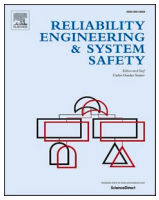
Reliability Engineering and System Safety 262 (2025) 111176

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

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



Degradation-Aware Remaining Useful Life Prediction of Industrial Robot via Multiscale Temporal Memory Transformer Framework

Zhan Gao [a], Chengjie Wang [a], Jun Wu [a,*], Yuanhang Wang [b], Weixiong Jiang [a], Tianjiao Dai [a]

[a] School of Naval Architecture and Ocean Engineering, Huazhong University of Science and Technology, Wuhan, China
[b] Sino-German College of Intelligent Manufacturing, Shenzhen Technology University, Shenzhen, China

ARTICLE INFO

Keywords:
Remaining useful life
State change identification
Transformer network
Multiscale temporal features
Industrial robot

ABSTRACT

Remaining useful life (RUL) prediction is of great importance to ensure stable operation of industrial robots (IRs). Deep learning-based methods have been proven effective in the RUL prediction tasks of IR. However, they are not effective in perceiving the state variation from a health state to a degradation state of IR and fail to reveal multi-term patterns of IR for RUL prediction. To address these challenges, a multiscale temporal memory Transformer framework is proposed to implement RUL prediction combined with state change identification. This proposed framework comprises a memory autoencoder Transformer network and a multiscale temporal Transformer network. The former Transformer network captures variation hidden in temporal information to detect the state change point, while the latter Transformer network is adopted to mine multi-term temporal dependencies for RUL prediction once state change point is identified. A self-built IR platform is constructed to validate our proposed method. Compared with the other advanced methods, the prediction results show that our method can locate the state change point in advance and achieve high-precision RUL prediction for IRs.

1. Introduction

Industrial robots (IRs) have gained widespread attention over the past two decades due to the availability of cutting-edge technologies and the requirement for high production. IRs can perform various tasks such as assembly, welding, and material handling [1–3]. Recently, the application of IRs has seen a surge in the industrial field, which helps humans improve work efficiency and reduce operational costs [4,5]. However, owing to the complex and harsh working conditions, IRs inevitably sustain a certain degree of damage. Hence, prognostics and health management (PHM) technology is introduced to perform condition monitoring and health maintenance of IRs. As a core part of PHM, remaining useful life (RUL) prediction can prevent deterioration and improve the reliability [6,7].

With the advancement of sensing technology, sensors are often used to monitor the health condition and acquire an amount of data [8–10]. Many data-driven methods have been proposed for RUL prediction. Existing data-driven methods are mainly divided into two categories: machine learning (ML)-based methods and deep learning (DL)-based methods. ML-based methods, such as Gaussian process [11], Kalman filter [12], support vector machine (SVM) [13], and XGBoost [14], can effectively construct mapping relationships between raw data and prediction output. For example, Haensch [15] et al. proposed an improved hidden Markov model (HMM) to implement prediction tasks, where the regularization step is integrated into the HMM to solve the covariance matrix. Ordóñez [16] et al. proposed a hybrid method that combined ARIMA with SVM. The ARIMA is built to evaluate the parameters of model, and the SVM is constructed to map the previous step results with the output RUL values. Cheng et al. constructed a dual-Wiener process to describe the degradation behavior of bearing, where Monte Carlo is adopted to update the parameters of model. ML-based methods have no requirement for a comprehensive physical understanding of the unit, and they can achieve quicker calculation and accurate prediction results.

Compared with ML-based methods, DL-based methods have a strong capacity to construct nonlinear relationships between raw data and outputs [17–19]. Many DL-based methods have been employed for RUL prediction of IR, and they have gained satisfactory performance. They can capture representative degradation features from monitoring signals without requiring prior knowledge [20,21]. Yang [22] et al. proposed a RUL prediction method based on long short-term memory (LSTM) for liquid crystal display (LCD) transfer robots, where domain

* Correspondence
E-mail address: wuj@hust.edu.cn (J. Wu).

https://doi.org/10.1016/j.ress.2025.111176
Received 7 September 2024; Received in revised form 1 March 2025; Accepted 21 April 2025
Available online 25 April 2025

generalization is used to reduce the individual differences among robots. Lizana [23] et.al combined random forest with variational autoencoder to implement degradation prediction for micro-robot. Taha [24] et al. constructed stacked LSTM to capture temporal information to achieve degradation prediction of robot arms. Xiao [25] et al. introduced Markov model-based temporal convolution networks (TCN) to construct the health index from the degradation behavior of IR and predict the life state. Jiang [26] et al. developed back propagation (BP) neural network to achieve the life prediction for reducer of IR, where maximum correlated Kurtosis deconvolution is used to reduce noise of measured signals. However, these methods fail to capture global temporal dependencies from measured data of IR, leading to higher uncertainty and significant lags in the prediction tasks.

Fortunately, Transformer networks have become increasingly prevalent in the prediction tasks. It has outstanding performance to capture global temporal dependencies based on multi-head attention mechanism. Some efforts have been made to enhance Transformer networks for RUL prediction. For example, Guo [27] et al. proposed an hourglass Transformer network for RUL prediction. The 1D-CNN and hourglass-shaped multiscale feature extraction is adopted to mine multiscale temporal features and integrate them for improving temporal representation. Shabani [28] et al. proposed an interactive multiscale Transformer for time-series prediction, where multiple scales framework and normalization scheme is introduced into model to improve the prediction performance. Chen [29] et al. developed a continuous-time Transformer for prediction tasks. The neural ordinary differential equations are integrated into multi-head attention to enhance temporal representation by modeling continuous dynamics. Ren [30] et al. proposed a multiscale Transformer enhanced by T^2-tensor, which can capture multiscale temporal features and implement light-weighting calculation for RUL prediction. Gao [31] et al. constructed a graph Transformer network for RUL prediction, where nonlinear slow-varying features are introduced into the network for enhancing the temporal dependencies. Li [32] et al. proposed an improved Transformer to predict the RUL of aircraft, where TCN is used as the encoder to mine positional relationships and feature decomposition is integrated into the decoder to mine trend features.

Although these DL-based methods have achieved satisfactory prediction performance, there are still two challenges that need to be solved to improve the RUL prediction performance. One is that an IR usually operates stably in the normal state and begins to degrade at an uncertain change point. However, these Transformer methods assume that IR degrades at the beginning of service time. They are unable to perceive the state change in advance, which can result in significant lag in the prediction tasks. The other is that the collected data of IR take the form of a long life-cycle sequence containing multi-term (long-term, medium-term, and short-term) latent degradation patterns. These methods fail to capture multi-term patterns for RUL prediction of IR. This can lead to a loss of temporal information and a decrease in prediction accuracy in the prediction tasks.

To tackle the above-mentioned challenges, a multiscale temporal memory Transformer framework is proposed in this paper for degradation-aware RUL prediction. The framework comprises two Transformer networks. First, a memory autoencoder Transformer network is built for signal reconstruction to identify state change point. Second, a multiscale temporal Transformer network is utilized to capture multi-term dependencies for enhancing prediction capacity beyond the state change point. To validate the effectiveness of our method, an IR platform with accelerated life testing is built. The main contributions are summarized as follows:

(1) A degradation-aware RUL prediction scheme is proposed to implement the RUL prediction for IRs by using multiscale temporal memory Transformer framework. It can not only timely identify health state and degradation state, but also achieve high-precision RUL prediction.
(2) A memory autoencoder Transformer network is designed to detect the change point from health state to degradation state. This network can mine dynamic temporal dependencies to enhance the temporal representation, which can detect state change of IRs in advance.
(3) A multiscale temporal Transformer network can construct and select the pathways for RUL prediction of IRs, which has strong capacity to capture multi-term dependencies to enhance temporal representation. It can predict future RUL values without requiring run-to-failure data.

The remainder of this paper is arranged as follows. Section 2 introduced inverse discrete Fourier transformer and multiscale patching operation. Section 3 describes the proposed methodology, which contains feature extraction, state change identification, and RUL prediction. Section 4 presents the experimental study of IR. Moreover, the ablation study, parameter analysis and comparison analysis are implemented in Section 5. Section 6 concludes this paper.

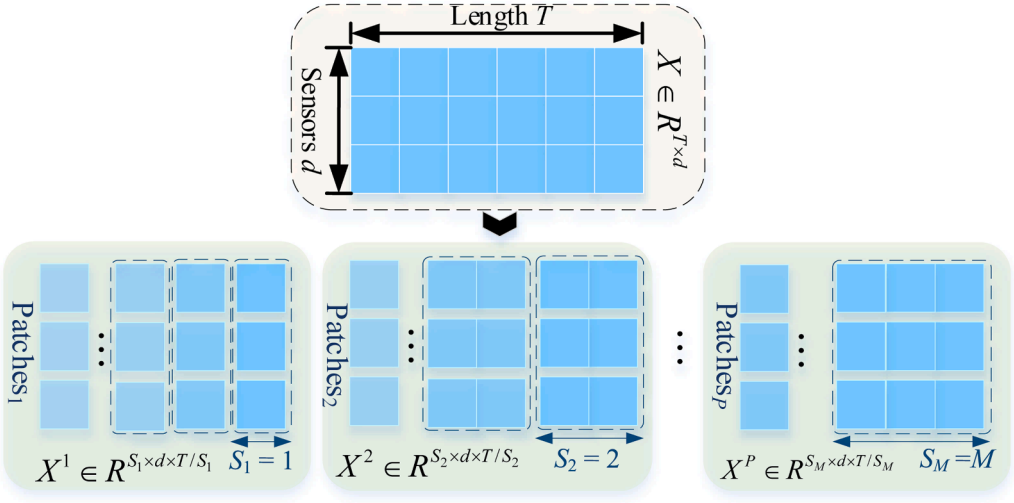


Fig. 1. The details of multiscale patching operation.

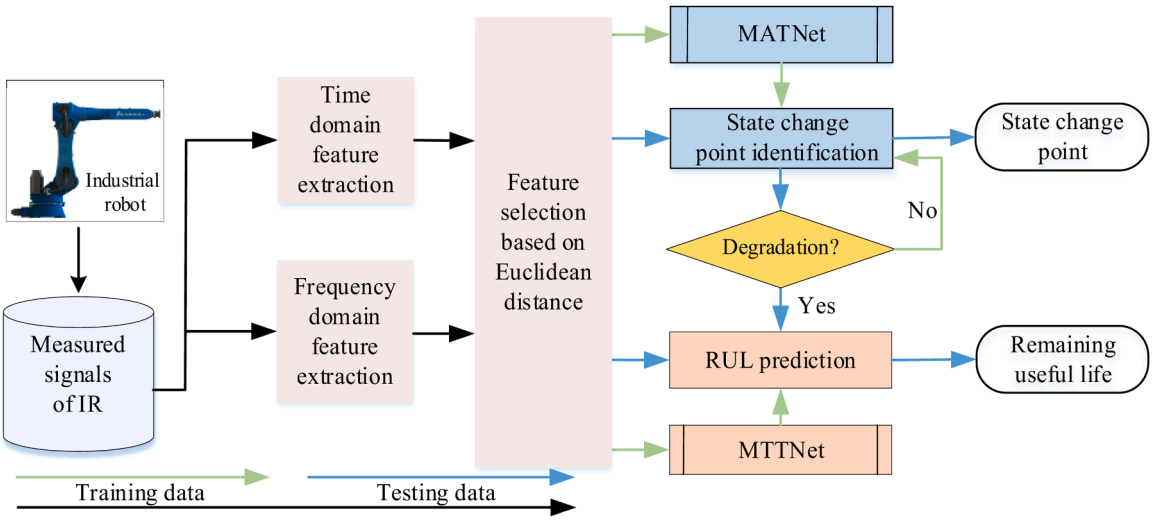


Fig. 2. Flowchart of multiscale temporal memory Transformer framework.

