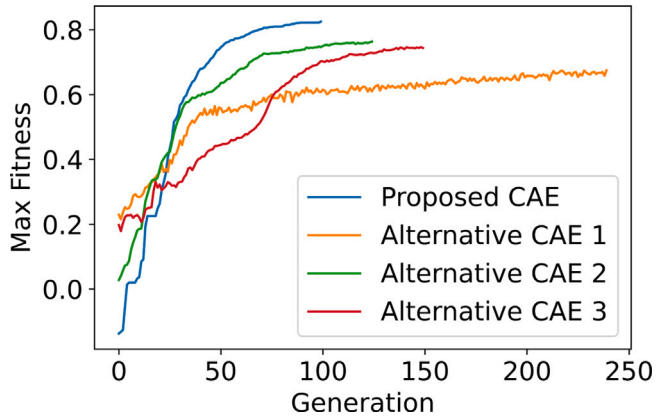


Fig. 8. Fitness (f) of the best particle in each generation until convergence for different CAE architectures.

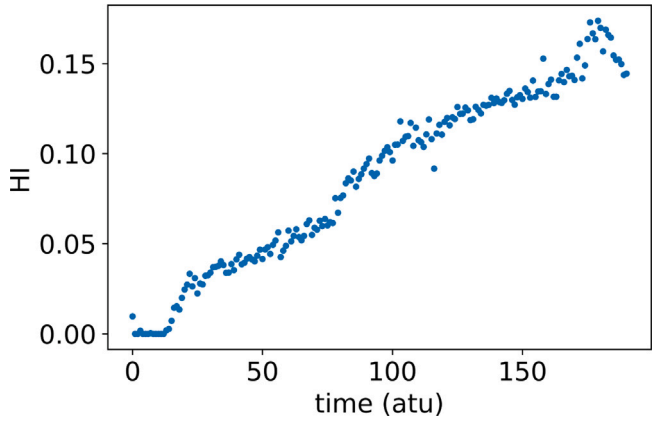


Fig. 9. HI corresponding to the proposed CAE and the hybrid training algorithm.

3.3. Analysis of the degradation trend

In this experiment, several alternative degradation trends are assumed for the synthetic dataset:

1. no degradation

$$D_1(t) = 0 \tag{16}$$

2. exponential degradation trend:

$$D_2(t) = \frac{1000^{\frac{t}{T}} - 1}{T} \tag{17}$$

3. linear degradation trend with varying slopes:

$$D_3(t) = \alpha t, \quad \alpha = 0.01, 0.015, 0.02, 0.025 \tag{18}$$

where $T = 200$ in arbitrary time unit (atu). The other two factors, $C(t)$ and $w(t)$, are kept the same as in Experiment 1. The proposed method is then compared with a supervised CNN with an architecture equivalent to that of the CAE Encoder and trained using labels referring to a degradation trend that varies linearly between 0 at the initial timestamp and 1 at failure, as commonly adopted in previous works (Table 1). The performance of the two algorithms is compared in terms of their ability to extract the true degradation trend out of the 10 sensor signals.

A robust method should construct a monotonically increasing HI if and only if the component undergoes progressive degradation. Otherwise, if the training set includes a considerable number of bearings that fail suddenly, with little to no gradual degradation, the trained model will underperform. Furthermore, the ability to extract the true degradation trend leads to a more accurate RUL prediction since the test set can include bearings that degrade under considerably different trends than those in the training set.

The HIs constructed with the CAE and the equivalent CNN are shown in Fig. 10 for the case of no degradation and in Fig. 11 for the case of exponential degradation. In the case of a linear degradation trend with varying slopes, the models are trained using the data related to the slope of 0.025 and tested on the remaining three. The obtained HIs are shown in Fig. 12. When the training data does not include a monotonic degradation trend (Fig. 10), the proposed method shows more robustness than the equivalent supervised CNN, which still constructs a highly monotonic HI despite the lack of degradation in the input signals. This is evidenced by the τ_{MK} values, which are equal to 0.2625 and 0.6068, respectively. When the true degradation trend is exponential, Fig. 11 shows that the proposed method is able to capture the true degradation trend, while the supervised CNN lacks adaptability as it still outputs a linear HI according to the labels used for its training. When the true degradation trend is linear, both the CAE and the CNN are able to capture it as evidenced by Fig. 12. Therefore, for a better comparison between the two approaches, a line is fitted to each of the obtained HIs, and its slope and the R^2 of the fit are compared in Table 2. The transformations made in both models and the normalisation of their input data do not allow the HI slopes to be equal to the true degradation trend slopes. Therefore, to allow a better comparison of the slopes, they are multiplied by a ratio so that the training data HI has a slope of $\alpha = 0.025$ and the slopes of the test HIs are compared with their corresponding true values. The results demonstrate that both models capture the true slope ratios to a good extent. However, the HIs obtained by the CNN are closer to the true degradation trend, considering their comparatively higher R^2 values. This is expected since the CNN is trained in a supervised way with labels corresponding to a linear degradation trend. However, in real-world applications, degradation rarely develops linearly, and this highlights the importance of the developed method.

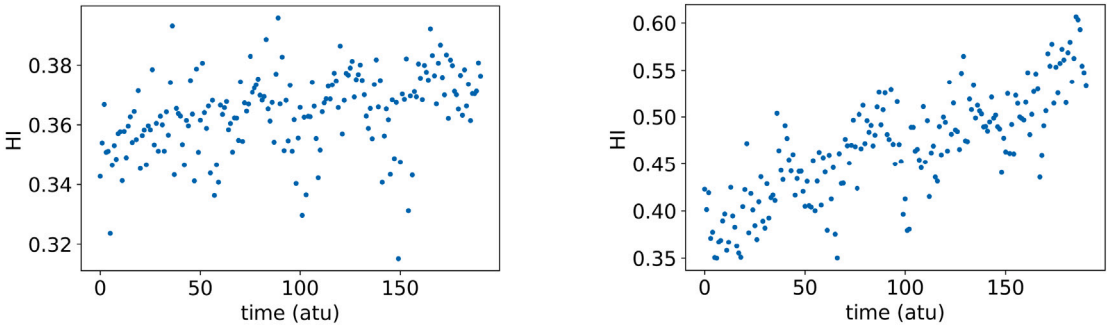


Fig. 10. No degradation — HI corresponding to (a) the proposed CAE trained by the hybrid algorithm and (b) the equivalent CNN.

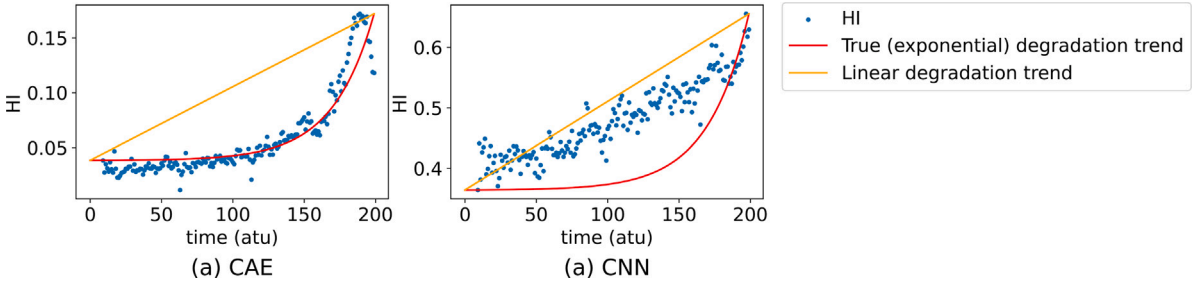


Fig. 11. Exponential degradation trend — HI corresponding to the (a) proposed CAE trained by the hybrid algorithm, and (b) the equivalent CNN.

