A novel gearbox fault diagnosis method based on dynamic temporal convolution and multi-scale attention and its application

Caixia Fan[1], Hongchao Wang[2]
[1]Department of Engineering, Huanghe Science and Technology College, 666 Zijingshan South Road, Zhengzhou, 450063, China
[2]Mechanical and Electrical Engineering Institute, Zhengzhou University of Light Industry, 5 Dongfeng Road, Zhengzhou 450002, China
[2]Corresponding author
E-mail: [1]*fcx211@163.com*, [2]*hongchao1983@126.com*

Received 17 April 2026; accepted 22 September 2026; published online 20 November 2026
DOI https://doi.org/10.21595/jve.2026.26567

Check for updates

Copyright © 2026 Caixia Fan, et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract. Accurately identifying early weak faults in rotating machinery is essential for safe operation and intelligent condition monitoring. However, practical industrial diagnosis is frequently constrained by limited fault samples, strong background noise, and variations in speed and load. Conventional deep-learning models commonly employ fixed convolution kernels or computationally intensive architectures, limiting their adaptability to time-varying vibration patterns and increasing the risk of overfitting under small-sample conditions. To address these challenges, this paper proposes a lightweight DDSC-T-HRC-LSTM-HAA diagnostic framework integrating dynamic depth-wise separable temporal convolution (DDSC-T), hierarchical residual-connected long short-term memory (HRC-LSTM), hierarchical adaptive attention (HAA), and an auxiliary convolutional neural network (CNN) branch. DDSC-T uses input-dependent weights to aggregate parallel depth-wise separable temporal-convolution branches with different receptive fields, thereby adaptively extracting local transient and resonance-related features. HRC-LSTM subsequently models hierarchical dependencies among repetitive fault responses through stacked recurrent layers and residual projections. HAA sequentially recalibrates the learned representations along the channel and temporal dimensions, while the auxiliary CNN branch preserves complementary shallow morphological information through lightweight learnable fusion. Comprehensive evaluations on the public Politecnico di Torino rolling-bearing dataset (T1) and the laboratory QPZZ-II gearbox dataset (T2) yield overall accuracies of 99.7 % and 98.3 %, respectively. For the challenging 150 μm bearing-defect categories, the recall remains above 99.2 %. In direct transfer from the 100 Hz-0 N source condition to unseen speed–load conditions, the proposed model achieves condition-averaged Accuracy and Macro-F1 values of 95.2 % and 94.7 %, respectively, and retains 91.5 % Accuracy under the most challenging combined speed–load shift. Under additive white Gaussian noise, the accuracy remains above 98% at signal-to-noise ratios no lower than 5 dB and reaches 61.08 % at 0 dB. With only 0.31 M trainable parameters and 0.48 GFLOPs, the model achieves a single-segment inference latency of 9.8 ms on the NVIDIA Jetson Xavier NX, demonstrating a favorable balance among diagnostic performance, cross-condition robustness, and embedded deployment efficiency.

Keywords: rotating machinery fault diagnosis, small sample, dynamic temporal convolution, noise robustness, edge deployment.

1. Introduction

With the development of rotating machinery towards high speed and intelligence, the operating state of the gearbox, as a key component, directly determines the safety and reliability of the entire system [1-4]. In practical working conditions, gearboxes often operate in complex environments characterized by variable speeds and fluctuating loads, causing their vibration signals to exhibit

significant non-stationarity and strong noise interference. Meanwhile, since industrial equipment operates in a healthy state for the vast majority of the time, the cost of acquiring and annotating fault data is extremely high, resulting in typical "small sample" and data scarcity characteristics in field scenarios. It is particularly necessary to clarify that the "small sample" discussed in this paper is not the extreme few-shot learning in the traditional computer vision domain. Rather, it targets the practical pain point of industrial data-driven diagnosis: the available training samples for various faults are far fewer than the large-scale data volume required to adequately train a deep time-series network to convergence. In such a typical data-constrained scenario, traditional deep learning models are highly prone to overfitting, further limiting fault recognition accuracy and cross-condition generalization capabilities [5]. Therefore, researching intelligent diagnosis models that possess strong robustness, high stability, noise resistance, and operating-condition adaptability under small sample conditions has become a hotspot in the current fault diagnosis field [6].