2. Theoretical background

2.1. Inverse discrete Fourier transform

Discrete Fourier transform (DFT) and inverse DFT (IDFT) form an invertible pair of transforms [33]. DFT can convert time-domain signal $x(n)$ back to frequency-domain signal $X(k)$, whose formula is expressed as:

$$X(k) = Re\{X(k)\} + jIm\{X(k)\} = \frac{1}{N}\sum_{k=0}^{N-1} x(n)\cdot\left[\cos\left(\frac{2\pi kn}{N}\right) + j\sin\left(\frac{2\pi kn}{N}\right)\right] \tag{1}$$

where N is the total sample number, j^2=-1, k denotes the frequency index, and n denotes the time index. The terms $Re\{\cdot\}$ and $Im\{\cdot\}$ represent the real and imaginary parts of $X(k)$, respectively. The amplitude and phase information are expressed as:

$$\begin{cases} A = \sqrt{Re\{X(k)\}^2 + Im\{X(k)\}^2} = \frac{1}{N}\sum_{k=0}^{N-1} x(n)\cdot\cos\left(\frac{2\pi kn}{N}\right) \\ \Phi = \tan^{-1}\left(\frac{Im\{X(k)\}}{Re\{X(k)\}}\right) = \frac{1}{N}\sum_{k=0}^{N-1} x(n)\cdot\sin\left(\frac{2\pi kn}{N}\right) \end{cases} \tag{2}$$

where A is the amplitude, and Φ is the phase.

IDFT can effectively provide a spectral analysis of the input sequence, revealing the amplitude and phase information of each frequency present transforms frequency domain $X(k)$ back into time domain sequence $x(n)$. It is expressed as:

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)\cdot e^{j\frac{2\pi kn}{N}} \tag{3}$$

IDFT is widely used in fields such as signal processing, communication systems, and data compression. It enables signal reconstruction from frequency components, allowing for data compression, noise reduction, and retention of key frequencies for analysis.

2.2. Multiscale patching operation

Multiscale patching operation [34] is used to divide the input data or an image into fixed-size patches, and then reconstruct these patches into a sequence of vectors to serve as the input for a Transformer network. The details of the patching operation are shown in Fig. 1.

For an input sequence $X \in R^{T\times d}$, where T is the time dimension and d is the feature dimension. The sequence can be divided into P patches with size S, and then vectors $v \in R^{S\times d}$ are obtained. The relationship between S and P is expressed as:

$$P = T/S \tag{4}$$

To obtain the multiscale patching sets, a set of patch sizes is defined as $\{S_1, S_2, \ldots, S_M\}$ based on M distinct patch size values. For patch size S_i, corresponding patches are obtained as $X^i \in R^{S_i\times d\times T/S_i}$. Moreover, the multiscale patching operation is used to obtain a patching set $X^S = (X^1, X^2, \ldots, X^p)$. This patching operation provides multiple views of the data, which can enhance the temporal representations based on different temporal resolutions.

3. Multiscale temporal memory Transformer framework

As shown in Fig. 2, the multiscale temporal memory Transformer framework is proposed to implement degradation-aware RUL prediction task of IR. It contains three parts: feature extraction and selection, memory autoencoder Transformer network (MATNet), and multiscale temporal Transformer network (MTTNet).

3.1. Feature extraction and selection

Feature extraction and selection is used to obtain degradation features from the measured signals. 7-dimensional time domain and 5-dimensional frequency domain features are extracted, and Euclidean distance is adopted to select these features. 7-dimensional time domain features are mean amplitude (tf_1), root mean square (tf_2), peak (tf_3), square mean root (tf_4), skewness (tf_5), kurtosis (tf_6), and standard deviation (tf_7). 5-dimensional frequency features are mean frequency (ff_1), variance frequency (ff_2), frequency center (ff_3), first moment (ff_4), and skewness frequency (ff_5). However, some features are ineffective at revealing degradation phenomena in IR, and high-dimensional data increase calculation burden. Therefore, Euclidean distance is used to select suitable features [35].

3.2. State change identification based on MATNet

The anomalous signals might emerge in the process from normal state to degradation state. These signals are typically weak and mixed with noise. To tackle this challenge, the MATNet is used to perceive the subtle change from measured signals and locate the state change point. It mainly contains three modules: Transformer encoder, enhanced memory module, and dense decoder. The structure of MATNet is shown

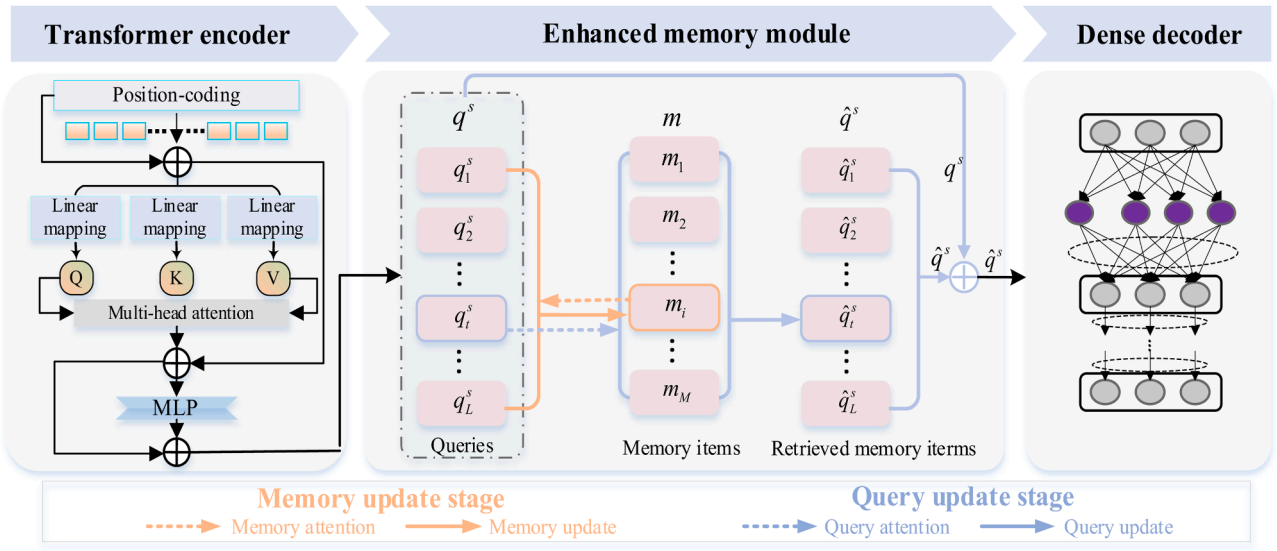


Fig. 3. Structure of MATNet.

in Fig. 3.

3.2.1. Transformer encoder

The Transformer encoder is used to embed measured signals into latent space. Let $X \in R^{T\times d}$ be the measured signals, where d represents the input dimension and T represents the number of time steps. The Transformer encoder adopts vanilla Transformer architecture to encode the measured signals X into long-term state. In the Transformer encoder, the measured signals X are augmented with positional information by using position coding [36]. These signals are converted into query vectors Q, key vectors K, and value vectors V. Moreover, these vectors are fed into the multi-head attention to capture long-term temporal dependencies.

The multi-head attention is expressed as:

$$\mathrm{MHA}(Q,K,V) = Concat\left(\{\mathrm{Attention}\ (Q,K,V)\}_{j=1}^{J}\right)W^{A} \tag{5}$$

where d_k is the transformed dimension. $W^A \epsilon R^{Jd_k\times d_m}$ denotes a trainable matrix, and d_m is the output dimension. The output of multi-head attention is fed into feedforward layer to transform the dimension and obtain the hidden state $q^s \in R^{T\times C}$:

$$q^s = ReLU(\mathrm{MHA} \otimes W_0 \oplus b_0)W_1 \oplus b_1 \tag{6}$$

where W_0 and W_1 represent the trainable parameters. b_0 and b_1 denote biases. C represents hidden dimension.

3.2.2. Dense decoder

The dense decoder is used to signal reconstruction from the hidden state. It adopts two linear layers to map the latent space into input space:

$$\widehat{X}^s = (q^s W_0 \oplus b_0)W_1 \oplus b_1 \tag{7}$$

where W_0 and W_1 are learnable parameters, and b_0 and b_1 are bias items. The reconstruction loss between Transformer encoder and dense decoder is expressed as:

$$L_{\mathrm{rec}} = \frac{1}{N}\sum_{s=1}^{N} \| X^s - \widehat{X}^s \|_2^2 \tag{8}$$

3.2.3. Enhanced memory module

Traditional autoencoder architectures usually adopt little reconstruction loss to train the model, while their reconstruction capacity is limited due to over-generalization and loss of temporal information [37, 38]. An enhanced memory module is developed by integrating memory mechanism to solve this problem. This mechanism can extract the hidden representation based on the prototypical normal patterns stored in the memory module, and limit the capacity of encoder to capture the anomaly features. The memory mechanism contains two phases, i.e., memory update phase, and query update phase.

In the memory update phase, the memory items $m_i \in R^C (i = 1,\ldots,M)$ are trained to contain prototypical normal patterns from input signals [37], where i represents the number of items. The incremental method is adopted to update the memory items, which contain prototypes of queries $q_t^s \in R^{T\times C}$ in the normal patterns at time-step t. This update is implemented by constructing query-conditioned memory attention $a_i^s \in R^T (i = 1,\ldots,M)$:

$$a_i^s = P\left(m_i \to q_t^s\right) = \frac{\exp\left(\langle m_i, q_t^s\rangle/\lambda\right)}{\sum_{p=1}^{T}\exp\left(\left\langle m_i, q_p^s\right\rangle/\lambda\right)} \tag{9}$$

where λ is a trainable temperature parameter used to adjust the concentration level of the distribution [39]. An update gate $\vartheta \in R^{M\times C}$ is used to train memory items in the normal patterns. It can control the extent to which the new patterns obtained by queries are integrated into current prototypical normal states of memory items. The update process is expressed as:

$$\vartheta = \sigma\left(\phi m_i + \varphi \sum_{t=1}^{T} a_i^s q_t^s\right) \tag{10}$$

$$m_i = (1-\vartheta)\otimes m_i + \vartheta \otimes \sum_{t=1}^{T} a_i^s q_t^s \tag{11}$$

where $\phi \in R^{C\times C}$ and $\varphi \in R^{C\times C}$ are trainable transformation matrices. σ and $\otimes$ are Sigmoid function and multiplication. The memory update phase is only implemented in the training stage.