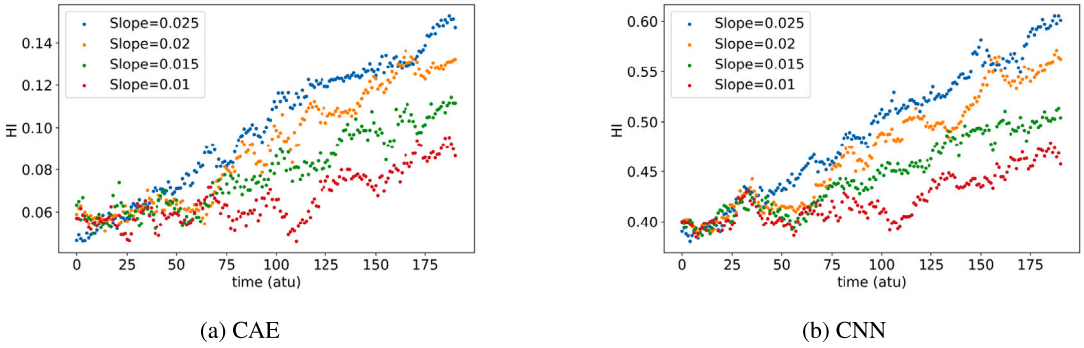


Fig. 12. Linear degradation with varying slopes — HI corresponding to (a) the proposed CAE trained by the hybrid algorithm and (b) the equivalent CNN.

Table 2
Linear degradation with varying slopes — Trend analysis.

	True slope	0.025	0.02	0.015	0.01
CAE	Predicted slope	0.0250	0.0215	0.0140	**0.0084**
	R^2	0.9769	0.9410	0.8954	0.6931
CNN	Predicted slope	0.0250	**0.0210**	0.0140	0.0080
	R^2	**0.9830**	**0.9474**	**0.9276**	**0.7352**

Table 3
PHM 2012 dataset — Operational conditions.

Operational condition	1	2	3
Rotational speed [rpm]	1800	1650	1500
Radial load [N]	4000	4200	5000
Number of tests	7	7	3

4. Case study 2

4.1. IEEE PHM 2012 prognostic challenge dataset

In this case study, the proposed method is applied to the PHM 2012 Challenge experimental dataset consisting of ball-bearing accelerated degradation test data obtained from the PRONOSTIA experimental platform developed by the FEMTO-ST institute (Nectoux et al., 2012). This dataset includes the run-to-failure data of seventeen bearings which have been tested under three different operational conditions, as described in Table 3. During the experiments, vibration signals were measured in both the horizontal (x-axis) and vertical (y-axis) directions using two accelerometers located on the bearing housing. Every 10 s, measurements were recorded for 0.1 s with a sampling frequency of 25.6 kHz, as described in Fig. 13, resulting in 2560 recordings in the time domain for each axis. A detailed description of the experimental setup and the experiments is provided in Nectoux et al. (2012).

The usual practice in DL-based methods is training the model with 70% of the available data and testing it with the remaining 30% (Gholamy et al., 2018). Accordingly, the proposed CAE is trained and tested with data from 12 and 5 bearings, respectively, as in Chen et al. (2021) and described in Table 4, where Bxy stands for bearing experiment y, with $y = 1, 2, \ldots 7$, under operational condition x, with $x = 1, 2, 3$.

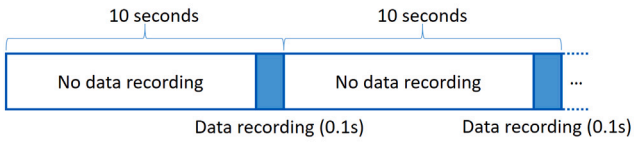


Fig. 13. PHM 2012 vibration data recording scheme.

Table 4
PHM 2012 dataset — Data division.

Operational condition	1	2	3
Training dataset	B12	B21	B31
	B14	B22	B32
	B15	B24	
	B16	B25	
	B17	B27	
Testing dataset	B11	B23	B33
	B13	B26	

4.2. Experiment

The time domain vibration signals are processed according to Fig. 14. To extract relevant features from the vibration signals, a Fast Fourier Transform (FFT) is applied to the vibration signals to produce their discrete frequency spectra. Only vibrations in the horizontal direction are considered in this study because when running the experiments, including the vibrations along both directions increased the computational burden significantly without yielding a considerable improvement in the results. Each set of 2560 time-domain recordings in the horizontal direction is transformed into the frequency spectrum using an FFT with a frequency range of 0 to 12 800 Hz. The spectrum is then down-sampled by dividing it into 64 bands, according to Ren et al. (2018), each with a frequency width of 200 Hz. This down-sampling significantly reduces the computational burden while maintaining acceptable performance. Signal representative features are obtained by considering the maximum spectral amplitude within each band Ren et al. (2018).

For each bearing, a 2-dimensional dataset of shape $T \times F$ is built, where T is the lifetime of the bearing expressed in $[10s]$ units, which is different for each run-to-failure experiment, and $F = 64$ is the number of features associated with each set of recordings. The time window W is set to 10 experimentally. Therefore, each bearing dataset of shape $T \times 64$ is transformed into a dataset with $T - W + 1 = T - 9$ samples, each of shape 10×64. The cost function is measured for each training sequence, and the mean fitness among all training sequences is assigned as the particle fitness value. Considering the input shape of the CAE in this case study, the weight matrix includes 1202 parameters. Therefore, the search space includes $K = 1202$ dimensions. The remaining hyperparameters are set to their values obtained in Section 3.2.

4.3. Results

The CAE is first trained with the training set and then tested using the remaining 30% of the dataset. To obtain consistent HIs for all bearings, the training loss function f' is used (Eq. (10)). For each test set bearing, the 0.1s vibration sensor recordings, collected every 10 s, are first transformed into the frequency domain, and 64 features are then extracted. Each sample of size $W \times F$, where $W = 10$ and $F = 64$, is used as input to the CAE to estimate the corresponding HI value. This process continues until the HI crosses the failure threshold ($HI = 1$).

The HIs constructed for the test set are shown in Fig. 15. For most test set bearings, the failure time predicted by the HI is very close to the true failure time. For bearings B11, B13, B23, and B33, the failure time is predicted to occur before what is observed in the experiments. Considering the HI trend for B26, it is expected to cross the failure threshold a few time stamps after the true failure time.

Bearings fail under different failure modes with various degradation trends, and this is captured by the HI trends, which show that bearings B11, B13, and B33 are characterised by a gradual degradation leading to failure. In contrast, bearings B23 and B26 undergo a rather sudden failure process. According to Jiang et al. (2022), the behaviour of the RMS of the vibration signals in the time domain can be used as an indicator of the bearing degradation process. In particular, a gradual or sudden increase of the RMS values when approaching the failure indicates that the degradation process is gradual or that the bearing undergoes a sudden failure, respectively. The behaviour of the normalised RMS values of the five bearings in the test set is shown in Fig. 16 and confirms the degradation trends of the constructed HIs.

As done in case study 1, the proposed unsupervised CAE is compared to a supervised CNN model, with an architecture equivalent to the CAE Encoder, and trained with linear HI labels, according to the most common approach adopted in the literature (Table 1). The HIs constructed by the CNN for the test set are shown in Fig. 17–(a). Unlike the CAE, the results show that the CNN cannot generate appropriate HIs. In most test set bearings, the failure time indicated by the HI crossing the threshold is predicted to occur significantly earlier than the true failure time, *i.e.* the final time stamp of the HI. This might be because the selected CNN architecture is too small for a supervised model trained by BP. This means that the proposed method outperforms the supervised CNN at an equivalent model size in terms of the number of parameters. Therefore, LeNet5 (LeCun et al., 1998), which is a popular CNN architecture, is selected for further comparison with CAE. This CNN architecture has been used in numerous previous works on bearing HI construction and fault detection (Yoo and Baek, 2018; Wen et al., 2018; Choudhary et al., 2021). Given the network input shape of 10×64 in this case study and using the original architecture proposed in LeCun et al. (1998), the LeNet5 features 38 189 weight parameters and is a significantly larger model compared to the proposed CAE, which features only 1202 parameters. The HIs constructed by LeNet5 for the test set are shown in Fig. 17–(b). In this case, none of the test set HIs crossed the failure threshold within the experiment duration.

In this case study, the key evaluation metrics used to compare the performance of the CAE, the equivalent CNN, and LeNet5 are the monotonicity score and the accuracy. The monotonicity score, τ_{MK}, defined in Eq. (4), quantifies the monotonicity of the HI up to the time it crosses the failure threshold or the final time stamp of the HI, $end = T - 9$, depending on whether the predicted HI crosses the threshold within the experiment duration or not. The accuracy, acc, quantifies how accurate the constructed HI is in predicting the true failure time, $t_{f,true}$, and is defined as:

$$acc = \begin{cases} \dfrac{t_{\mathrm{f,predicted}}}{t_{\mathrm{f,true}}} & \text{if} \quad HI(\mathrm{end}) \geq 1 \\ HI(end) & \text{if} \quad HI(end) < 1 \end{cases} \tag{19}$$

where $t_{f,predicted}$ is the failure time predicted by the HI. The accuracy metric is bounded within 0 and 1 and quantifies how close the coordinates of the last point of the HI are to $[end, 1]$.