From the perspective of mechanical system dynamics, early localized defects in gears and rolling bearings, such as pitting, micro-cracks, and indentation defects, can cause periodic abrupt changes in contact stiffness or meshing stiffness, thereby exciting high-frequency structural resonances in the system. However, in actual complex mechanical transmission chains, such weak transient high-frequency impacts are often attenuated by long physical transmission paths and masked by low-frequency, high-energy vibrations such as gear meshing harmonics, rotor imbalance, and load-induced resonance. Traditional mechanical signal processing methods heavily rely on experts to manually select resonance bands, characteristic frequencies, or statistical thresholds a priori, and they often fail under operating conditions characterized by non-stationary speeds and the coupling of multiple excitation sources. Umar et al. [7] recently proposed a distribution-level statistical framework for pipeline leak detection by integrating time-domain, frequency-domain, and time-frequency features, showing that multi-domain signal analysis can improve reliability when labeled fault data are scarce. Although this type of statistically grounded method is interpretable and data-efficient, it still depends strongly on handcrafted feature construction and is difficult to directly model complex nonlinear temporal degradation patterns in rotating machinery.

In recent years, deep learning technology has made significant progress in gear and bearing fault diagnosis. In particular, Convolutional Neural Networks (CNNs) [8] and Recurrent Neural Networks (RNNs) [9] have demonstrated powerful capabilities in local feature extraction and temporal modeling. Researchers have successively proposed new methods, including attention-guided WDCNN [10], CNN-LSTM-GNN [11], GMNR and ResNet-CABA-MAGRU [12], as well as CNN-LSTM [13]. To address the adaptability issue under complex working conditions, Wang et al. [14] introduced multi-scale convolution and attention mechanisms to enhance the perception of weak features. Liu et al. [15] proposed a lightweight residual network. Zhang et al. [16] presented the PDSResNet-AM method to reduce model complexity and highlight key features. For variable operating conditions, Siddique et al. [17] developed a multistage transfer learning framework for rotating machinery fault diagnosis, demonstrating that transferable representations are important for maintaining diagnostic performance under changing spindle speeds. However, the aforementioned methods still exhibit obvious limitations. First, many models mainly rely on static convolution kernels with fixed receptive fields and weights, making it difficult to adaptively adjust according to the time-varying spectral characteristics of the input signal. Second, this static mechanism causes the model to respond slowly to early weak transient impacts under complex background noise. Furthermore, conventional deep networks often introduce a large number of parameters, which is not always compatible with edge deployment in industrial monitoring systems.

Signal processing methods based on attention mechanisms are powerful tools for fault diagnosis under variable operating conditions [18], capable of dynamically allocating attention based on information such as data modalities [19]. For instance, Xu et al. [20] proposed a composite method combining a two-stage attention-based multi-scale CNN with Prediction

Interval Estimation (PIE); Wang et al. [21] constructed a multi-branch temporal attention network. Chen et al. [22] proposed a diagnostic model combining frequency-domain self-attention and residual connections. Nevertheless, existing attention methods generally suffer from insufficient capacity for the hierarchical adaptive regulation of multi-scale features. At the front-end feature extraction level, although methods like the Multi-Angle Domain Adversarial Network (MADAN) [23] and Parallel Convolution Module (PCM-ITN) [24] have emerged, they still rely on static convolution kernels, making it difficult to cope with weak impacts and frequency band drift [25]. At the temporal memory level, hybrid methods based on CEEMDAN-VMD [26] or multi-dimensional parameter fusion [27] are highly dependent on artificial thresholds and offline computations. In addition, general feature extractors [28] and attention mechanisms restricted solely to the channel dimension [29] lack deep synergy between long-range temporal modeling and multi-dimensional feature recalibration. Khan et al. [30] further emphasized, from the perspective of calibrated deep-learning risk assessment, that intelligent models should not only pursue high accuracy but also consider reliability, interpretability, and transfer validation. This view is also instructive for industrial fault diagnosis, where model robustness and generalization are as important as classification accuracy.

Fault detection is also crucial in other structural members beyond rolling bearings and gearboxes. Parida et al. [31] combined RNN and modified particle swarm optimization to predict multiple transverse cracks in composite beams. Rout et al. [32] investigated the dynamic response of multi-cracked hollow rectangular beams through theoretical modeling, finite element analysis, and modal testing. Parida et al. [33] further proposed a hybrid RNN-fuzzy-PSO model for forecasting multiple transverse cracks in laminated composite beam-like structures. These studies indicate that fault-related vibration responses are strongly associated with structural dynamics, damage severity, and modal characteristics. However, most structural-damage methods depend on predefined modal indicators, finite element modeling, or task-specific optimization, and they are not directly designed for small-sample multi-class diagnosis of rotating machinery under strong noise and speed-load variations.