In the query update phase, the updated queries $\widehat{q}_t^s$ is generated and it is input into the dense decoder. Softmax function is adopted to calculate the queries and memory items, and the memory-conditioned query-attention $b_i^s \in R^T (i = 1,\ldots,M)$ is acquired:

$$b_i^s = P\left(q_t^s \to m_i\right) = \frac{\exp\left(\langle q_t^s, m_i\rangle/\lambda\right)}{\sum_{j=1}^{M}\exp\left(\langle q_t^s, m_j\rangle/\lambda\right)} \tag{12}$$

Moreover, retrieved memory items $\widetilde{q}_t^s \in R^{T\times C}$ can be further acquired

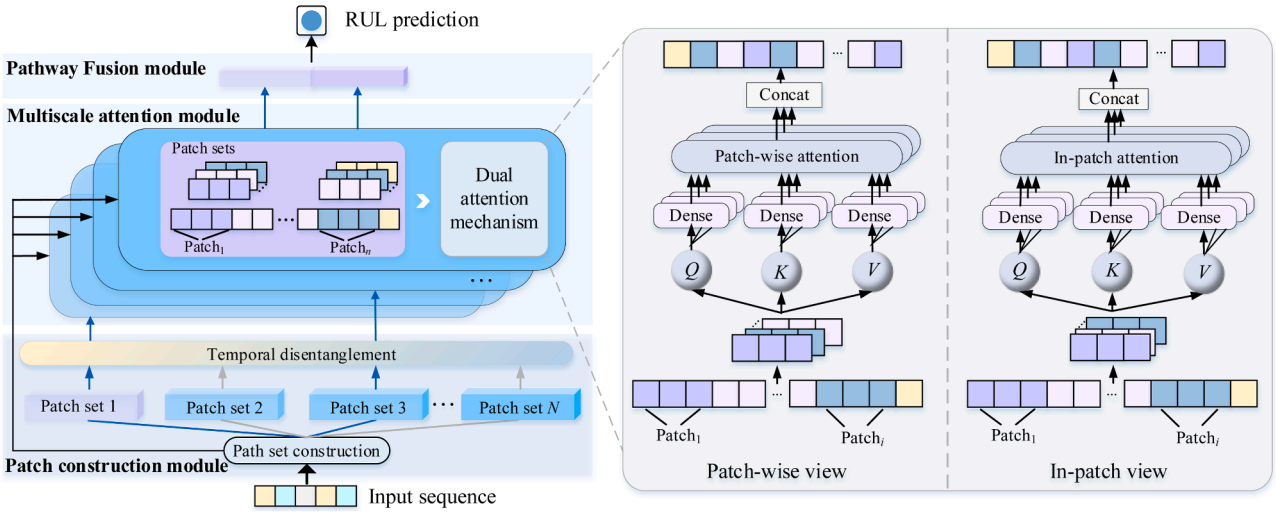


Fig. 4. Structure of MTTNet.

by integrating memory items m_i with memory-conditioned query-attention b_i^s:

$$\widetilde{q}_t^s = \sum_{i=1}^{M} m_i b_i^s \tag{13}$$

And, the updated queries $\widehat{q}_t^s \in R^{T\times 2C}$ are obtained by concatenating $\widetilde{q}_t^s$ with q_t^s. It is a new input for dense decoder. Moreover, the latent space and input space are used to establish the deviation-based state change identification criterion. The deviation of latent space (DLS) is defined as distance between queries and their nearest memory items $m_{t,nearest}^s$ at time-step t. Memory items contain prototypes of normal states, thereby the DLS in the degradation state is larger than those in the normal state. In addition, the deviation of input space (DIS) is defined as the distance between input signals and reconstructed signals at t time-step. DIS is multiplied by DLS to amplify the gap between normal and degradation states. The state score S_i is expressed as:

$$DLS = \| q_t^s - m_{t,nearest}^s \|_2^2 \tag{14}$$

$$DIS = \| X_t^s - \widehat{X}_t^s \|_2^2 \tag{15}$$

$$S_i = \mathrm{softmax}(DLS \otimes DIS) \tag{16}$$

The state scores in the degradation state are higher than those in the normal state. Then, a threshold δ is set to determine the current state of a sample point:

$$\mathscr{Y}_i = \begin{cases} D: \text{ Degradation state } S \geq \delta \\ N: \text{ normal state } S < \delta \end{cases} \tag{17}$$

When state scores of n_f sample points exceed the threshold successively, the point corresponding to the first sample point is identified as the state change point.

3.3. RUL prediction based on MTTNet

To implement RUL prediction task beyond the determined state change point, MTTNet is used to reveal the degradation behavior of IRs. As shown in Fig. 4, it consists of a patch construction module, a multiscale attention module, and a pathway fusion module.

3.3.1. Patch construction module

The input sequence $X \in R^{T\times d}$ is first segmented into P patches with different patch size S, and the patching set $X^S = (X^1, X^2, \ldots, X^P)$ is obtained. The diverse sequences may prefer different scales due to their specific temporal dynamics, and some scales may introduce redundant information. To capture multiscale temporal features, the patching construction module is built. This module constructs path sets and selects the optimal patch size for multiscale attention module.

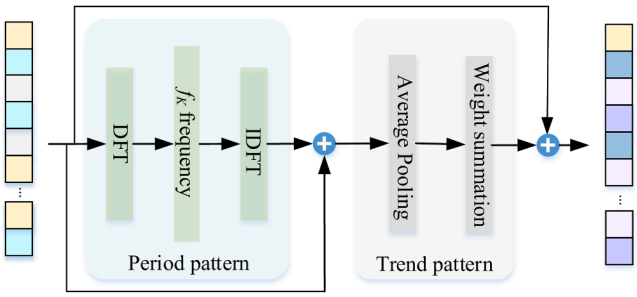


Fig. 5. Structure of temporal disentanglement.

The scale of a sequence may be influenced by the dynamical patterns, such as trend patterns and period patterns [40,41]. In the patch construction module, we choose temporal disentanglement to obtain the period and trend patterns from input sequence. The temporal disentanglement contains period decomposition and trend decomposition, shown in Fig. 5.

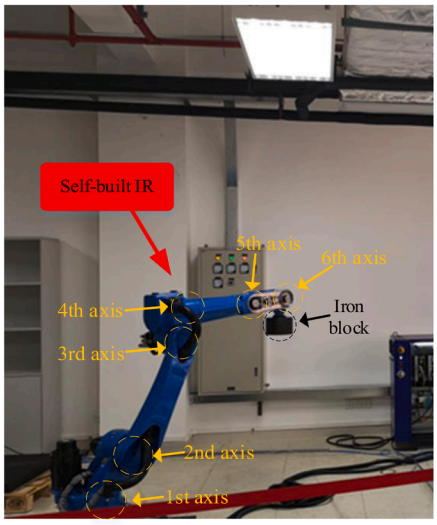
The period decomposition utilizes discrete Fourier transform (DFT) to decompose the input sequences X into Fourier basis $F = \{f_1, \ldots, f_K\}$:

$$\begin{cases} F = DFT(X) = Re(F) + jIm(F) \\ A = \sqrt{Re(F)^2 + Im(F)^2} \\ \Phi = \arctan\left(\dfrac{Im(F)}{Re(F)}\right) \end{cases} \tag{18}$$

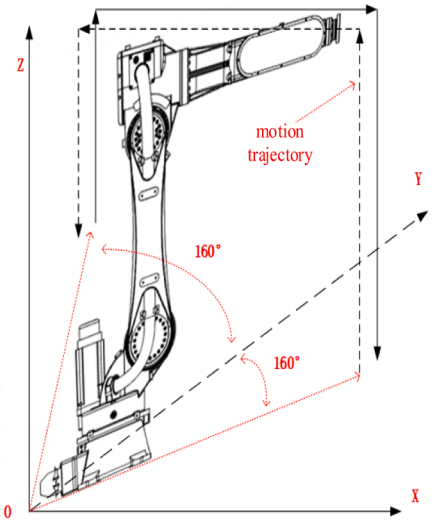
where A and Φ represent the amplitude and phase, respectively. j^2=-1, $Re(\cdot)$ and $Im(\cdot)$ are the real and imaginary part, respectively. The first f_K frequencies of Fourier basis F are retained to keep the sparsity of the frequency domain and remove redundant information. f_K is set as 3. Then, the inverse DFT (IDFT) is used to mine the periodic characteristics hidden in the frequency domain. It is constructed based on K_f of Fourier frequencies:

$$P = \mathrm{IDFT}(F_k, A, \Phi) \tag{19}$$

The trend decomposition is used to separate the periodic components from the non-periodic components. The trend patterns hidden in periodic components can be obtained:



(a) Structure of self-built IR platform



(b) Motion trajectory of the mechanical arms

(c) Reducer fault of 6th-axis

Fig. 6. Details of self-built IR platform.

$$X_{res} = X^s - P \tag{20}$$

To enhance the trend representation, average pooling operations with different kernel sizes are used to extract different trend characteristics from X_{res} . Softmax activation function is adopted to assign weights for different trend characteristics, and the high-level trend characteristics Tre are obtained. It is expressed as:

$$Tre = \text{Softmax}(X_{res}) \cdot (\text{Avgpool}(X_{res})_{k_1}, \ldots, \text{Avgpool}(X_{res})_{k_N}) \tag{21}$$

where $\text{Avgpool}(\cdot)_{k_i}$ denotes the pooling operation with the i-th kernel, and N denotes the number of the kernels.

Moreover, the pathway function R_T is used to generate pathway weights. It can determine which patch sizes should be chosen. To avoid repeatedly choosing some scales and ignoring others, the Gaussian noise term is introduced to keep the stability of the generated pathway weighting:

$$R_T = \text{Softmax}(W_r Tre + Mish(Dense(Tre) W_n)) \tag{22}$$

where W_r and $W_n \in R^{d \times M}$ represent the trainable parameters. d and M represent dimension and patch size number, respectively. $Mish(\cdot)$ denotes the Mish function. To keep the sparsity of the pathway, the top-K operation is implemented to retain former K pathways, while other weights of pathways are set as 0.