The results in Table 5 show that in most cases, the HIs constructed by the CAE outperform the ones obtained by the two alternative methods. Furthermore, a qualitative comparison of the results in Figs. 15 and 17–(b) shows that the LeNet5-based HIs cannot differentiate between sudden failure and gradual degradation. Even though B23 and B26 undergo sudden failure, their HIs show degradation from the start of their operation. The rest of the test set bearings have similar behaviour, while it is expected that the degradation should be more gradual at the beginning and accelerate towards the end of life. A behaviour that is visible in the CAE-based HIs.

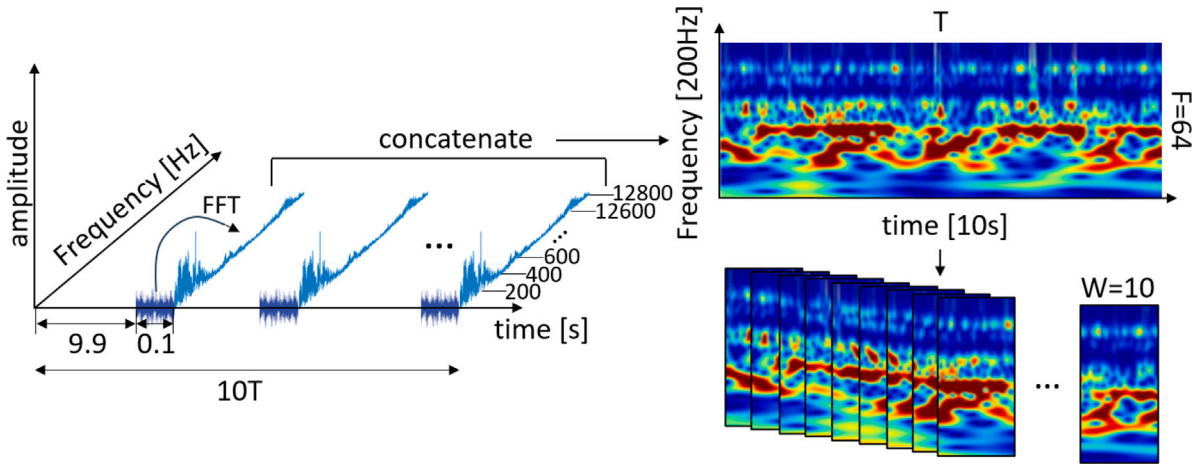


Fig. 14. Case study 2 - Data preparation.

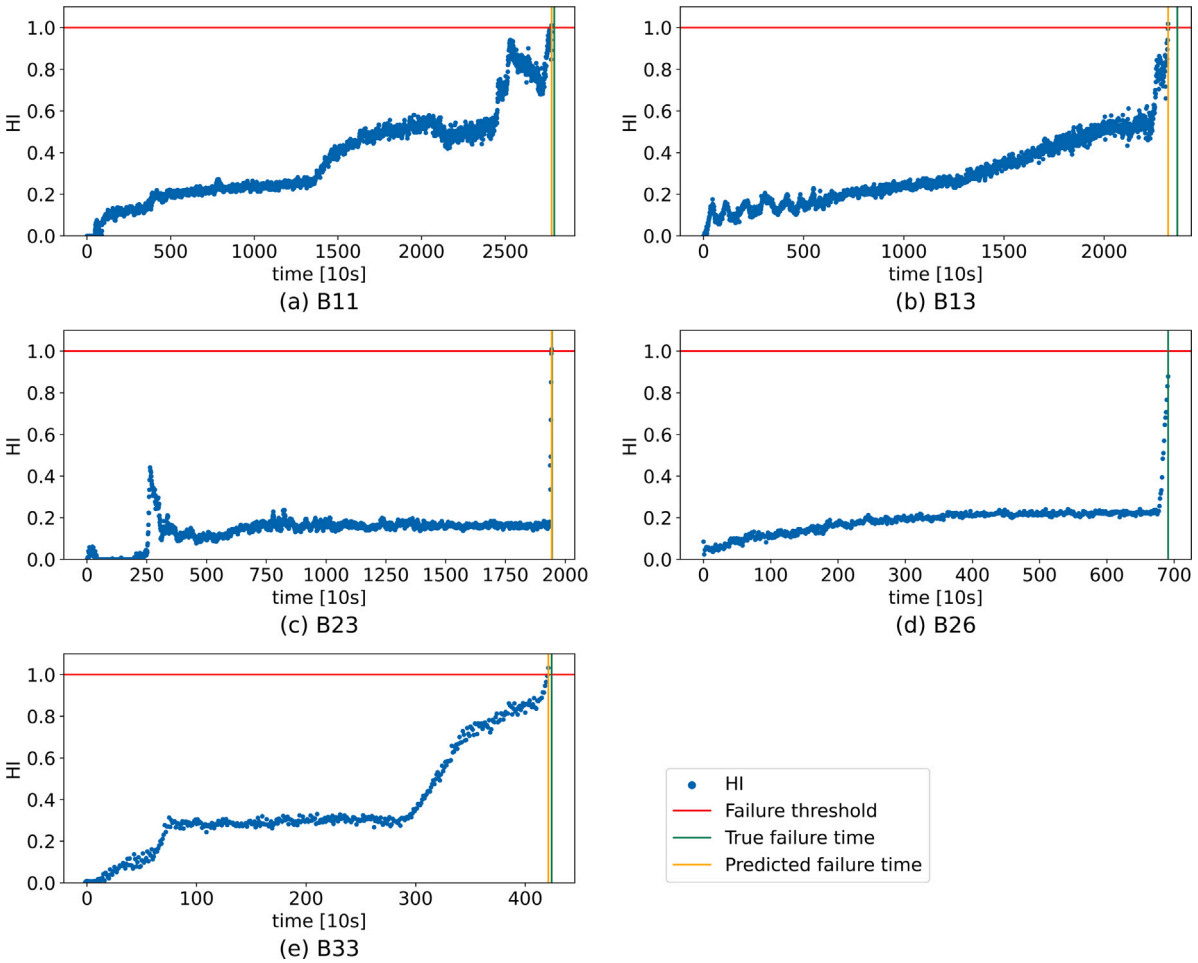


Fig. 15. CAE-based HIs for the test set of case study 2.

Accurate modelling of the HIs to predict their future values is an important step for RUL prediction and is out of the scope of this paper. Here, the aim is to compare the HIs in terms of their suitability for RUL prediction. Since equivalent CNN-based HIs underperform significantly, this comparison is only performed among the LeNet5- and CAE-based HIs and only those referring to bearings B11, B13, and B33. This is because they experience a gradual degradation, which is conducive to RUL prediction. In a sudden failure mechanism, as in B23 and B26, it is inherently difficult, if not impossible, to accurately predict the RUL well ahead of failure. Several studies propose a double exponential function (Eq. (20)) for modelling bearing HIs to predict their RUL, such as in Guo et al. (2017), Jin et al. (2016), Gebraeel et al. (2004). However, in

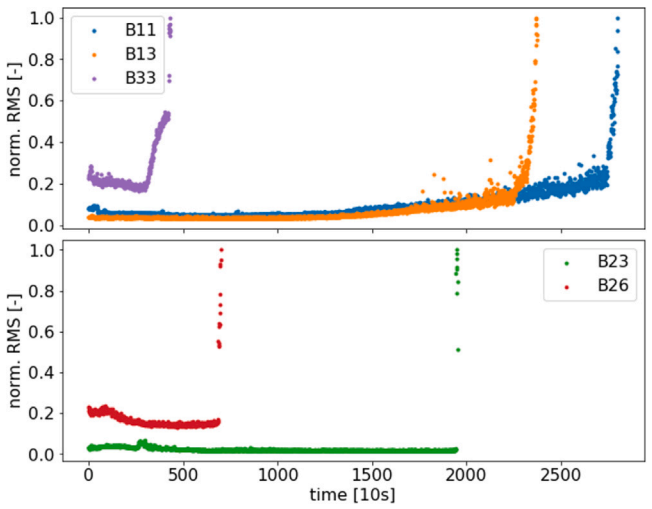


Fig. 16. Case study 2 - Test set RMS features.