Targeting the aforementioned engineering pain points, this paper proposes a lightweight and robust intelligent diagnosis framework tailor-made for small sample, strong noise, and variable-condition scenarios. This framework deeply integrates dynamic depth-wise separable temporal convolution (DDSC-T), hierarchical residual-connected long short-term memory (HRC-LSTM), hierarchical adaptive attention (HAA), and an auxiliary CNN branch. The core contribution of this paper is not a pure algorithmic innovation divorced from physical mechanisms, but rather a physically driven, engineering-level coupling of adaptive temporal perception, hierarchical sequence modeling, and channel–temporal feature recalibration tailored to the non-stationary vibration characteristics of rotating machinery. The specific contributions are summarized in the following aspects:

(1) Adaptive multi-scale temporal feature extraction under weak fault responses: Addressing the issue that weak transient impacts are easily submerged by background vibration, this paper proposes DDSC-T to extract dynamically weighted multi-scale temporal features. Compared with static convolution kernels, DDSC-T can adaptively emphasize temporal branches that better match the current vibration state, thereby improving the perception of local impacts and resonance-related fault components.

(2) Hierarchical temporal modeling and attention-based noise suppression under restricted samples: Given that early defects often manifest as weak and repetitive temporal responses, this paper introduces HRC-LSTM to model hierarchical dependencies among local fault features through residual temporal connections. HAA further recalibrates the learned representations along channel and temporal dimensions, enhancing fault-sensitive responses while suppressing redundant noise-dominated features.

(3) Cross-condition generalization and lightweight edge deployment: The proposed framework is evaluated on both a public bearing dataset and a laboratory gearbox dataset. In addition to within-dataset diagnosis, a direct cross-condition experiment is constructed on the T1 dataset to

examine speed-shift, load-shift, and combined speed-load-shift generalization without target-domain fine-tuning. The model contains only 0.31 M trainable parameters and achieves a single-segment inference latency of 9.8 ms on the NVIDIA Jetson Xavier NX platform, demonstrating its potential for real-time embedded fault diagnosis.

2. Theoretical basis and methodology

2.1. Dynamic depth-wise separable temporal convolution (DDSC-T)

Standard one-dimensional convolution simultaneously performs temporal filtering and cross-channel mixing, which usually results in a relatively large number of trainable parameters and high computational cost. For vibration-based fault diagnosis under small-sample and edge-deployment scenarios, this coupling may also reduce the sensitivity to weak transient impulses because fault-related local responses can be mixed with redundant inter-channel information at an early stage. To address this issue, depth-wise separable convolution is adopted in this study, in which the convolution operation is decomposed into two successive steps: depth-wise convolution and point-wise convolution.

Depth-wise convolution independently performs local cross-correlation on each input channel. This design is particularly suitable for bearing and gearbox vibration signals because early defects usually appear as short-duration impulse responses or local resonance components within specific channels or frequency bands. Compared with standard convolution, depth-wise convolution can preserve channel-wise fault-sensitive temporal patterns while substantially reducing the number of parameters. The depth-wise convolution operation can be expressed as:

$$Z_c^{(l)}(t) = \sigma\left(\sum_{i=0}^{k-1} K_c^{(l)}(i) X_c^{(l-1)}(t+i) + b_c^{(l)}\right), \tag{1}$$

where $X_c^{(l-1)}$ denotes the input sequence of the c-th channel in the $(l-1)$-th layer, $K_c^{(l)}$ is the depth-wise convolution kernel of the c-th channel in the l-th layer, k is the kernel length, $b_c^{(l)}$ is the bias term, and $\sigma(\cdot)$ denotes the nonlinear activation function. Eq. (1) follows the general formulation of depth-wise convolution, which has been widely used in lightweight neural networks.

After depth-wise filtering, point-wise convolution is used to linearly combine the channel-wise temporal features. It employs a 1×1 convolution to project the extracted local features into a new feature subspace without changing the temporal resolution. The point-wise convolution can be formulated as:

$$Y_m^{(l)}(t) = \sigma\left(\sum_{c=1}^{C} W_{m,c}^{(l)} Z_c^{(l)}(t) + b_m^{(l)}\right), \tag{2}$$

where C is the number of input channels, $W_{m,c}^{(l)}$ denotes the 1×1 convolution weight connecting the c-th input channel to the m-th output channel, and $b_m^{(l)}$ is the corresponding bias term. Through the combination of depth-wise and point-wise operations, the module achieves a more compact representation than standard convolution, while retaining the ability to capture local fault-related temporal features.