3.3.2. Multiscale attention module

The multiscale attention module is used to capture global and local temporal dependencies from the patch sets. It employs two views, i.e., the patch-wise view and the in-patch view, to mine both global and local features. In the patch-wise view, the relationships among different time steps within patches are established to obtain local features. Each patch $X^i \in R^{S \times d}$ in the patch set is embedded into patch-wise representation $X_i^{pw} \in R^{S \times d_m'}$, where d_m' denotes the embedding dimension. The linear transformation is applied for patch-wise representation to obtain key vectors K_{pw}^i and value vectors V_{pw}^i:

$$\begin{cases} K_{pw}^i = W_K X_i^{pw} \\ V_{pw}^i = W_V X_i^{pw} \end{cases} \tag{23}$$

where W_K and W_V denote linear transformation. An initialized query vectors $Q_{pw} \in R^{1 \times d_m}$ are defined to calculate the patch-wise attention:

$$Attn_{pw}^i(Q, K, V) = softmax\left(\frac{Q_{pw}^i \left(K_{pw}^i\right)^T}{\sqrt{d_m}}\right) V_{pw}^i \tag{24}$$

The attention results from all patches are integrated to form the final outputs of the patch-wise view:

$$Attn_{pw} = Concat\left(Attn_{pw}^1, \cdots, Attn_{pw}^M\right) \tag{25}$$

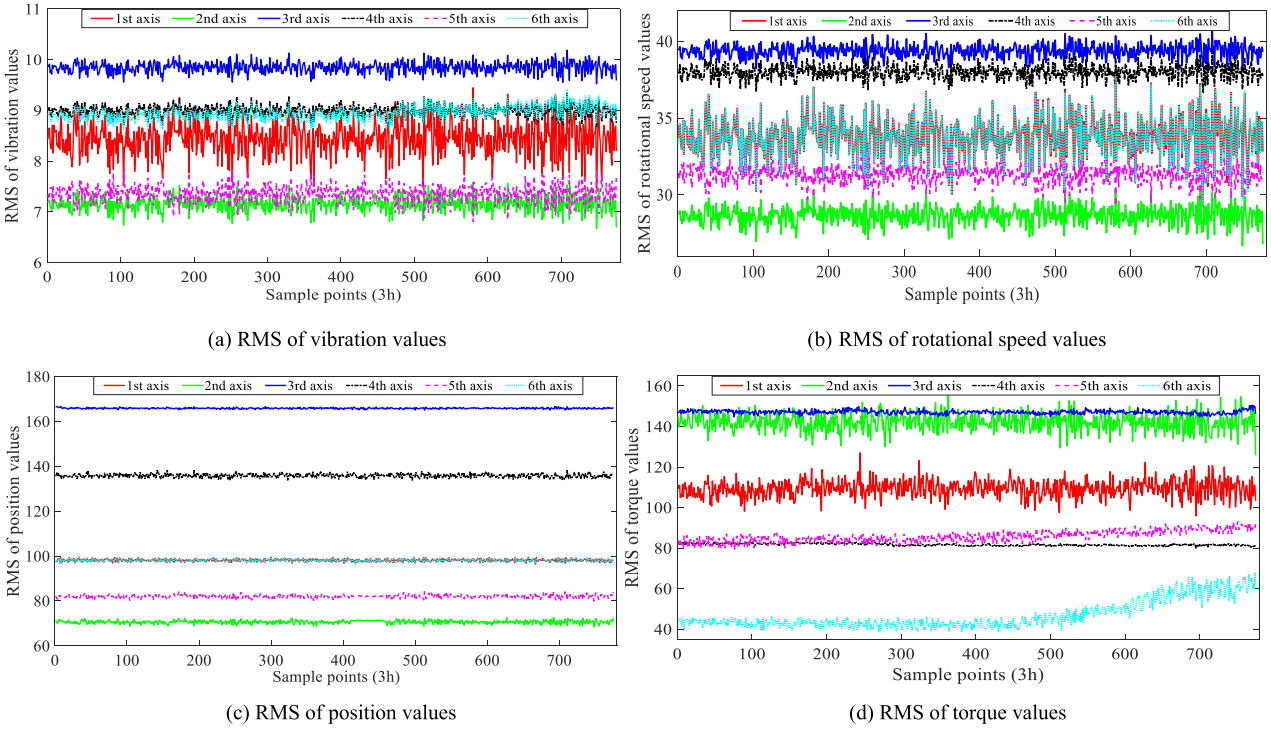
The In-patch view is used to establish relationships among patches, and it can obtain global features. The embedding is adopted to map each patch, and rearrange them to form in-patch representation $X_i^{ip} \in R^{P \times d_m'}$, where $d_m' = S \times d_m$. And, the linear transformation is to convert patch-wise representation for obtaining key vectors K_{ip}^i and value vectors V_{ip}^i:

$$\begin{cases} K_{ip}^i = W_K' X_i^{ip} \\ V_{ip}^i = W_V' X_i^{ip} \end{cases} \tag{26}$$

where W_K' and W_V' denote linear transformation, and the in-patch attention can be calculated as:

$$Attn_{ip}^i(Q, K, V) = softmax\left(\frac{Q_{ip}^i \left(K_{ip}^i\right)^T}{\sqrt{d_m'}}\right) V_{ip}^i \tag{27}$$

where $Q_{ip}^i \in R^{1 \times d_m'}$ represents the trainable vectors. The attention out-



(a) RMS of vibration values

(b) RMS of rotational speed values

(c) RMS of position values

(d) RMS of torque values

Fig. 7. The extracted RMS values of different sensors.

puts of all patches are integrated to form the final outputs of the in-patch view:

$$Attn_{ip} = Concat\left(Attn_{ip}^{1}, \ldots, Attn_{ip}^{P}\right) \tag{28}$$

The outputs of dual attention mechanism $Attn \in R^{P\times S\times d_m}$ are expressed as:

$$Attn = Attn_{ip} + Dense\left(Attn_{pw}\right) \tag{29}$$

where $Dense(\cdot)$ is used to transform the dimension of $Attn_{pw}$ and align its dimension with $Attn_{ip}$.

3.3.3. Pathway fusion module

The pathway fusion module is used to integrate the features of multiscale patches. The dimension of R_T is related to the patch size of multiscale attention module. $R_w>0$ denotes the multiscale attention module for the patch size S_i is implemented. $R_w = 0$ indicates multiscale attention module for patch size S_i is ignored. Assuming that the outputs of patch size S_i in the multiscale attention module are defined as $Attn_i$, weighted aggregation is executed to integrate these outputs. The outputs of different patch sets are fused for RUL prediction:

$$RUL_{pred} = \sum_{i=1}^{M} \mathcal{I}(R_T > 0) R_T Dense(Attn_i) \tag{30}$$

where $\mathcal{I}(R_w > 0)$ represents the indicator function. Specifically, the output of indicator function is 1 when $R_w > 0$, while output is 0 when $R_w = 0$.

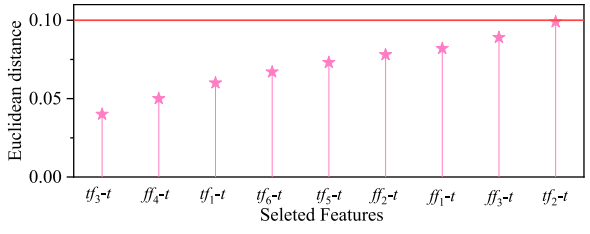


Fig. 8. Feature selection based on Euclidean distance.

4. Experimental study

The experimental study is conducted to verify the proposed method for RUL prediction. The proposed Transformer framework is written using Python v3.9.0 with Pytorch 1.11.0, executed on a computer with an AMD Ryzen 9-5900X and an NVIDIA RTX 3080 GPU.

4.1. Experimental setup

A self-built IR platform is constructed to validate our proposed method. The self-built IR mainly comprises 6 axes, controlling electric motors, reducers, and transmissions. The rated load of the IR is 20kg. An accelerated life test is conducted on the IR for grasping an iron block, and the operating load is set at 1.5 times the rated load. A series of actions, i.e., vertical grasping, lateral translation at a 320° angle, and vertical placing, are executed repeatedly. Four types of sensors, vibration sensor, speed sensor, position sensor, and torque sensor, are mounted on each axis for signal collection. The monitoring system collects data online by calling the function library of application programming interface (API) at a frequency of 15 Hz, and each sample period contains 3000 sample points. The sampling data is recorded every 3 hours. The reducers of the 5th-axis and 6th-axis gradually begin to generate noise and grease leakage at 1480h because of mechanical wear. Details of the self-built IR platform is shown in Fig. 6.

The extracted RMS values of different sensors are exhibited in Fig. 7. The speed, vibration, and position values do not show an obvious degradation process. This is because a closed-loop control is applied on each joint. We can see that the 3rd-axis and 4th-axis exhibit slight degradation in the torque values. The 5th-axis and 6th-axis exhibit obvious degradation. Eventually, a reducer fault occurred on the 6th-axis at 2325h, and 1480h is defined as the real state change point of IR. The 6th-axis is used for RUL prediction.

Table 1
Hyper-parameters of the customized dual-Transformer framework.

Network	Hyper-parameters	Values	Network	Hyper-parameters	Values
MATNet	d_m	32	MTTNet	K	4
	Window size	5		N	6
	δ	0.9		Patch size	{1,2,4,8}
	Λ	0.2		Dropout rate	0.1
	Learning rate	0.0005		d_m	32
	Number of memory	128		Sliding window size	40
	Batch size	128		Learning rate	0.0005
	Max epochs	50		Max epochs	100

4.2. Experimental results

The data normalization and moving average smoothing algorithm are adopted to preprocess the collected data. Feature extraction and selection are used to process the measured signals. 7-dimensional time domain and 5-dimensional frequency domain features are extracted for each axis, and they are shown in Fig. 8. The threshold of Euclidean distance is set as 0.1. The extracted features from torque sensors, i.e., tf_3-t, ff_4-t, tf_1-t, tf_6-t, tf_5-t, ff_2-t, tf_1-t, ff_3-t, tf_2-t, are eventually selected. These extracted features fall below the set threshold while others exceed it. Therefore, these nine features are used to state change identification and RUL prediction of IR.

To achieve RUL prediction, the first step is to train the MATNet for state change detection, and the second step is to train the MTTNet for RUL prediction of IR. Hyper-parameters of the proposed Transformer framework are shown in Table 1. These parameters are determined by grid-search, and the Adam optimizer is employed with a learning rate set as 0.0005 for MATNet and MTTNet. The early stopping strategy is used for training the proposed method.

The fake degradation (FD) and fake normal (FN) are chosen for detection performance evaluation. FD is defined as false degradation labels before the predicted state change point, and FN is defined as false normal labels after the predicted state change point. The former 200 samples are used to train the MATNet and the remaining samples are used for testing. n_f of sample points is set as 3. The detection results of state change detection are shown in Fig. 9. D and N represent the degradation state and normal state, respectively. Red and green areas represent the normal stage and degradation stage, respectively. The state change point is 488, and the number of FD is 1 while FN is 0. The detection results show that our proposed MATNet can effectively detect state change and warn the degradation state earlier. Three evaluation metrics, normalized root mean square error (NRMSE), Score, and accuracy (AC) are adopted to quantitatively estimate the prediction performance of the methods:

$$\text{NRMSE} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\Delta R^2}}{\frac{1}{N}\sum_{i=1}^{N}\widehat{R}_i} \tag{31}$$

$$\text{Score} = \begin{cases} \frac{1}{N}\sum_{i=1}^{N}\left(\exp\left(-\frac{\Delta R}{13}\right)-1\right), & \text{if } \Delta R \le 0 \\ \frac{1}{N}\sum_{i=1}^{N}\left(\exp\left(\frac{\Delta R}{10}\right)-1\right), & \text{if } \Delta R > 0 \end{cases} \tag{32}$$

$$\text{AC} = \left(1-\frac{|\Delta R|}{\widehat{R}_i}\right)\times 100\% \tag{33}$$

where $\widehat{R}_i$ denotes predicted RUL values. $\Delta R = \widehat{R}_i - R_{real}$ denotes the prediction error between predicted RUL values and actual RUL values. N is the total number of sample points.

The RUL prediction performance of the MTTNet is validated at diverse known sample points (100 - 260). The known sample points are used for training and the remaining sample points are used for testing. Fig. 10 shows that prediction performance is related to the number of given sample points. As the sample points become longer, the prediction capacity improves. The accuracy reaches 95.30% when number of known sample points is 260.

To further validate the prediction performance of MTTNet, 90%, 60%, and 30% of sample points in the degradation state are used to

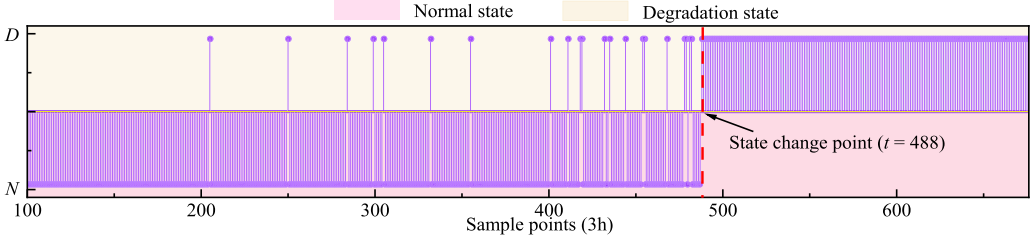


Fig. 9. Detection results of MATNet.