Table 5
PHM 2012 dataset — Comparison of the three methods.

	CAE		Eq CNN		LeNet5	
	τ_{MK}	acc	τ_{MK}	acc	τ_{MK}	acc
B11	**0.8682**	**0.9939**	0.6729	0.2226	0.5654	0.7426
B13	**0.9006**	**0.9801**	0.7419	0.5173	0.8470	0.4932
B23	0.4023	**0.9985**	**0.6752**	0.3113	0.5203	0.6713
B26	0.7918	0.8788	0.7176	0.7777	**0.8733**	**0.9826**
B33	**0.8115**	**0.9906**	0.7718	0.7765	0.8028	0.9660

Table 6
Case study 2 - RUL prediction absolute errors.

	B11	B13	B33	Average
CAE	**6.89**	**19.01**	**4.59**	**10.16**
LeNet5	29.76	101.69	77.57	69.67

this experiment, it was observed that a 3rd order polynomial function (Eq. (21)) provides a better fit for some HIs. Therefore, for each HI, the model with the lowest fitting Mean Squared Error (MSE) between the true and modelled HI is selected for predicting the RUL. The models are fitted to the first 75% of the HI data, and from that point on, the RUL is predicted. The results are shown in Fig. 18.

$$m_{exp}(t) = ae^{bt} + ce^{dt} \tag{20}$$

$$m_{poly}(t) = at^3 + bt^2 + ct + d \tag{21}$$

The RUL prediction performance is evaluated using the error metric, Er, defined in Nectoux et al. (2012) as:

$$Er = 100 \times \frac{RUL_{true} - RUL_{predicted}}{RUL_{true}} \tag{22}$$

where RUL_{true} and $RUL_{predicted}$ are defined as the time intervals between the last timestamp of the fitting data and the true and predicted failure points, respectively. Table 6 compares the RUL prediction errors. The results show a better performance of the HIs based on the proposed method for RUL prediction with around 85% lower error on average, which further indicates that they can represent the true degradation trends of the bearings more accurately.

To evaluate the model latency in building the HIs in real-time, a timing analysis is performed. The generation of HIs for the five test set bearings is repeated 100 times, resulting in an average per-bearing prediction latency of $\approx$ 200 ms with a standard deviation of $\approx$ 15 ms. Considering that one new data point is generated every 10 s, this latency is satisfactory and allows the method to be applied in real-time.

To validate the performance of the HIs based on the proposed method in comparison with similar work proposed in the literature and to investigate their sensitivity to the selection of training and test bearings, leave-one-out cross-validation is performed. Mon (Eq. (2)) and Tre (Eq. (3)) metrics are used as HI performance metrics since they are predominantly adopted in the literature. An average of these metrics (Cr) is determined to allow a comparison with a range of other methods in the literature which build HIs for the PHM dataset:

$$Cr = \frac{Mon + Tre}{2} \tag{23}$$

Since Mon is highly sensitive to noise, the obtained HIs should be denoised for an effective evaluation (Duong et al., 2018; She and Jia, 2019; Li et al., 2019). In this work, Locally Weighted Regression (LOESS) (Cleveland and Devlin, 1988) is adopted as suggested in Duong et al. (2018), which is non-parametric. The results are reported in Table 7, which shows that the majority of the HIs built by the proposed method outperform the ones built by the four other methods in the literature. Bearings B21, B22, and B25 show noticeably lower Cr values compared to the rest. To investigate this observation, their HIs and time-domain RMS features are plotted in Fig. 19. Bearing B21 experiences a relatively lengthy stable degradation state characterised by a flat HI, leading to lower monotonicity. The RMS of bearing B22 shows a prominent decreasing trend shortly before failure. This behaviour is inconsistent with the common monotonic characteristic of bearing degradation and leads to an incorrect prediction of the failure time. However, the general trend of the built HI matches the behaviour of the RMS value. This observation highlights a potential limitation of the proposed method when the degradation trend is not monotonic. For B25, the RMS value indicates a lack of degradation until the very end of the bearing lifetime and has relatively low values throughout. Absolute values of the raw vibration data shown in Fig. 20 support this behaviour. Unlike the other experiments where failure was assumed to be reached and the experiment stopped when vibration acceleration levels exceeded 20 g, in B25, the maximum acceleration is around 10 g. The corresponding HI also reflects this observation, only indicating an acceleration towards failure near the very end of life.

5. Case study 3

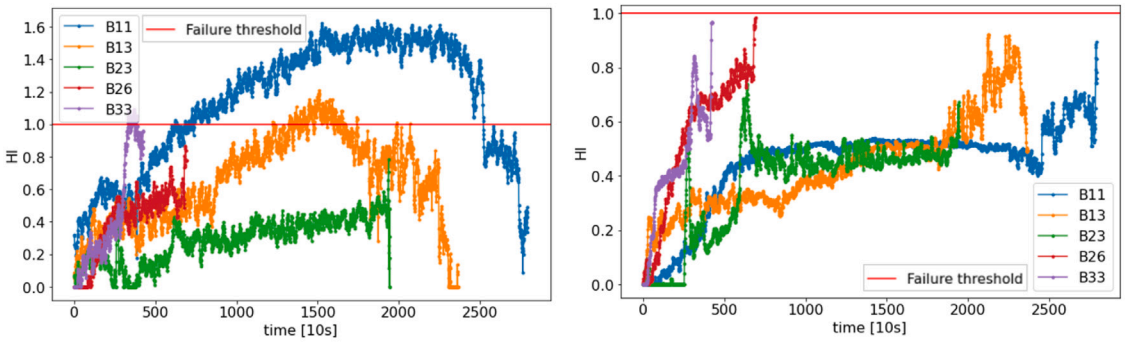
5.1. XJTU-SY bearing dataset

To further demonstrate the robustness of the developed method, it is applied to the XJTU-SY bearing dataset (Wang et al., 2020). This dataset includes run-to-failure vibration data of 15 rolling element bearings tested under three operating conditions reported in Table 8. To collect the vibration signals, two accelerometers are mounted on the bearing housing on horizontal and vertical axes. Every minute, measurements were recorded for 1.28 s with a sampling frequency of 25.6 kHz, resulting in 32 768 recorded data points in the time domain for each axis.

Following the 70/30 per cent train/test splitting rule, five bearings, *i.e.* the first two bearings from operational conditions 1 and 2 and the first bearing from operational condition 3 (C11, C12, C21, C22, C31), are set aside for testing, and the CAE is trained using the remaining 10 bearings.

5.2. Results

Similar to case study 2, at each recording period, time domain vibration signals in the horizontal axis are transformed to the frequency domain using an FFT and the spectrum is divided into 64 bands. Representative features of the signal are obtained by considering the maximum spectral amplitude within each band. The CAE training process detailed in case study 2 is repeated, and the test set HIs are shown in Fig. 21. For C12 and C31, the HI crosses the failure threshold



(a) Equivalent CNN-based HIs (b) LeNet5-based HIs

Fig. 17. Case study 2 - Test set HIs based on the Equivalent CNN and LeNet5.