Although conventional depth-wise separable convolution is computationally efficient, a fixed convolution kernel may be insufficient for variable-speed or variable-load conditions. In rotating machinery, changes in rotational speed and load can shift the dominant resonance band and alter the distribution of vibration energy. Therefore, a fixed receptive field may fail to consistently match the fault-sensitive frequency components. To improve adaptability, the proposed DDSC-T

introduces multiple depth-wise temporal convolution branches with different kernel sizes, denoted as $K_1, K_2, \ldots, K_M$, and dynamically assigns an importance weight to each branch according to the input feature distribution. As shown in Fig. 1, the input feature X is simultaneously fed into a lightweight dynamic weighting path and several parallel depth-wise separable convolution branches.

In the dynamic weighting path, Global Average Pooling (GAP) is first applied to summarize the global channel-wise response of the current observation window. This operation provides a compact descriptor of the overall vibration energy distribution and reduces the influence of local random fluctuations. The pooled descriptor is then passed through two fully connected layers with a ReLU activation and a Sigmoid function to generate branch-wise dynamic weights:

$$\alpha = \mathrm{Sigmoid}(W_2\delta(W_1\mathrm{GAP}(X) + b_1) + b_2), \tag{3}$$

where $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_M]$ represents the dynamic weights of the M convolution branches, W_1 and W_2 are the trainable weight matrices of the two fully connected layers, b_1 and b_2 are the bias terms, and $\delta(\cdot)$ denotes the ReLU activation function. The Sigmoid function constrains each branch weight to the interval (0, 1), allowing the network to adaptively emphasize convolution branches that better match the current signal characteristics.

For the i-th branch, the output of the depth-wise separable temporal convolution can be written as:

$$F_i = \mathrm{PWConv}_i\big(\mathrm{DWConv}_i(X; K_i)\big), \tag{4}$$

where $\mathrm{DWConv}_i(\cdot)$ and $\mathrm{PWConv}_i(\cdot)$ denote the depth-wise and point-wise convolution operations of the i-th branch, respectively. The final DDSC-T feature is obtained by weighted aggregation of all branch outputs followed by one-dimensional average pooling:

$$F_{\mathrm{DDSC-T}} = \mathrm{AvgPool1D}\left(\sum_{i=1}^{M} \alpha_i F_i\right). \tag{5}$$

This dynamic aggregation mechanism enables the module to select an appropriate temporal receptive field for different vibration states. For signals dominated by short transient impulses, branches with smaller kernels can preserve local impact information; for signals containing longer periodic responses or broader resonance patterns, branches with larger kernels can provide more contextual temporal information. Therefore, DDSC-T is more flexible than a single-scale 1D-CNN and more lightweight than attention-heavy Transformer-based temporal encoders. Compared with standard CNN, GRU/LSTM-only structures, and conventional lightweight convolutional networks with fixed kernels, DDSC-T provides a better trade-off among local feature extraction, operating-condition adaptability, and computational efficiency.

The architecture of DDSC-T is illustrated in Fig. 1. The upper path generates the dynamic weights $(\alpha_1, \alpha_2, \ldots, \alpha_M)$ from the input feature X, whereas the lower path extracts multi-scale temporal features through parallel depth-wise separable convolution branches. Each branch output is multiplied by its corresponding dynamic weight and then fused through weighted aggregation. The aggregated feature is further compressed by one-dimensional average pooling to produce F_{DDSC-T}, which is subsequently delivered to the temporal modeling module for hierarchical sequential feature learning.

2.2. Hierarchical residual-connected long short-term memory (HRC-LSTM)

After the local fault-sensitive temporal features are extracted by DDSC-T, the remaining key problem is how to model the sequential dependency among repetitive impacts, modulation

components, and degradation-related temporal patterns. For rotating machinery, fault vibration signals are not isolated random impulses. Instead, they usually exhibit periodic or quasi-periodic characteristics associated with shaft rotational frequency, bearing characteristic frequency, gear meshing frequency, and their sideband components. Therefore, a temporal modeling module is required to capture both short-term transient responses and longer-range periodic dependencies.