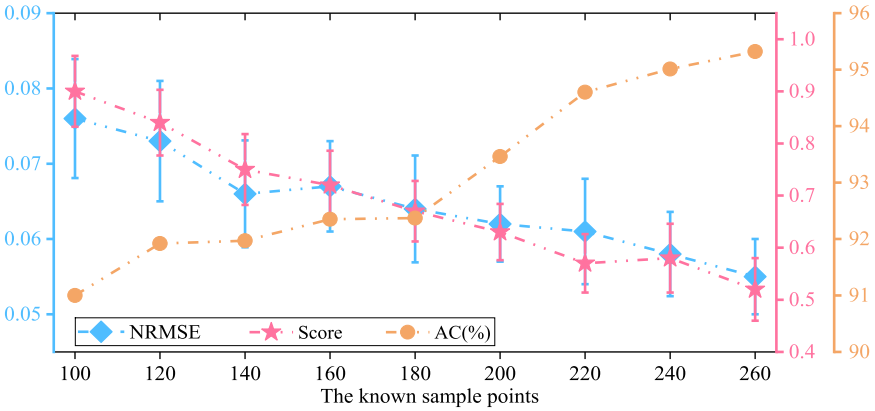


Fig. 10. RUL prediction performance of known sample points.

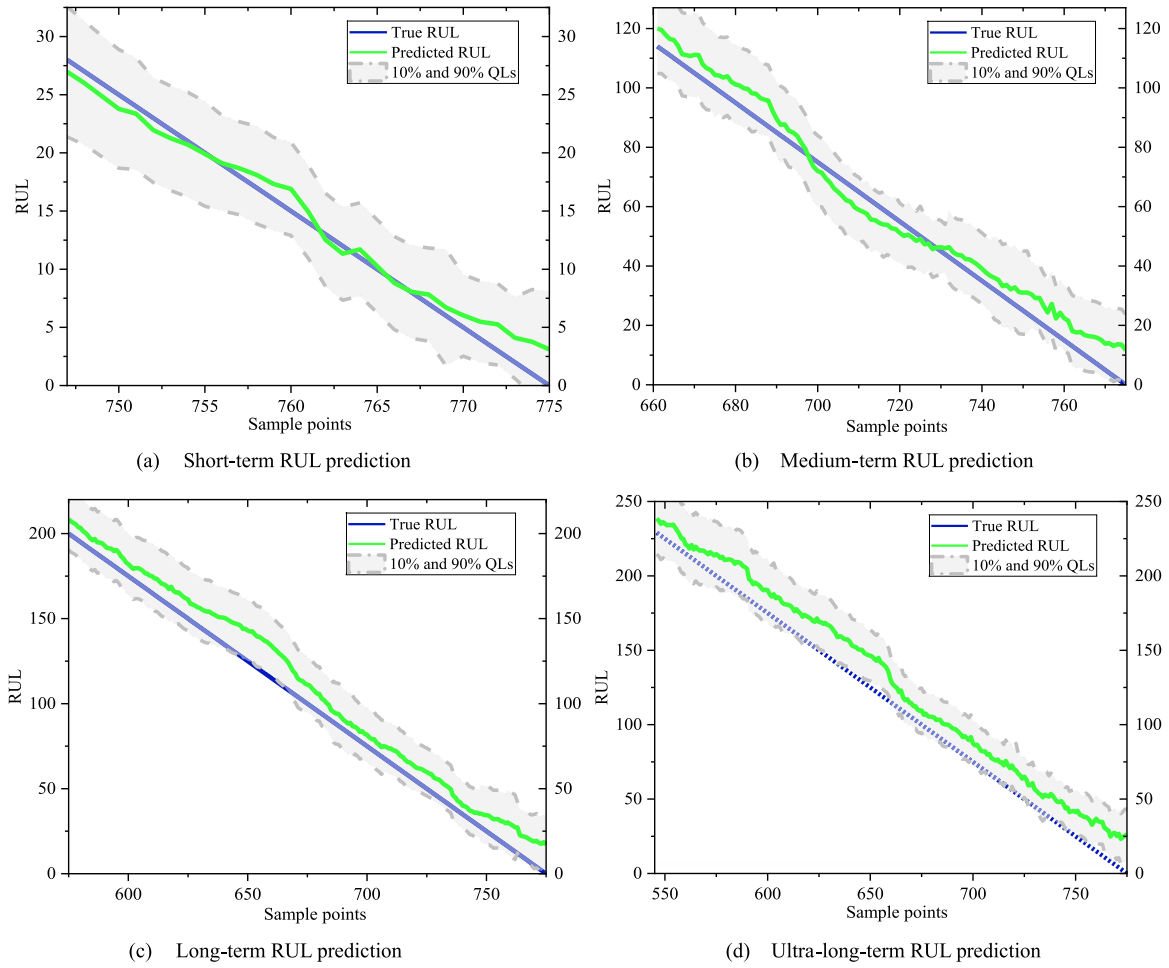


Fig. 11. RUL prediction in different temporal terms.

perform the short-term, medium-term, and long-term prediction [40, 42]. 20% of sample points in the degradation state are used to validate the ultra-long-term prediction performance. The RUL prediction results with uncertainty quantification in the different temporal terms are shown in Fig. 11. The predicted RUL values are 50% quantile level (QL), while the zones between 10% QL and 90% QL are marked as grey. It is observed that predicted RUL values fluctuate in the initial and medium degradation stages and converge in the end degradation stage. The uncertainty zones of RUL prediction are wider in the long-term and ultra-long-term RUL prediction. This is because training sample points are limited compared with the short-term and medium-term predictions. Fig. 11 demonstrates that the proposed MTTNet can effectively describe the degradation trend of the IR. Based on the uncertainty quantification, the maintenance schedules for IR can be well-designed.

5. Discussion

The discussion is conducted based on ablation analysis, parameter analysis, and comparison with other advanced methods. The ablation study for MATNet and MTTNet is performed. In the parameter analysis, three critical parameters, patch sizes, top-K operation, and threshold δ, are discussed. Additionally, comparison results with other advanced methods are executed to show the superiority of the proposed method.

5.1. Ablation study

To explore the effects of different parts in our proposed method, the ablation study is executed. To validate the detection performance of MATNet, three methods are investigated:

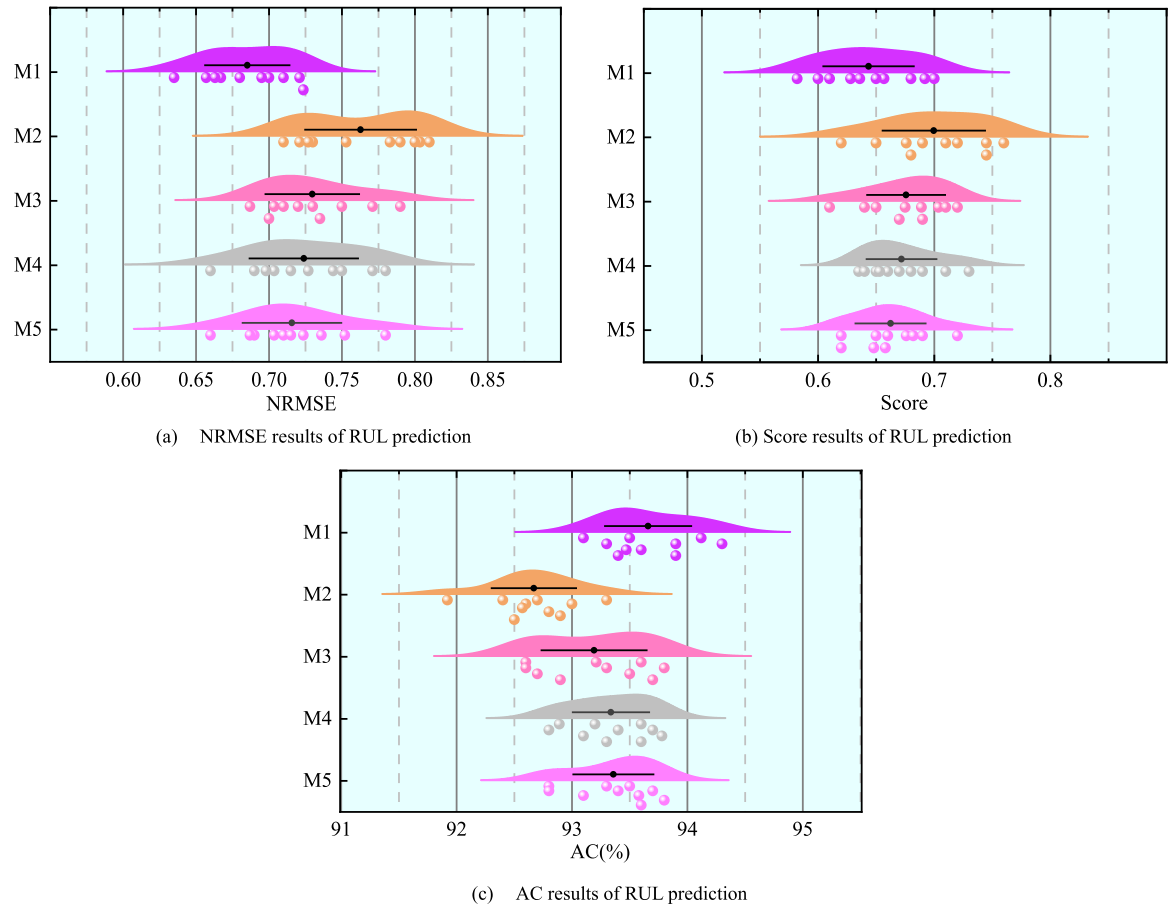
- Method-1(M1): The proposed MATNet.
- Method-2(M2): The proposed MATNet without enhanced memory module.
- Method-3(M3): Transformer encoder of MATNet is replaced as dense encoder.

Table 2
Detection results of three different methods.

Metrics	M1	M2	M3
FD	21	22	22
FN	0	0	1
Location	488	490	491

Table 2 depicts detection results of three methods. We can see that the proposed MATNet realizes superior detection results. The results of M2 indicate that the enhanced memory module can effectively enhance the temporal representation of degradation process, which improves the early detection ability and detection accuracy for state change. The results of M2 is better than M3, demonstrating that Transformer encoder can effectively capture the temporal information for state change detection.

To validate the RUL prediction performance of MTTNet, the former 200 points are used for training and the remaining points are adopted for testing. Five methods are adopted for comparison:



(a) NRMSE results of RUL prediction

(b) Score results of RUL prediction

(c) AC results of RUL prediction

Fig. 12. The prediction results of ablation study for different methods.

- Method-1(M1): The proposed MTTNet.
- Method-2(M2): The proposed MTTNet without state change detection.
- Method-3(M3): The proposed MTTNet without patching construction module.
- Method-4(M4): The proposed MTTNet without patch-wise view.
- Method-5(M5): The proposed MTTNet without in-patch view.