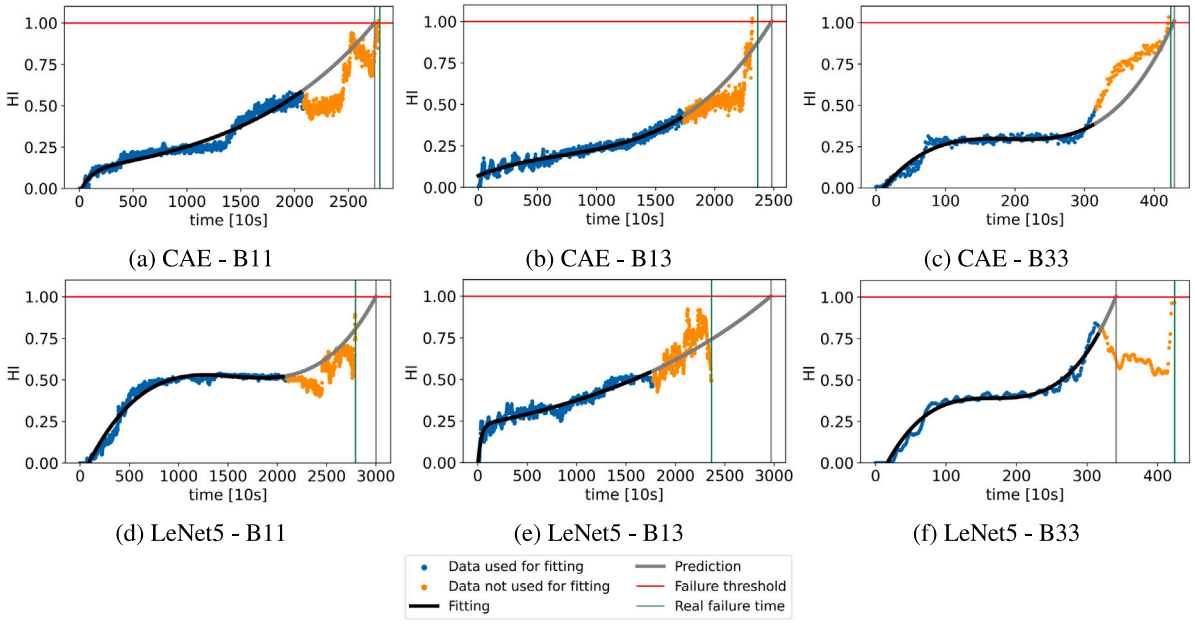


Fig. 18. Case study 2 - RUL prediction: comparison between the CAE (Top) and the LeNet5-based HIs (Bottom).

Table 7
PHM 2012 dataset — comparison of the performance of the proposed method and those available in the literature in terms of the Cr metric.

	Proposed method	Duong et al. (2018)	She and Jia (2019)	Wang et al. (2022)	Zhuang et al. (2024)
B11	**0.8910**	0.8872	–	0.2873	0.4651
B12	**0.9672**	–	–	0.5347	–
B13	**0.9651**	0.8314	–	–	–
B14	**0.9048**	0.8393	–	–	–
B15	**0.9157**	0.8195	–	–	–
B16	**0.9304**	0.9081	0.6331	–	–
B17	0.6772	**0.8952**	–	0.5220	–
B21	**0.5762**	–	–	0.4641	0.4641
B22	0.5334	**0.9654**	–	–	–
B23	**0.8567**	0.6639	–	–	–
B24	**0.9950**	0.8991	–	–	–
B25	0.5957	**0.9593**	–	–	–
B26	**0.9719**	0.7764	–	–	–
B27	**0.9865**	0.7049	–	–	–
B31	0.9617	–	–	–	–
B32	0.7254	**0.8730**	–	0.4134	0.4134
B33	**0.9639**	0.9260	–	–	–

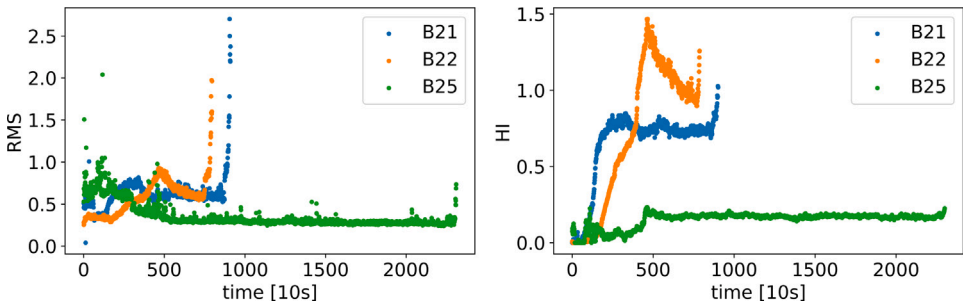


Fig. 19. Case study 2 - Underperforming bearings: RMS and HI plots.

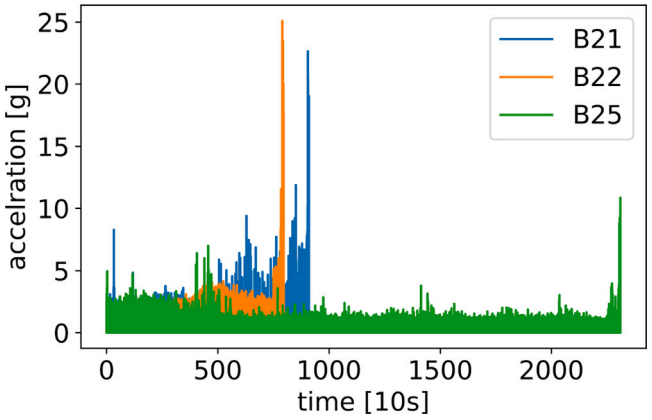


Fig. 20. Case study 2 - Underperforming bearings: Absolute values of raw vibration data.

Table 8
XJTU-SY dataset — Operational conditions.

Operational condition	1	2	3
Rotational speed [rpm]	2100	2250	2400
Radial load [kN]	12	11	10
Number of tests	5	5	5

just before the end of the experiment, *i.e.*, the real failure time, while for the other three bearings, the trend indicates that it will cross the threshold a few time steps after the true failure time. For bearings C21 and C31, the onset of degradation is significantly later than for the other three bearings. Therefore, for a better analysis of the behaviour of the HIs immediately before bearing failure, these are plotted for the last two hours before failure in Fig. 22. It can be seen that, except for C21, the other four bearings show a similar degradation trend leading to failure. Post-mortem analysis of the failed bearings revealed that these four bearings had experienced outer race failure, while the failing element in C21 was the inner race.

Subsequently, the equivalent CNN and the LeNet5 models are trained to construct the HI. Their test set HIs during the last two hours before failure are plotted in Fig. 23 and are compared with the CAE-based HIs in terms of the τ_{MK} and *acc* metrics in Table 9. It is important to note that the lower τ_{MK} in bearings C21 and C31, in comparison with the other three bearings, is due to their significantly later degradation onset, which results in a large portion of their HIs being flat. Furthermore, the negative τ_{MK} values for the equivalent CNN- and LeNet5-based C31 HIs are due to their slight decreasing trend until the degradation onset, as shown in Fig. 24. These results show a significant underperformance of the HIs based on these two models in terms of trend, monotonicity, and failure prediction accuracy. Therefore, the comparison in terms of the RUL prediction capability is not performed in this case study.

Table 9
XJTU-SY dataset — Comparison of the three methods.

	CAE		Eq CNN		LeNet5	
	τ_{MK}	*acc*	τ_{MK}	*acc*	τ_{MK}	*acc*
C11	**0.8379**	**0.9082**	0.6996	0.6754	0.5140	0.8509
C12	**0.8877**	**0.9803**	0.7679	0.9745	0.6873	0.9216
C21	0.5893	**0.9617**	0.7211	0.8058	**0.7766**	0.7997
C22	**0.8024**	**0.8382**	0.2332	0.7450	0.7182	0.5592
C31	**0.2431**	**0.9945**	−0.3197	0.9765	−0.4617	0.9759

6. Conclusions and future work

In recent years, numerous DL-based methods have been developed for bearing HI construction, which is a crucial step towards a successful RUL prediction. However, most of these methods are supervised and require prior knowledge of the component degradation trend. Since bearings degrade and fail under various failure modes with various degradation trends and speeds, the assumption of a single HI trend is unrealistic. In this work, a hybrid training algorithm is developed to train a CAE for unsupervised bearing HI construction, simultaneously maximising the HI monotonicity built in its middle layer and minimising the reconstruction error. It is shown that compared to PSO, this training algorithm converges faster and to a better global optimum. The application to three case studies shows that the proposed HI construction approach is up to 100 times less computationally expensive compared to an unsupervised DL method proposed in the literature, allowing its applicability to large datasets, such as the PHM 2012 dataset. Furthermore, the proposed method is able to build HIs that closely follow the true component degradation trend, outperforming indicators built by both equivalent and significantly larger supervised CNNs, considering their monotonicity, failure prediction accuracy, and RUL prediction capability. As a result, they achieve 85% lower RUL prediction errors on average. Furthermore, it is shown that they outperform those built by several other methods in the literature. Future work will include the development of advanced RUL prediction methods that consider the inherent uncertainty associated with the bearing degradation process and failure. It is expected that the application of these methods, along with the proposed unsupervised HI construction approach, can lead to significantly more accurate RUL predictions.