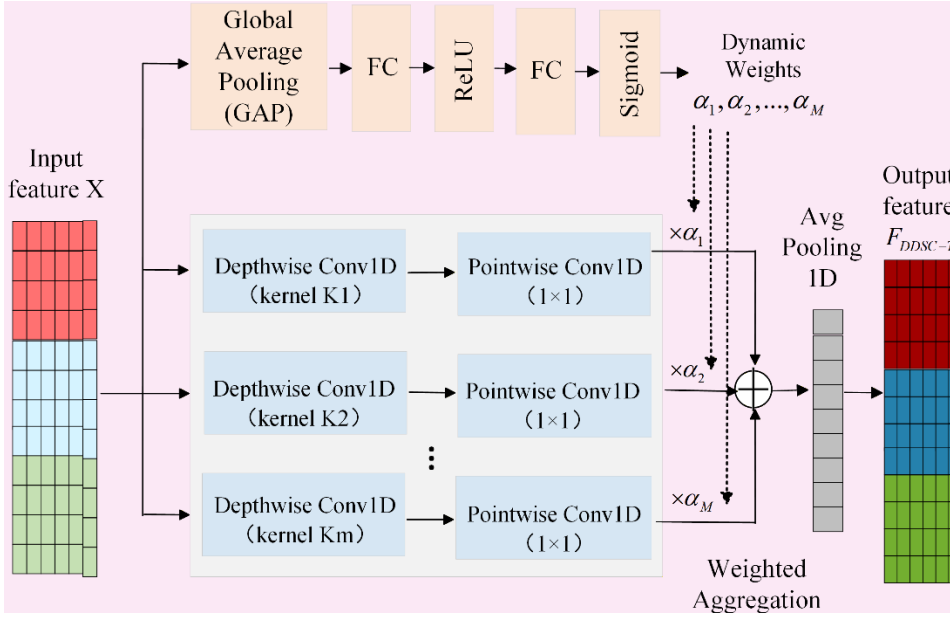


Fig. 1. Architecture of the proposed dynamic depth-wise separable temporal convolution (DDSC-T) module

Long Short-Term Memory (LSTM) is adopted as the basic temporal unit because its gating mechanism can selectively retain or discard historical information. Compared with a conventional recurrent neural network, LSTM is more suitable for vibration sequence modeling because it alleviates gradient vanishing during long-sequence backpropagation. Compared with a pure convolutional temporal encoder, it explicitly models sequential state transitions; compared with Transformer-based temporal encoders, it is more lightweight and easier to deploy in small-sample industrial diagnosis scenarios. Following the standard LSTM formulation, the input gate, forget gate, output gate, candidate cell state, cell state, and hidden state are calculated as follows:

$$\begin{aligned} &i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i), \quad f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right), \\ &o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o), \quad \tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c), \\ &c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \quad h_t = o_t \odot \tanh(c_t), \end{aligned} \tag{6}$$

where x_t denotes the input feature at time step t, h_{t-1} and c_{t-1} are the hidden state and cell state of the previous time step, respectively; i_t, f_t, and o_t represent the input, forget, and output gates; W_*, U_*, and b_* are trainable weight matrices and bias terms; $\sigma(\cdot)$ is the Sigmoid activation function, $\tanh(\cdot)$ is the hyperbolic tangent function, and $\odot$ denotes element-wise multiplication. Eq. (6) follows the classical LSTM mechanism initially proposed by Hochreiter and Schmidhuber [34].

Although a single LSTM layer can capture temporal dependence to some extent, the vibration response of gearbox-bearing systems contains hierarchical temporal information. Shallow recurrent layers are more sensitive to local impact intervals and short-term fluctuations, whereas deeper recurrent layers can learn more abstract periodic degradation patterns and long-range

correlations. Therefore, this study constructs a Hierarchical Residual-Connected LSTM network by vertically stacking multiple LSTM layers. The output sequence of the l-th LSTM layer is denoted as:

$$\hat{H}^{(l)} = \mathrm{LSTM}^{(l)}\left(H^{(l-1)}\right), \tag{7}$$

where $H^{(0)} = X$, and X denotes the input feature sequence delivered from the preceding DDSC-T module. Here, $\hat{H}^{(l)}$ represents the direct output of the l-th LSTM layer before residual fusion.

However, simply increasing the number of stacked LSTM layers may introduce optimization difficulty and overfitting, especially when the available fault samples are limited. To address this problem, cross-layer residual connections are introduced into the HRC-LSTM module. As shown in Fig. 2, the hidden sequence from a shallower layer is transmitted to a deeper layer through a residual projection path. This path provides an additional information and gradient propagation route, allowing shallow temporal features to be reused by deeper recurrent layers. The residual fusion process is defined as:

$$H^{(l)} = \hat{H}^{(l)} + P_l\left(H^{(l-1)}\right), \quad l = 2,3,\ldots,L, \tag{8}$$

where $P_l(\cdot)$ denotes the residual projection used for dimensional alignment between $H^{(l-1)}$ and $\hat{H}^{(l)}$. In implementation, $P_l(\cdot)$ can be realized by a linear projection or a 1×1 convolution when the feature dimensions of adjacent layers are inconsistent; otherwise, it degenerates into an identity mapping. Inspired by residual learning, this design forms a cross-layer "gradient highway", which helps preserve shallow fault-sensitive information and improves the trainability of the stacked recurrent structure.