We can see from Fig. 12 that the proposed method can achieve outstanding prediction performance. Compared with M2, the NRMSE and Score values of M1 are lower. This demonstrates that state change detection is an important task for identifying the initial degradation point. Combined with state change detection, the performance of NTNNet is effectively improved to predict the RUL. The results of M3 show that the multiscale operation operation can enable the model to mine multi-term temporal features for enhancing the prediction performance. It shows that prediction capacity is weakened when the patch-wise view is removed. This is because the patch-wise view provides local temporal information on degradation process. Removal of the in-patch view can lead to the loss of global temporal information. The performance of M4 is inferior to M5. This may be because global temporal information contains more abundant temporal features than local temporal information. To sum up, the RUL prediction is significantly improved by combining state change detection. Multiscale division and multiscale attention module can effectively mine the temporal information to promote the prediction capacity.

Table 3
Different patch sets.

Number	P1	P2	P3	P4	P5
Patch sets	{1}	{1,2}	{1,2,4}	{1,2,4,8}	{1,2,4,8,10}

5.2. Parameter analysis

5.2.1. Patch size

To investigate the influence of patch size and patch number, different patch sets are constructed for comparison. We choose the first 200 sample points for training the proposed MATNet, and all presented results are NRMSE average values of ten runs. We choose different patch sizes to form 5 kinds of patch sets, shown in Table 3. It is noted that the patch size should be evenly divided by the time window value. The patch size increases, and the patch sets become abundant. Fig. 13 shows that as the number of patch size enlarges, the NRMSE becomes small. However, as the selected patch sizes enlarge, the improvement of prediction performance becomes less. NRMSE of P3 is smaller than P4 in the ultra-long-term prediction, while the NRMSE of P4 is larger than P3 in the other terms. Moreover, the NRMSE of P5 exceeds that of P4. This may be because the increasing number of patch sizes causes the overfitting. Therefore, the patch sets with patch size {1, 2, 4, 8} can reach outstanding performance.

5.2.2. Top-K operation

The top-K operation controls the sparsity of patch set for prediction. Fig. 14 presents the NRMSE and Score results based on different K values. As the K value increases, the improvement of NRMSE results gradually diminishes. However, the Score metric does not follow the

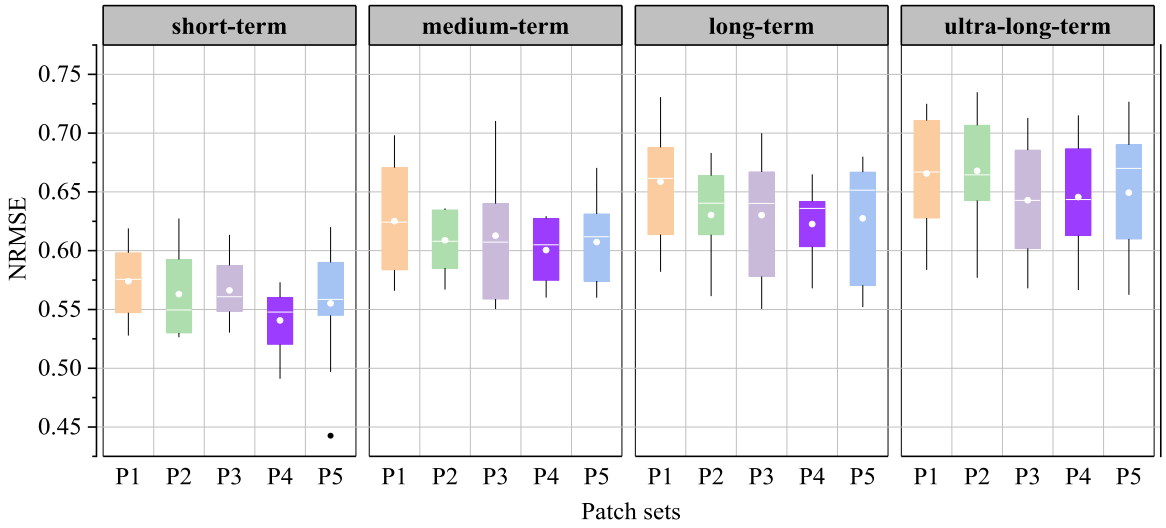


Fig. 13. Boxplots of NRMSE results with different patch sets.

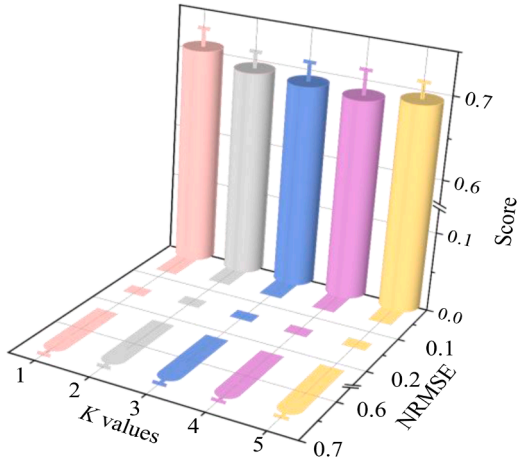


Fig. 14. NRMSE and Score results of different top-K operation.

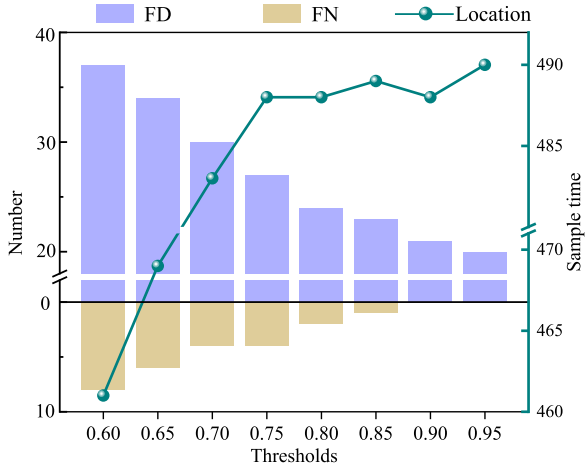


Fig. 15. . Detection performance of different thresholds.

same trend as NRMSE. When the K value exceed 4, the Score results worsen. This is because a large K value denotes that excessive multiscale temporal information is introduced into the model, which can lead to overfitting. Overall, $K = 4$ can reach excellent RUL prediction results by comprehensively considering NRMSE and Score results.

5.2.3. *Threshold δ*

To explore the effect of different thresholds on detection performance, eight different thresholds are chosen to determine optimal value δ. FN, FD, and location are used to evaluate detection performance of MATNet. It is observed from Fig. 15 that the number of FN and FD decreases as the threshold values increase. Additionally, as the threshold values become larger, the detected locations are gradually delayed. Taking detection results of different metrics into account, the detection performance of the MATNet achieves outstanding performance when the threshold δ is set as 0.9.

5.3. *Comparison with other advanced methods*

To further exhibit the superiority of our methods, comparison analysis is implemented. The detection performance of MATNet and prediction performance of MTTNet are compared with other advanced methods. To show the excellent detection performance of MATNet, seven advanced detection methods are chosen for comparison:

- Autoencoder (AE): The encoder is based on Transformer encoder and decoder is based on the dense layer.
- RVAE [43]: A variational autoencoder (VAE) that encoder and decoder are based on Transformer.
- STOC [44]: An improved autoencoder that is composed of Transformer encoder and convolution decoder.
- FCVAE [45]: A novel revisiting VAE that extracts local and global temporal information for anomaly detection.
- SSPCL [39]: A novel contrast learning method for state change detection.
- RSTRN [46]: A relation network-based detection method by estimating the similarity relations between health state and fault state.
- CWTKB [47]: An intelligent fault detection method with considering zero-fault samples.

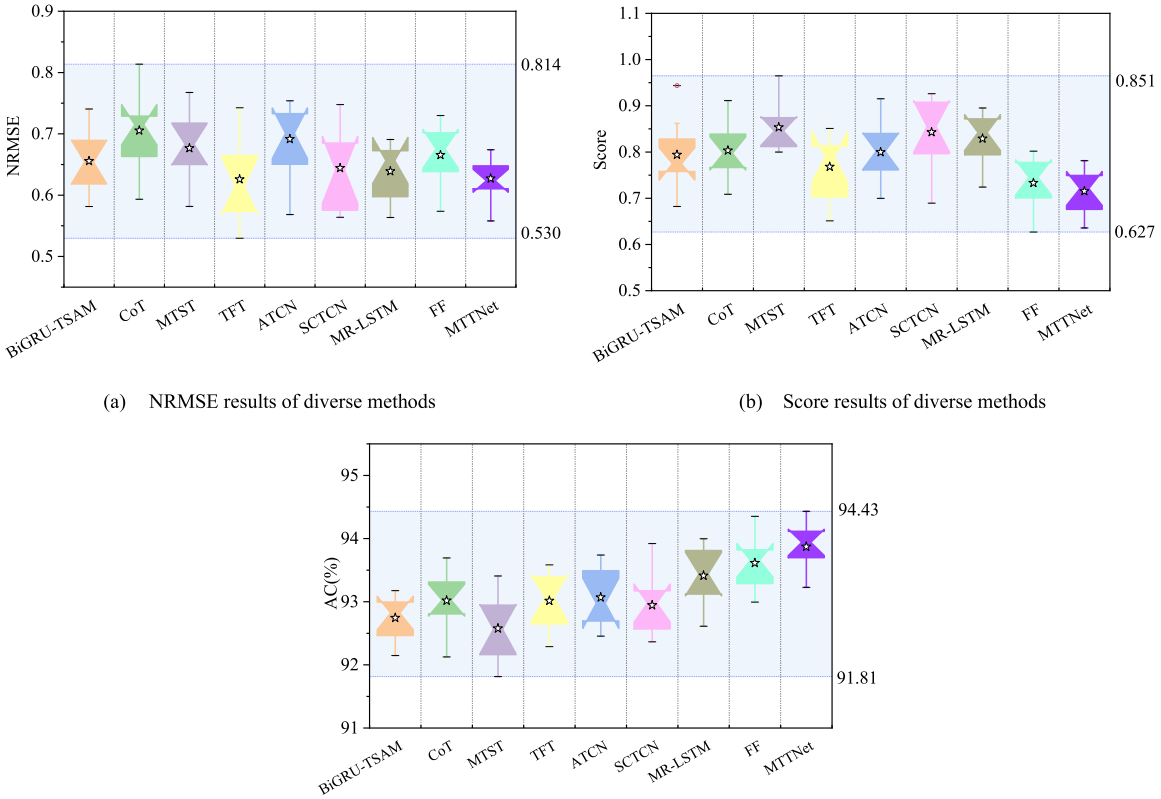
Location, FD, FN, and average training time (ATT) are adopted to assess the performance of MATNet. The results of state change identification are shown in Table 4. Compared with other advanced detection methods, our proposed MATNet detects the earlier detection location. The total number of FD and FN of the MATNet is the smallest among these methods. Compared with AE, the proposed MATNet has strong capacity to capture temporal dependencies based on the enhanced memory module, which can reveal variation of temporal information for

Table 4
Comparison with advanced detection methods.

Metrics	AE	RVAE	STOC	FCVAE	SSPCL	RSTRN	CWTKB	MATNet
Location	490	500	498	497	495	490	488	488
FD	32	21	23	22	18	21	22	21
FN	1	1	1	1	5	1	0	0
ATT(s)	1392.5	2450.2	2408.3	2445.4	3512.8	3639.0	2529.1	2553.7

Table 5
Comparison with advanced prediction methods.