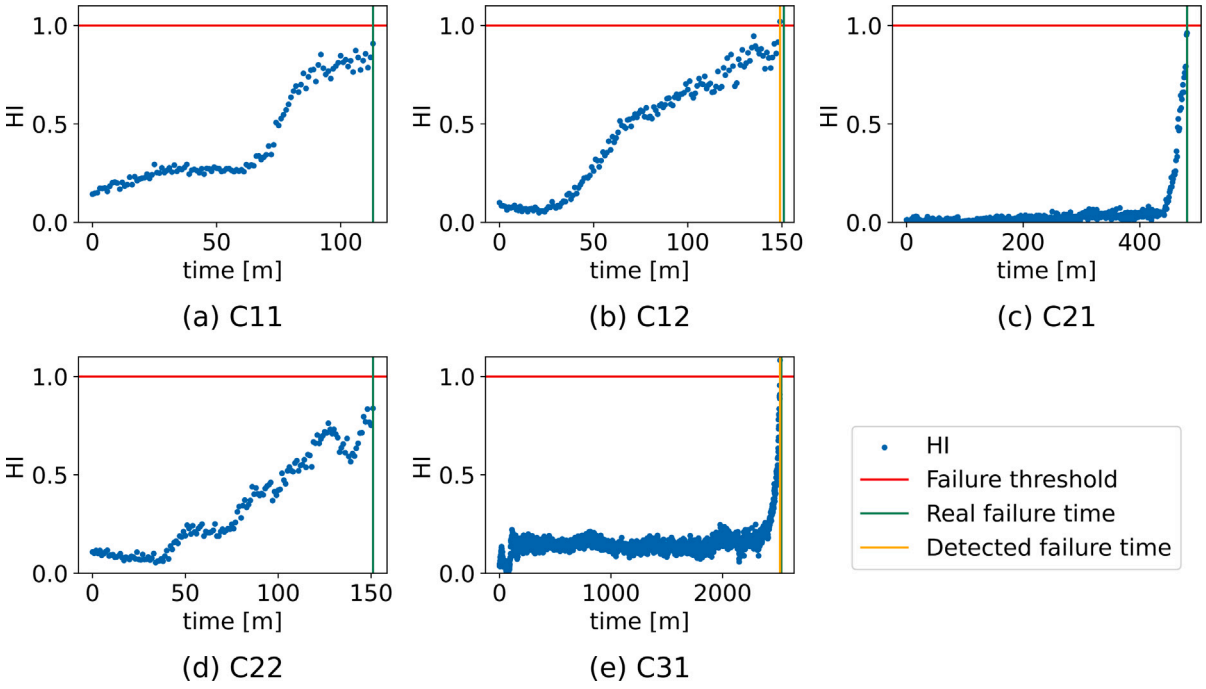


Fig. 21. CAE-based HIs for the test set of case study 3.

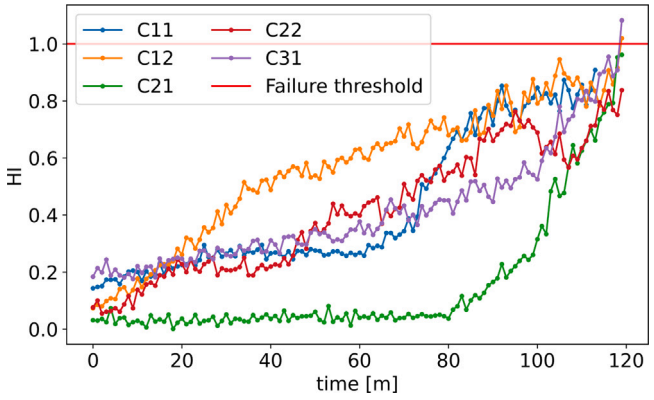


Fig. 22. CAE-based HIs for the test set of case study 3: Last two hours before failure.

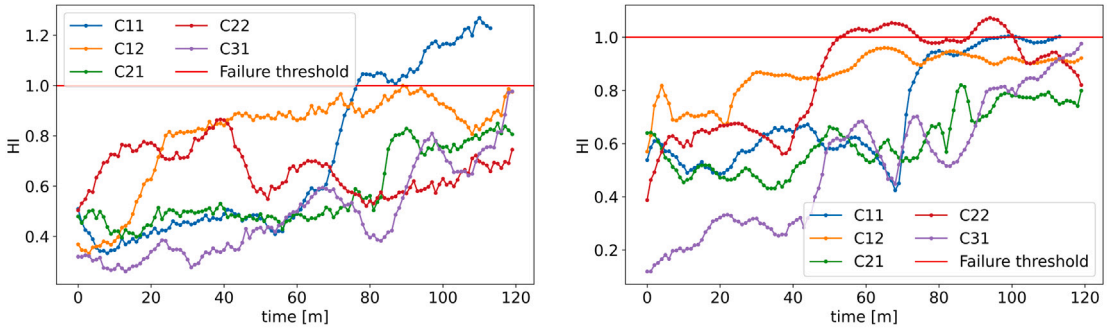


Fig. 23. Case study 3 - Test set HIs based on the Equivalent CNN and LeNet5: Last two hours before failure.

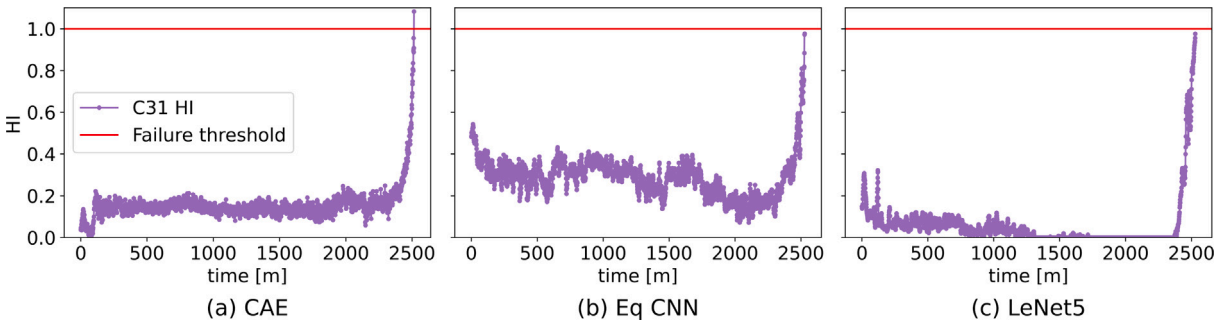


Fig. 24. Bearing B31 HIs based on the three models.

CRediT authorship contribution statement

Ali Eftekhari Milani: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Donatella Zappalá:** Writing – review & editing, Supervision, Project administration, Funding acquisition. **Simon J. Watson:** Writing – review & editing, Supervision, 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.

Data availability

Data will be made available on request.

References

Bianchini, Claudio, Immovilli, Fabio, Cocconcelli, Marco, Rubini, Riccardo, Bellini, Alberto, 2011. Fault detection of linear bearings in brushless AC linear motors by vibration analysis. IEEE Trans. Ind. Electron. 58 (5), 1684–1694. http://dx.doi.org/10.1109/TIE.2010.2098354, URL https://ieeexplore.ieee.org/document/5664782.

Cha, Young-Jin, Wang, Zilong, 2017. Unsupervised novelty detection–based structural damage localization using a density peaks-based fast clustering algorithm. Struct. Health Monit. (ISSN: 1741-3168) 17 (2), 313–324. http://dx.doi.org/10.1177/1475921717691260.

Chen, Dingliang, Qin, Yi, Wang, Yi, Zhou, Jianghong, 2021. Health indicator construction by quadratic function-based deep convolutional auto-encoder and its application into bearing RUL prediction. ISA Trans. 114, 44–56. http://dx.doi.org/10.1016/j.isatra.2020.12.052.

Choudhary, Anurag, Mian, Tauheed, Fatima, Shahab, 2021. Convolutional neural network based bearing fault diagnosis of rotating machine using thermal images. Measurement 176, 109196. http://dx.doi.org/10.1016/j.measurement.2021.109196.

Cleveland, William S., Devlin, Susan J., 1988. Locally weighted regression: An approach to regression analysis by local fitting. J. Amer. Statist. Assoc. (ISSN: 1537-274X) 83 (403), 596–610. http://dx.doi.org/10.1080/01621459.1988.10478639.