The overall architecture of HRC-LSTM is illustrated in Fig. 2. The input sequence X first passes through the first LSTM layer to generate $H^{(1)}$. Then, for each subsequent layer, the output of the current LSTM layer is added to the projected output from the previous layer. The dashed paths in Fig. 2 represent cross-layer residual connections, while the residual projection blocks $P_2, P_3, \ldots, P_L$ ensure that the feature dimensions are compatible before addition. Finally, the output sequence $H^{(L)}$ is obtained and used as the high-level temporal representation for the following attention-based feature enhancement module.

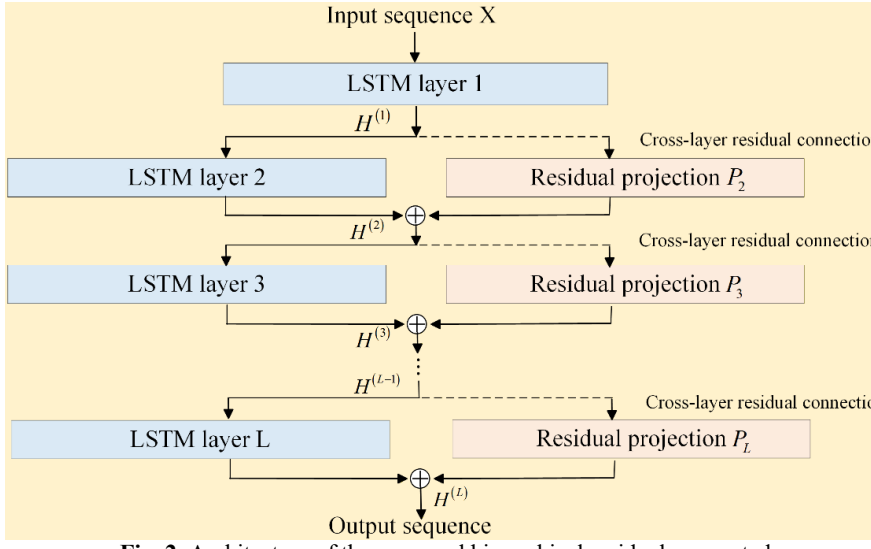


Fig. 2. Architecture of the proposed hierarchical residual-connected long short-term memory (HRC-LSTM) module

The proposed HRC-LSTM is selected instead of a single-layer LSTM, GRU-only structure, or purely convolutional temporal module for three reasons. First, the stacked LSTM structure can represent temporal dependencies at different abstraction levels, which is beneficial for identifying repetitive fault impulses and modulation patterns. Second, the residual connections reduce information loss across layers and improve gradient propagation, making the model more stable under small-sample training conditions. Third, compared with Transformer-style sequence modeling, HRC-LSTM has lower computational overhead and does not require large-scale training data, which is consistent with the lightweight and edge-deployable objective of this study. Therefore, HRC-LSTM provides a balanced solution for sequential fault feature learning by combining temporal memory, hierarchical representation, and efficient residual optimization.

2.3. Hierarchical adaptive attention (HAA)

To further enhance fault-sensitive representations and suppress redundant noise responses under complex working conditions, this study designs a Hierarchical Adaptive Attention (HAA) module. As shown in Fig. 3, HAA contains two sequential attention paths: channel attention and temporal attention. The channel attention path first identifies "which feature channels are more fault-discriminative", while the temporal attention path further determines "which time positions contain more salient impact or modulation information". In this way, HAA performs hierarchical feature recalibration from both the channel and temporal dimensions.

Given the input feature map $F \in \mathbb{R}^{C\times T}$, where C and T denote the number of channels and the temporal length, respectively, the channel attention path first applies temporal global average pooling to each channel. This operation compresses the temporal response of each channel into a compact descriptor, which reflects the average activation intensity of different feature channels within the current observation window. It can be expressed as:

$$z_c = \mathrm{GAP}(F_c) = \frac{1}{T}\sum_{t=1}^{T} F_c(t), \quad c = 1,2,\dots,C. \tag{9}$$

The obtained channel descriptor is then fed into a lightweight bottleneck network consisting of fully connected reduction, ReLU activation, fully connected expansion, and Sigmoid normalization. The channel attention weight vector is calculated as:

$$M_c = \sigma(W_2\delta(W_1 z + b_1) + b_2), \tag{10}$$

where $z = [z_1, z_2, \dots, z_C]$ is the pooled channel descriptor, W_1 and W_2 are the trainable weights of the reduction and expansion layers, b_1 and b_2 are bias terms, $\delta(\cdot)$ denotes the ReLU activation function, and $\sigma(\cdot)$ denotes the Sigmoid function. The generated channel attention M_c adaptively enhances channels containing fault-related resonance or modulation information, while suppressing channels mainly dominated by broadband noise and irrelevant responses. The channel-recalibrated feature is obtained by:

$$F_c = M_c \odot F, \tag{11}$$

where $\odot$ denotes element-wise multiplication with broadcasting along the temporal dimension.