Metrics	BiGRU-TSAM	CoT	MTST	TFT	ATCN	SCTCN	MR-LSTM	FF	MTTNet
NRMSE	0.69	0.68	0.67	0.66	0.67	0.65	0.66	0.64	0.61
Score	0.82	0.81	0.82	0.79	0.80	0.84	0.85	0.74	0.73
AC	92.8	92.9	92.7	93.0	93.3	93.2	93.3	93.5	93.8
ATT(s)	870.5	590.2	712.8	545.4	672.8	709.5	826.0	748.3	803.7



(a) NRMSE results of diverse methods

(b) Score results of diverse methods

(c) AC results of diverse methods

Fig. 16. Comparison of RUL prediction results of different methods.

detection. Moreover, our MATNet has superior detection performance compared with other improved autoencoder-based methods. These results show that MATNet has the outstanding detection capacity compared with other advanced methods. However, the ATT of MATNet is higher than that of many other methods, which is attributed to the complex structure of enhanced memory module.

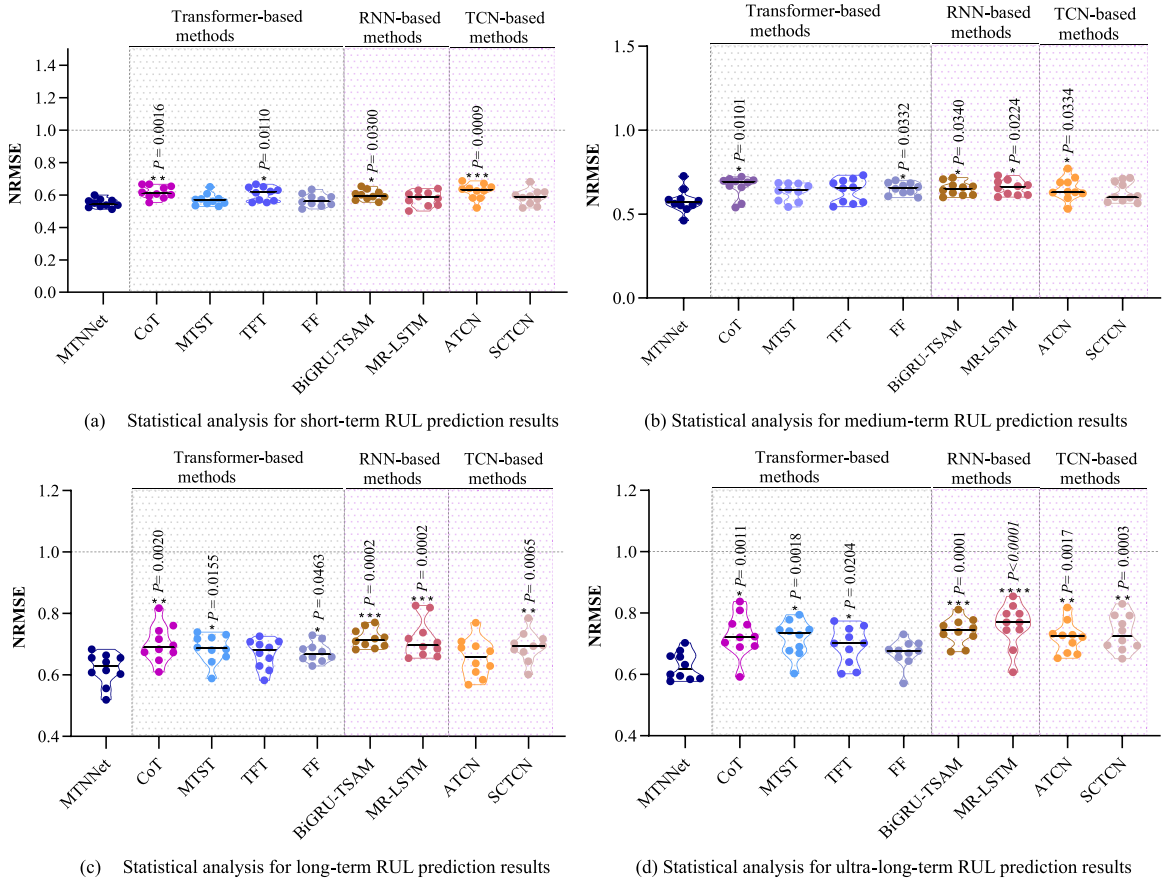
In addition, eight advanced prediction methods are chosen for comparison to demonstrate the outstanding prediction performance of the proposed MTTNet. The details of these prediction methods are as follows:

- BiGRU-TSAM [48]: A RUL prediction method based on bidirectional GRU with self-attention mechanism.
- CoT [49]: An improved Transformer network by combining multi-scale convolution and Transformer decoder.
- MTST [50]: A multi-resolution Transformer to achieve long-term time-series forecasting.
- TFT [51]: An improved Transformer based on multi-head prob--sparse attention for RUL prediction.
- ATCN [52]: A self-attention-based TCN for RUL prediction.
- SCTCN [53]: A TCN enhanced by self-calibration module for RUL prediction.
- MR-LSTM [54]: A multiscale LSTM network used for RUL prediction.
- FF [55]: A frequency information enhanced Transformer for time series forecasting.

We choose the first 150 sample points in the degradation state for

Table 6
Comparison with advanced prediction methods in multi-term RUL prediction tasks.

Methods	Short-term prediction			Medium-term prediction			Long-term prediction			Ultra-long-term prediction		
	NRMSE	Score	AC	NRMSE	Score	AC	NRMSE	Score	AC	NRMSE	Score	AC
BiGRU-TSAM	0.58	0.63	93.5	0.67	0.78	93.0	0.71	0.90	92.0	0.75	1.05	91.4
CoT	0.59	0.76	93.7	0.67	0.80	93.2	0.70	0.89	92.5	0.72	1.00	91.0
MTST	0.58	0.72	93.8	0.64	0.76	92.7	0.68	0.85	92.2	0.73	0.96	90.8
TFT	0.61	0.63	94.4	0.63	0.78	93.4	0.66	0.87	92.8	0.70	0.97	90.5
ATCN	0.62	0.71	94.4	0.66	0.81	93.5	0.67	0.84	92.9	0.73	0.98	91.0
SCTCN	0.59	0.66	94.0	0.64	0.82	93.6	0.69	0.90	93.0	0.71	0.97	91.7
MR-LSTM	0.58	0.65	94.3	0.65	0.84	93.8	0.71	0.92	92.3	0.77	1.03	89.8
FF	0.56	0.63	94.6	0.63	0.71	94.0	0.65	0.79	93.1	0.67	0.92	92.6
MTNNet	**0.54**	**0.61**	**94.9**	**0.60**	**0.69**	**94.2**	**0.62**	0.79	**93.6**	**0.64**	0.90	**93.1**
IMP (%)	3.57	3.17	0.32	4.76	2.82	0.21	4.62	-	0.54	4.48	0.22	0.54



(a) Statistical analysis for short-term RUL prediction results

(b) Statistical analysis for medium-term RUL prediction results

(c) Statistical analysis for long-term RUL prediction results

(d) Statistical analysis for ultra-long-term RUL prediction results

Fig. 17. Statistical analysis for multi-term RUL prediction results of different methods.

training the MTTNet, and the remaining points are used to test the trained model. Table 5 and Fig. 16 exhibit the prediction results. The prediction dispersion of our method is smaller than other methods. Although ATT results show that the proposed MTTNet has no advantage over other Transformer-based methods, RUL prediction accuracy is more important than time cost in industrial scenarios. To sum up, our method has a strong capacity to capture temporal information for high-precision RUL prediction.

The multi-term prediction results of these methods are presented in Table 6. Table 6 displays that our proposed MTTNet has the smallest NRMSE, and the prediction accuracy is higher than other methods. Although the Score of FF is equal to MTTNet in the long-term prediction, the performance in other multi-term predictions of FF is inferior to the MTTNet. The prediction performance of RNN-based and TCN-based methods is worse than that of our method because they suffer from learning long-term dependencies. Our proposed MTTNet has strong capacity to mine long-term dependencies by patch-wise view. Moreover, our method outperforms other improved Transformer methods because its in-patch view can focus on local features, which can introduce short-term temporal information to enhance the prediction capacity. As the temporal terms increase, the improvement of performance decreases. The average prediction accuracy of short-term, medium-term, and long-term, and ultra-long-term is improved by 2.35%, 2.60%, 1.72%, and 1.75%. The prediction results demonstrate that the proposed MTTNet can effectively achieve outstanding performance in the RUL prediction tasks.

The prediction results are chosen for paired *t*-test based on NRMSE, shown in Fig. 17. Asterisks (*) indicate a significant difference between results of two methods (*$p<0.05$, **$p<0.01$, ***$p<0.001$,****$p<0.001$). The prediction results demonstrate that our method outperforms these eight methods at an approximately 5% statistical significance level. In the short-term RUL prediction task, the predictive performance

Table 7
Mean difference results between our method and different methods.

Metrics	BiGRU-TSAM	CoT	MTST	TFT	ATCN	SCTCN	MR-LSTM	FF
diff	-0.085	-0.085	-0.058	-0.058	-0.068	-0.068	-0.084	-0.047

differences among the methods are minimal. As the prediction length extends, the number of methods with significantly higher NRMSE than our method steadily increases. Table 7 exhibits the mean difference (*diff*) for each method, and results show that the proposed method exhibits higher *diff* values compared with these methods. We can conclude that the proposed method provides superior prediction performance and robustness compared to other advanced methods.

6. Conclusion

In this paper, we propose a multiscale temporal memory Transformer framework for degradation-aware RUL prediction of IRs. MATNet is developed to detect the state change between the normal state to the degradation state. Furthermore, MTTNet is designed to extract both local and global temporal features for RUL prediction beyond the state change point. To validate the effectiveness and superiority of the proposed method, a self-built IR platform under accelerated life testing is developed for experimental validation. Experimental results demonstrate that the proposed method accurately predicts the RUL of IRs and that the proposed Transformer framework outperforms many other advanced methods.

In the future, we will further explore degradation assessment of IRs, as the degradation state can be divided into various health stages. A multi-stage classification method based on the proposed Transformer framework will be further investigated. RUL prediction and health assessment for each stage will be conducted to better support decision-making for IRs. An accelerated life test of a palletizing robot in industrial scenarios is currently being implemented, and we will validate our method on this robot. Additionally, we will consider uncertainties, such as sensor uncertainty, for further experimental validation.

CRediT authorship contribution statement

Zhan Gao: Writing – original draft, Validation, Methodology, Data curation. **Chengjie Wang:** Software, Investigation, Formal analysis, Data curation. **Jun Wu:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization. **Yuanhang Wang:** Validation, Supervision, Resources. **Weixiong Jiang:** Investigation, Formal analysis. **Tianjiao Dai:** Visualization, Project administration, Investigation.

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 the National Natural Science Foundation of China under the Grant No. 52075202 and 523B2100, in part by the Cross disciplinary Research Support Program of Huazhong University of Science and Technology under Grant No. 2024JCYJ028.

Data availability

Data will be made available on request.