Ding, Peng, Jia, Minping, Yan, Xiaoan, 2021. Stationary subspaces-vector autoregressive with exogenous terms methodology for degradation trend estimation of rolling and slewing bearings. Mech. Syst. Signal Process. 150, 107293. http://dx.doi.org/10.1016/j.ymssp.2020.107293.

Duong, Bach Phi, Khan, Sheraz Ali, Shon, Dongkoo, Im, Kichang, Park, Jeongho, Lim, Dong-Sun, Jang, Byungtae, Kim, Jong-Myon, 2018. A reliable health indicator for fault prognosis of bearings. Sensors (ISSN: 1424-8220) 18 (11), 3740. http://dx.doi.org/10.3390/s18113740.

Gebraeel, N., Lawley, M., Liu, R., Parmeshwaran, V., 2004. Residual life predictions from vibration-based degradation signals: A neural network approach. IEEE Trans. Ind. Electron. (ISSN: 0278-0046) 51 (3), 694–700. http://dx.doi.org/10.1109/tie.2004.824875.

Gholamy, Afshin, Kreinovich, Vladik, Kosheleva, Olga, 2018. Why 70/30 or 80/20 relation between training and testing sets: A pedagogical explanation. Technical Report, University of Texas at El Paso, URL https://scholarworks.utep.edu/cs_techrep/1209/.

Guo, Liang, Lei, Yaguo, Li, Naipeng, Yan, Tao, Li, Ningbo, 2018. Machinery health indicator construction based on convolutional neural networks considering trend burr. Neurocomputing (ISSN: 0925-2312) 292, 142–150. http://dx.doi.org/10.1016/j.neucom.2018.02.083, URL https://www.sciencedirect.com/science/article/pii/S0925231218302583.

Guo, Liang, Li, Naipeng, Jia, Feng, Lei, Yaguo, Lin, Jing, 2017. A recurrent neural network based health indicator for remaining useful life prediction of bearings. Neurocomputing 240, 98–109. http://dx.doi.org/10.1016/j.neucom.2017.02.045.

Guo, Liang, Yu, Yaoxiang, Duan, Andongzhe, Gao, Hongli, Zhang, Jiangquan, 2022. An unsupervised feature learning based health indicator construction method for performance assessment of machines. Mech. Syst. Signal Process. 167, 108573. http://dx.doi.org/10.1016/j.ymssp.2021.108573.

Hinton, G.E., Salakhutdinov, R.R., 2006. Reducing the dimensionality of data with neural networks. Science 313 (5786), 504–507. http://dx.doi.org/10.1126/science.1127647, URL https://www.science.org/doi/abs/10.1126/science.1127647.

Ismail, A., Jeng, D.-S., Zhang, L.L., 2013. An optimised product-unit neural network with a novel PSO–BP hybrid training algorithm: Applications to load–deformation analysis of axially loaded piles. Eng. Appl. Artif. Intell. (ISSN: 0952-1976) 26 (10), 2305–2314. http://dx.doi.org/10.1016/j.engappai.2013.04.007.

Jiang, Changhong, Liu, Xinyu, Liu, Yizheng, Xie, Mujun, Liang, Chao, Wang, Qiming, 2022. A method for predicting the remaining life of rolling bearings based on multi-scale feature extraction and attention mechanism. Electronics 11 (21), 3616. http://dx.doi.org/10.3390/electronics11213616.

Jin, Xiaohang, Sun, Yi, Que, Zijun, Wang, Yu, Chow, Tommy W.S., 2016. Anomaly detection and fault prognosis for bearings. IEEE Trans. Instrum. Meas. (ISSN: 1557-9662) 65 (9), 2046–2054. http://dx.doi.org/10.1109/tim.2016.2570398.

Kipchirchir, E., Do, M.H., Njiri, J.G., Söffker, D., 2023. Prognostics-based adaptive control strategy for lifetime control of wind turbines. Wind Energy Sci. 8 (4), 575–588. http://dx.doi.org/10.5194/wes-8-575-2023, URL https://wes.copernicus.org/articles/8/575/2023/.

Kong, Ziqian, Tang, Baoping, Deng, Lei, Liu, Wenyi, Han, Yan, 2020. Condition monitoring of wind turbines based on spatio-temporal fusion of SCADA data by convolutional neural networks and gated recurrent units. Renew. Energy (ISSN: 0960-1481) 146, 760–768. http://dx.doi.org/10.1016/j.renene.2019.07.033.

LeCun, Yann, Bottou, Léon, Bengio, Yoshua, Haffner, Patrick, 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86 (11), 2278–2324, URL https://ieeexplore.ieee.org/document/726791.

Li, Xiang, Zhang, Wei, Ding, Qian, 2019. Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction. Reliab. Eng. Syst. Saf. 182, 208–218. http://dx.doi.org/10.1016/j.ress.2018.11.011.

Lin, Min, Chen, Qiang, Yan, Shuicheng, 2013. Network in network. http://dx.doi.org/10.48550/ARXIV.1312.4400, URL https://arxiv.org/abs/1312.4400.

Lu, Chao, Chen, Jie, Hong, Rongjing, Feng, Yang, Li, Yuanyuan, 2016. Degradation trend estimation of slewing bearing based on LSSVM model. Mech. Syst. Signal Process. 76–77, 353–366. http://dx.doi.org/10.1016/j.ymssp.2016.02.031.

Masci, Jonathan, Meier, Ueli, Cireşan, Dan, Schmidhuber, Jürgen, 2011. Stacked convolutional auto-encoders for hierarchical feature extraction. In: Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 52–59. http://dx.doi.org/10.1007/978-3-642-21735-7_7, URL.

Milani, Ali Eftekhari, Antonello, Federico, Baraldi, Piero, Zio, Enrico, 2020. A coevolutionary optimization approach with deep sparse autoencoder for the extraction of equipment degradation indicators. In: Proceedings of the 30th European Safety and Reliability Conference and 15th Probabilistic Safety Assessment and Management Conference. Research Publishing Services, http://dx.doi.org/10.3850/978-981-14-8593-0_5858-cd.

Moradi, Morteza, Broer, Agnes, Chiachío, Juan, Benedictus, Rinze, Loutas, Theodoros H., Zarouchas, Dimitrios, 2023. Intelligent health indicator construction for prognostics of composite structures utilizing a semi-supervised deep neural network and SHM data. Eng. Appl. Artif. Intell. (ISSN: 0952-1976) 117, 105502. http://dx.doi.org/10.1016/j.engappai.2022.105502.

Nectoux, Patrick, Gouriveau, Rafael, Medjaher, Kamal, Ramasso, Emmanuel, Chebel-Morello, Brigitte, Zerhouni, Noureddine, Varnier, Christophe, 2012. PRONOSTIA : An experimental platform for bearings accelerated degradation tests. In: IEEE International Conference on Prognostics and Health Management, PHM'12, vol. sur CD ROM, IEEE Catalog Number : CPF12PHM-CDR, Denver, Colorado, United States, pp. 1–8, URL https://hal.science/hal-00719503.

Pohlert, Thorsten, 2015. Non-parametric trend tests and change-point detection. http://dx.doi.org/10.13140/RG.2.1.2633.4243, URL https://www.researchgate.net/publication/274014742_trend_Non-Parametric_Trend_Tests_and_Change-Point_Detection_R_package_version_001?channel=doi&linkId=551298ec0cf268a4aaea93c9&showFulltext=true.

Poli, Riccardo, Kennedy, James, Blackwell, Tim, 2007. Particle swarm optimization. Swarm Intell. (ISSN: 1935-3820) 1, 33–57. http://dx.doi.org/10.1007/S11721-007-0002-0, URL https://link.springer.com/article/10.1007/s11721-007-0002-0,

Rai, Akhand, Upadhyay, S.H., 2016. A review on signal processing techniques utilized in the fault diagnosis of rolling element bearings. Tribol. Int. (ISSN: 0301-679X) 96, 289–306. http://dx.doi.org/10.1016/j.triboint.2015.12.037, URL https://www.sciencedirect.com/science/article/pii/S0301679X15006052.

Ren, Lei, Sun, Yaqiang, Wang, Hao, Zhang, Lin, 2018. Prediction of bearing remaining useful life with deep convolution neural network. IEEE Access 6, 13041–13049. http://dx.doi.org/10.1109/access.2018.2804930.