After channel recalibration, the temporal attention path is introduced to locate salient time positions in the feature sequence. This design is necessary because mechanical faults usually appear as periodic impulses, transient shocks, or locally enhanced modulation components, rather than uniformly distributed responses over the whole time window. Following the structure in Fig. 3, average pooling and maximum pooling are performed along the channel dimension of F_c, respectively:

$$F_{avg}^{t} = AvgPool_c(F_c), \quad F_{max}^{t} = MaxPool_c(F_c). \tag{12}$$

The two temporal descriptors are concatenated and then processed by a one-dimensional convolution with a kernel size of 3. The temporal attention map is defined as:

$$M_t = \sigma\left(f_{k=3}^{1D}\left(\left[F_{avg}^{t}; F_{max}^{t}\right]\right)\right), \tag{13}$$

where $[\cdot;\cdot]$ represents the concatenation operation, and $f_{k=3}^{1D}(\cdot)$ denotes a one-dimensional convolution with a kernel size of 3. The average-pooled descriptor reflects the overall temporal activation trend, whereas the max-pooled descriptor highlights locally dominant transient responses. Their combination enables the temporal attention path to capture both stable periodic patterns and abrupt fault-induced impulses.

Finally, the output feature of HAA is obtained by applying the temporal attention map to the channel-recalibrated feature:

$$F_{HAA} = M_t \odot F_c. \tag{14}$$

Therefore, HAA constructs a hierarchical soft recalibration mechanism over the feature map. The channel attention path emphasizes fault-sensitive feature channels, and the temporal attention path further strengthens discriminative time locations. Compared with using only channel attention, only temporal attention, or a heavier self-attention mechanism, HAA provides a more compact and interpretable solution for vibration fault diagnosis. It enhances discriminative fault features while maintaining low computational overhead, which is consistent with the lightweight design objective of the proposed model.

The structure of HAA is illustrated in Fig. 3. The input feature F is first passed through the channel attention branch to generate M_c, and the resulting channel-recalibrated feature F_c is then delivered to the temporal attention branch. In the temporal branch, average pooling and maximum pooling along the channel dimension are concatenated and processed by a $(k = 3)$ one-dimensional convolution to generate M_t. The final output F_{HAA} is produced through temporal recalibration of F_c. This sequential design allows the proposed network to progressively refine fault-related representations from "channel importance" to "temporal saliency".

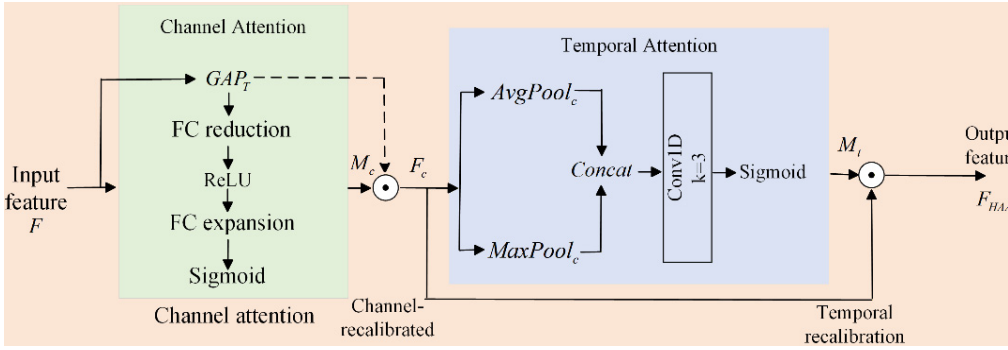


Fig. 3. Architecture of the proposed hierarchical adaptive attention (HAA) module

2.4. Multi-branch feature fusion and classification

To obtain a robust and complementary representation for gearbox-bearing fault diagnosis, this study adopts a multi-branch feature fusion and classification strategy. As described in Sections 2.1-2.3, the main diagnostic branch follows the sequence of DDSC-T, HRC-LSTM, and HAA. Specifically, DDSC-T first extracts adaptive multi-scale local temporal features, HRC-LSTM then models hierarchical sequential dependencies among recurrent fault impulses, and HAA finally

recalibrates the learned features from the channel and temporal dimensions. This progressive structure forms a complete feature-learning path from local impact extraction to long-range temporal modeling and attention-based noise suppression.