References

[1] Kim Y, et al. Not only rewards but also constraints: Applications on legged robot locomotion. IEEE Transactions on Robotics 2024.
[2] Gao Z, Jiang W, Wu J, Wang Y, Zhu H. A customized dual-transformer framework for remaining useful life prediction of mechanical systems with degraded state. Mech. Syst. Signal Process. 2025;230:112611.
[3] Zafar MH, Langås EF, Sanfilippo F. Exploring the synergies between collaborative robotics, digital twins, augmentation, and industry 5.0 for smart manufacturing: A state-of-the-art review. Robotics and Computer-Integrated Manufacturing 2024;89:102769.
[4] He Y, Zhao C, Zhou X, Shen W. MJAR: A novel joint generalization-based diagnosis method for industrial robots with compound faults. Robot. Comput.-Integr. Manuf. 2024;86:102668.
[5] Pal A, Leite AC, From PJ. A novel end-to-end vision-based architecture for agricultural human–robot collaboration in fruit picking operations. Robotics and Autonomous Systems 2024;172:104567.
[6] Xiang S, Qin Y, Luo J, Wu F, Gryllias K. A concise self-adapting deep learning network for machine remaining useful life prediction. Mech Syst Sig Process 2023;191:110187.
[7] Du P, Zhong W, Peng X, Li Z, Li L. Fault effect identification-based adaptive performance self-recovery control strategy for wastewater treatment process. IEEE Transactions on Industrial Informatics 2023;20(3):3585–96.
[8] Zhang F, et al. Cloud-free land surface temperature reconstructions based on MODIS measurements and numerical simulations for characterizing surface urban heat islands. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 2022;15:6882–98.
[9] Zhao X, et al. Intelligent fault diagnosis of gearbox under variable working conditions with adaptive intraclass and interclass convolutional neural network. IEEE Transactions on Neural Networks and Learning Systems 2022;34(9):6339–53.
[10] Du P, Zhong W, Peng X, Li L, Li Z. Self-healing control for wastewater treatment process based on variable-gain state observer. IEEE Transactions on Industrial Informatics 2023;19(10):10412–24.
[11] Kumaraswamidhas L, Laha S. Bearing degradation assessment and remaining useful life estimation based on Kullback-Leibler divergence and Gaussian processes regression. Measurement 2021;174:108948.
[12] Chen L, Wu X, Lopes AM, Yin L, Li P. Adaptive state-of-charge estimation of lithium-ion batteries based on square-root unscented Kalman filter. Energy 2022;252:123972.
[13] Li Q, Li D, Zhao K, Wang L, Wang K. State of health estimation of lithium-ion battery based on improved ant lion optimization and support vector regression. J. Energy Storage 2022;50:104215.
[14] Jiang W, Wu J, Zhu H, Li X, Gao L. Paired ensemble and group knowledge measurement for health evaluation of wind turbine gearbox under compound fault scenarios. J. Manuf. Syst. 2023;70:382–94.
[15] Haensch A, Tronci EM, Moynihan B, Moaveni B. Regularized hidden Markov modeling with applications to wind speed predictions in offshore wind. Mech. Syst. Signal Process. 2024;211:111229.
[16] Ordóñez C, Lasheras FS, Roca-Pardinas J, de FJ, Juez Cos. A hybrid ARIMA–SVM model for the study of the remaining useful life of aircraft engines. J. Comput. Appl. Math. 2019;346:184–91.
[17] Wang Y, Yu Z, Wu J, Wang C, Zhou Q, Hu J. Adaptive Knowledge Distillation Based Lightweight Intelligent Fault Diagnosis Framework in IoT Edge Computing. IEEE Internet Things J 2024.
[18] Li X, Shao H, Lu S, Xiang J, Cai B. Highly efficient fault diagnosis of rotating machinery under time-varying speeds using LSISMM and small infrared thermal images. IEEE Transactions on Systems, Man, and Cybernetics: Systems 2022;52(12):7328–40.
[19] Zhu X, Zhao X, Yao J, Deng W, Shao H, Liu Z. Adaptive multiscale convolution manifold embedding networks for intelligent fault diagnosis of servo motor-cylindrical rolling bearing under variable working conditions. IEEE/ASME Transactions on Mechatronics 2023.
[20] Jing T, Zheng P, Xia L, Liu T. Transformer-based hierarchical latent space VAE for interpretable remaining useful life prediction. Adv. Eng. Inf. 2022;54:101781.
[21] Zhao X, et al. Model-assisted multi-source fusion hypergraph convolutional neural networks for intelligent few-shot fault diagnosis to electro-hydrostatic actuator. Information Fusion 2024;104:102186.
[22] Yang Q, et al. Fault prognosis of industrial robots in dynamic working regimes: Find degradation in variations. Measurement 2021;173:108545.
[23] Cardenas-Lizana P, Pasaguayo L, Alvarado SAL, Al Masry Z. An Ensemble Learning Methodology for Predicting Medical Micro-robot Degradation Classes. In: European Safety and Reliability Conference; 2022.
[24] Taha HA, Yacout S, Birglen L. Detection and monitoring for anomalies and degradation of a robotic arm using machine learning. Advances in Automotive Production Technology–Theory and Application: Stuttgart Conference on Automotive Production (SCAP2020). Springer; 2021. p. 230–7.
[25] Xiao H, Zeng H, Jiang W, Zhou Y, Tu X. HMM-TCN-based health assessment and state prediction for robot mechanical axis. Int. J. Intell. Syst. 2022;37(12):10476–94.

[26] Lulu J, Yourui T, Jia W. Remaining Useful Life Prediction for Reducer of Industrial Robots Based on MCSA. 2021 Global Reliability and Prognostics and Health Management (PHM-Nanjing). IEEE; 2021. p. 1–7.
[27] Guo J, Lei S, Du B. MHT: A multiscale hourglass-transformer for remaining useful life prediction of aircraft engine. Eng. Appl. Artif. Intell. 2024;128:107519.
[28] Shabani A, Abdi A, Meng L, Sylvain T. Scaleformer: Iterative multi-scale refining transformers for time series forecasting. arXiv preprint arXiv:2206.04038 2022.
[29] Chen Y, Ren K, Wang Y, Fang Y, Sun W, Li D. ContiFormer: Continuous-time transformer for irregular time series modeling. Advances in Neural Information Processing Systems 2024;36.
[30] Ren L, Jia Z, Wang X, Dong J, Wang W. A $ T^{2} $-Tensor-Aided Multiscale Transformer for Remaining Useful Life Prediction in IIoT. IEEE Trans. Ind. Inform. 2022;18(11):8108–18.
[31] Gao Z, Jiang W, Wu J, Dai T, Zhu H. Nonlinear Slow-varying Dynamics-assisted Temporal Graph Transformer Network for Remaining Useful Life Prediction. Reliab. Eng. & Syst. Saf. 2024:110162.
[32] Li J, et al. A dual-scale transformer-based remaining useful life prediction model in industrial Internet of Things. IEEE Internet Things J 2024.
[33] Yang S, Tang B, Wang W, Yang Q, Hu C. Physics-informed multi-state temporal frequency network for RUL prediction of rolling bearings. Reliability Engineering & System Safety 2024;242:109716.
[34] Zhong Z, Yu Z, Yang Y, Wang W, Yang K. PatchAD: A Lightweight Patch-Based MLP-Mixer for Time Series Anomaly Detection. arXiv preprint arXiv:2401.09793 2024.
[35] Cheng Y, Zhu H, Wu J, Shao X. Machine health monitoring using adaptive kernel spectral clustering and deep long short-term memory recurrent neural networks. IEEE Trans. Ind. Inform. 2018;15(2):987–97.
[36] Liu L, Song X, Zhou Z. Aircraft engine remaining useful life estimation via a double attention-based data-driven architecture. Reliab Eng Syst Saf 2022;221:108330.
[37] Park H, Noh J, Ham B. Learning memory-guided normality for anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2020. p. 14372–81.
[38] Li W, Shang Z, Zhang J, Gao M, Qian S. A novel unsupervised anomaly detection method for rotating machinery based on memory augmented temporal convolutional autoencoder. Eng. Appl. Artif. Intell. 2023;123:106312.
[39] Ding Y, Zhuang J, Ding P, Jia M. Self-supervised pretraining via contrast learning for intelligent incipient fault detection of bearings. Reliab. Eng. Syst. Saf. 2022; 218:108126.
[40] Yang S, Tang B, Wang W, Yang Q, Hu C. Physics-informed multi-state temporal frequency network for RUL prediction of rolling bearings. Reliab Eng Syst Saf 2024;242:109716.
[41] Chen P, et al. Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. arXiv preprint arXiv:2402.05956 2024.
[42] Guo W, He M. An integrated method for bearing state change identification and prognostics based on improved relevance vector machine and degradation model. IEEE Trans. Instrum. Meas. 2022;71:1–14.
[43] Wang X, Pi D, Zhang X, Liu H, Guo C. Variational transformer-based anomaly detection approach for multivariate time series. Measurement 2022;191:110791.
[44] Kim J, Kang H, Kang P. Time-series anomaly detection with stacked Transformer representations and 1D convolutional network. Eng. Appl. Artif. Intell. 2023;120: 105964.
[45] Wang Z, et al. Revisiting VAE for Unsupervised Time Series Anomaly Detection: A Frequency Perspective. In: Proceedings of the ACM on Web Conference 2024; 2024. p. 3096–105.
[46] Chen Z, Wu K, Wu J, Deng C, Wang Y. Residual shrinkage transformer relation network for intelligent fault detection of industrial robot with zero-fault samples. Knowl.-Based Syst 2023;268:110452.
[47] Li G, Wei M, Shao H, Liang P, Duan C. Wavelet knowledge-driven transformer for intelligent machinery fault detection with zero-fault samples. IEEE Sensors Journal 2024.
[48] Zhang J, Jiang Y, Wu S, Li X, Luo H, Yin S. Prediction of remaining useful life based on bidirectional gated recurrent unit with temporal self-attention mechanism. Reliab Eng Syst Saf 2022;221:108297.
[49] Ding Y, Jia M. Convolutional transformer: An enhanced attention mechanism architecture for remaining useful life estimation of bearings. IEEE Trans. Instrum. Meas. 2022;71:1–10.
[50] Zhang Y, Ma L, Pal S, Zhang Y, Coates M. Multi-resolution time-series transformer for long-term forecasting. International Conference on Artificial Intelligence and Statistics 2024:4222–30. PMLR.
[51] Chang Y, Li F, Chen J, Liu Y, Li Z. Efficient temporal flow Transformer accompanied with multi-head probsparse self-attention mechanism for remaining useful life prognostics. Reliab Eng Syst Saf 2022;226:108701.
[52] Zhang Q, Liu Q, Ye Q. An attention-based temporal convolutional network method for predicting remaining useful life of aero-engine. Eng. Appl. Artif. Intell. 2024; 127:107241.
[53] He K, Su Z, Tian X, Yu H, Luo M. RUL prediction of wind turbine gearbox bearings based on self-calibration temporal convolutional network. IEEE Transactions on Instrumentation and Measurement 2022;71:1–12.
[54] Xu T, Han G, Zhu H, Taleb T, Peng J. Multi-resolution LSTM-based prediction model for remaining useful life of aero-engine. IEEE Transactions on Vehicular Technology 2023.
[55] Piao X, Chen Z, Murayama T, Matsubara Y, Sakurai Y. Fredformer: Frequency debiased transformer for time series forecasting. In: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining; 2024. p. 2400–10.