Rezamand, Milad, Kordestani, Mojtaba, Carriveau, Rupp, Ting, David S.-K., Orchard, Marcos E., Saif, Mehrdad, 2020. Critical wind turbine components prognostics: A comprehensive review. IEEE Trans. Instrum. Meas. 69 (12), 9306–9328. http://dx.doi.org/10.1109/TIM.2020.3030165, URL https://ieeexplore.ieee.org/document/9220115.

She, Daoming, Jia, Minping, 2019. Wear indicator construction of rolling bearings based on multi-channel deep convolutional neural network with exponentially decaying learning rate. Measurement 135, 368–375. http://dx.doi.org/10.1016/j.measurement.2018.11.040.

Sim, Jinwoo, Min, Jinhong, Kim, Seokgoo, Lee, Seok Woo, Choi, Joo-Ho, 2023. Construction of bearing health indicator under time-varying operating conditions based on Isolation Forest. Eng. Appl. Artif. Intell. (ISSN: 0952-1976) 126, 107058. http://dx.doi.org/10.1016/j.engappai.2023.107058.

Srivastava, Nitish, Hinton, Geoffrey, Krizhevsky, Alex, Sutskever, Ilya, Salakhutdinov, Ruslan, 2014. Dropout: A simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (1), 1929–1958, URL https://dl.acm.org/doi/10.5555/2627435.2670313.

Suresh, Abhijit, Harish, K.V., Radhika, N., 2015. Particle swarm optimization over back propagation neural network for length of stay prediction. Procedia Comput. Sci. (ISSN: 1877-0509) 46, 268–275. http://dx.doi.org/10.1016/j.procs.2015.02.020, URL https://www.sciencedirect.com/science/article/pii/S1877050915000848. Proceedings of the International Conference on Information and Communication Technologies, ICICT 2014, 3-5 December 2014 at Bolgatty Palace and Island Resort, Kochi, India.

Tao, Hongfeng, Qiu, Jier, Chen, Yiyang, Stojanovic, Vladimir, Cheng, Long, 2023. Unsupervised cross-domain rolling bearing fault diagnosis based on time-frequency information fusion. J. Franklin Inst. (ISSN: 0016-0032) 360 (2), 1454–1477. http://dx.doi.org/10.1016/j.jfranklin.2022.11.004.

Wang, Zilong, Cha, Young-Jin, 2020. Unsupervised deep learning approach using a deep auto-encoder with a one-class support vector machine to detect damage. Struct. Health Monit. (ISSN: 1741-3168) 20 (1), 406–425. http://dx.doi.org/10.1177/1475921720934051.

Wang, Biao, Lei, Yaguo, Li, Naipeng, Li, Ningbo, 2020. A hybrid prognostics approach for estimating remaining useful life of rolling element bearings. IEEE Trans. Reliab. (ISSN: 1558-1721) 69 (1), 401–412. http://dx.doi.org/10.1109/tr.2018.2882682.

Wang, Yu, Peng, Yizhen, Zi, Yanyang, Jin, Xiaohang, Tsui, Kwok-Leung, 2016. A two-stage data-driven-based prognostic approach for bearing degradation problem. IEEE Trans. Ind. Inform. 12 (3), 924–932. http://dx.doi.org/10.1109/tii.2016.2535368.

Wang, Zhijian, Ta, Yuntian, Cai, Wenan, Li, Yanfeng, 2023. Research on a remaining useful life prediction method for degradation angle identification two-stage degradation process. Mech. Syst. Signal Process. 184, 109747. http://dx.doi.org/10.1016/j.ymssp.2022.109747.

Wang, Huaqing, Zhang, Xisen, Guo, Xudong, Lin, Tianjiao, Song, Liuyang, 2022. Remaining useful life prediction of bearings based on multiple-feature fusion health indicator and weighted temporal convolution network. Meas. Sci. Technol. (ISSN: 1361-6501) 33 (10), 104003. http://dx.doi.org/10.1088/1361-6501/ac77d9.

Wen, Long, Li, Xinyu, Gao, Liang, Zhang, Yuyan, 2018. A new convolutional neural network-based data-driven fault diagnosis method. IEEE Trans. Ind. Electron. 65 (7), 5990–5998. http://dx.doi.org/10.1109/tie.2017.2774777.

Wen, Long, Yang, Guang, Hu, Longxin, Yang, Chunsheng, Feng, Ke, 2024. A new unsupervised health index estimation method for bearings early fault detection based on Gaussian mixture model. Eng. Appl. Artif. Intell. (ISSN: 0952-1976) 128, 107562. http://dx.doi.org/10.1016/j.engappai.2023.107562.

Wu, Chunzhi, Feng, Fuzhou, Wu, Shoujun, Jiang, Pengcheng, Wang, Jie, 2019. A method for constructing rolling bearing lifetime health indicator based on multi-scale convolutional neural networks. J. Braz. Soc. Mech. Sci. Eng. 41 (11), http://dx.doi.org/10.1007/s40430-019-2010-6.

Xiang, Ling, Wang, Penghe, Yang, Xin, Hu, Aijun, Su, Hao, 2021. Fault detection of wind turbine based on SCADA data analysis using CNN and LSTM with attention mechanism. Measurement (ISSN: 0263-2241) 175, 109094. http://dx.doi.org/10.1016/j.measurement.2021.109094.

Yaghini, Masoud, Khoshraftar, Mohammad M., Fallahi, Mehdi, 2013. A hybrid algorithm for artificial neural network training. Eng. Appl. Artif. Intell. (ISSN: 0952-1976) 26 (1), 293–301. http://dx.doi.org/10.1016/j.engappai.2012.01.023.

Yang, Zhe, Baraldi, Piero, Zio, Enrico, 2018. Automatic extraction of a health indicator from vibrational data by sparse autoencoders. In: 2018 3rd International Conference on System Reliability and Safety. ICSRS, IEEE, http://dx.doi.org/10.1109/icsrs.2018.8688720.

Yang, Zhe, Baraldi, Piero, Zio, Enrico, 2022. A method for fault detection in multi-component systems based on sparse autoencoder-based deep neural networks. Reliab. Eng. Syst. Saf. 220, 108278. http://dx.doi.org/10.1016/j.ress.2021.108278.

Yoo, Youngji, Baek, Jun-Geol, 2018. A novel image feature for the remaining useful lifetime prediction of bearings based on continuous wavelet transform and convolutional neural network. Appl. Sci. (ISSN: 2076-3417) 8 (7), http://dx.doi.org/10.3390/app8071102, URL https://www.mdpi.com/2076-3417/8/7/1102.

Yuan, Jing, Wang, Yu, Peng, Yizhen, Wei, Chenjun, 2017. Weak fault detection and health degradation monitoring using customized standard multiwavelets. Mech. Syst. Signal Process. 94, 384–399. http://dx.doi.org/10.1016/j.ymssp.2017.03.005.

Zhang, Yong, Sun, Jiahua, Zhang, Jing, Shen, Haoran, She, Yingying, Chang, Yang, 2023. Health state assessment of bearing with feature enhancement and prediction error compensation strategy. Mech. Syst. Signal Process. 182, 109573. http://dx.doi.org/10.1016/j.ymssp.2022.109573.

Zhang, Jing-Ru, Zhang, Jun, Lok, Tat-Ming, Lyu, Michael R., 2007. A hybrid particle swarm optimization–back-propagation algorithm for feedforward neural network training. Appl. Math. Comput. (ISSN: 0096-3003) 185 (2), 1026–1037. http://dx.doi.org/10.1016/j.amc.2006.07.025.

Zhou, Yang, Liu, Changfu, Yu, Xinli, Liu, Bo, Quan, Yu, 2022. Tool wear mechanism, monitoring and remaining useful life (RUL) technology based on big data: a review. SN Appl. Sci. 4 (8), http://dx.doi.org/10.1007/s42452-022-05114-9.

Zhuang, Jichao, Cao, Yudong, Ding, Yifei, Jia, Minping, Feng, Ke, 2024. An autoregressive model-based degradation trend prognosis considering health indicators with multiscale attention information. Eng. Appl. Artif. Intell. (ISSN: 0952-1976) 131, 107868. http://dx.doi.org/10.1016/j.engappai.2024.107868.