Let the output feature of the main branch be denoted as F_{HAA}. In parallel, an auxiliary convolutional neural network (CNN) branch is introduced to extract complementary global texture and shallow morphological information from the input vibration sequence. The auxiliary branch is not intended to replace the main branch, but to preserve additional low-level patterns that may be weakened during recurrent temporal abstraction. The output of the auxiliary branch is denoted as F_{CNN}. Before fusion, the two feature maps are aligned to a compatible temporal resolution and then concatenated along the channel dimension:

$$F_{cat} = [F_{HAA}; F_{CNN}], \tag{15}$$

where $[\cdot;\cdot]$ denotes the channel-wise concatenation operation. This operation preserves the heterogeneous information learned by the two branches: the main branch contributes dynamically selected temporal and sequential fault features, whereas the auxiliary branch provides complementary convolutional texture features.

After concatenation, a lightweight adaptive fusion layer is used to recombine the heterogeneous features. It should be noted that a 1×1 convolution contains trainable parameters; therefore, it is described here as a lightweight learnable fusion operation rather than a parameter-free operation. The fused feature is computed as:

$$F_{fus} = \phi\left(W_{fus} * F_{cat} + b_{fus}\right), \tag{16}$$

where W_{fus} and b_{fus} are the weight and bias of the 1×1 convolutional fusion layer, $*$ denotes convolution, and $\phi(\cdot)$ is the nonlinear activation function. The role of this fusion layer is to learn the relative contribution of different feature channels from both branches in a data-driven manner. Compared with manually assigning fixed branch weights, the learnable 1×1 fusion can adaptively emphasize discriminative features and suppress redundant or noisy responses.

The fused feature is then compressed by Global Average Pooling (GAP) along the temporal dimension to obtain a compact global representation:

$$g = GAP_T\left(F_{fus}\right). \tag{17}$$

GAP reduces the dependence on the exact temporal length of the input sequence and summarizes the overall activation statistics of the fused representation. This is beneficial for improving robustness when vibration windows contain slight phase shifts or local impulse-position variations.

Finally, the global feature vector is mapped to the fault-category probability distribution through a fully connected classification layer and a Softmax function:

$$\hat{y} = \mathrm{Softmax}(W_o g + b_o), \tag{18}$$

where $\hat{y}$ denotes the predicted probability distribution over all fault categories, and W_o and b_o are the weight matrix and bias term of the output classification layer, respectively. The predicted class is obtained according to the maximum posterior probability:

$$\hat{c} = \arg\max_j \hat{y}_j. \tag{19}$$

Overall, the proposed fusion strategy integrates the strengths of the three core modules and the auxiliary CNN branch. DDSC-T improves adaptability to frequency-band shifts by using dynamically weighted multi-scale temporal convolution; HRC-LSTM captures periodic and

long-range temporal dependencies through hierarchical recurrent modeling; HAA enhances fault-sensitive channels and salient time positions while suppressing background noise; and the auxiliary CNN branch supplements shallow texture information. Their combination provides a more comprehensive representation than using DDSC-T, HRC-LSTM, HAA, or CNN alone, which also explains the performance gain observed in the ablation experiments.

From a signal-processing perspective, the fusion architecture jointly considers local transient impulses, periodic fault responses, resonance-band variations, and noise-contaminated temporal saliency. From a feature-learning perspective, it combines convolutional local pattern extraction, recurrent sequential representation, attention-based recalibration, and cross-branch adaptive recombination. Therefore, the final classifier receives a compact but discriminative feature vector for fault identification under small-sample, noisy, and variable-condition scenarios.

3. Intelligent diagnostic process and framework

To establish an end-to-end mapping from raw vibration measurements to fault-category probabilities, this section describes the complete diagnostic process based on the proposed DDSC-T-HRC-LSTM-HAA framework. As illustrated in Fig. 4, the complete process consists of five stages: data input, signal preprocessing, diagnostic-model inference, diagnostic output, and performance evaluation. The first two stages convert the raw vibration records into standardized fixed-length segments, the third stage extracts and fuses fault-sensitive features, and the fourth stage produces the fault probabilities and predicted class. The fifth stage is used for offline evaluation of the trained model rather than for the online inference procedure.

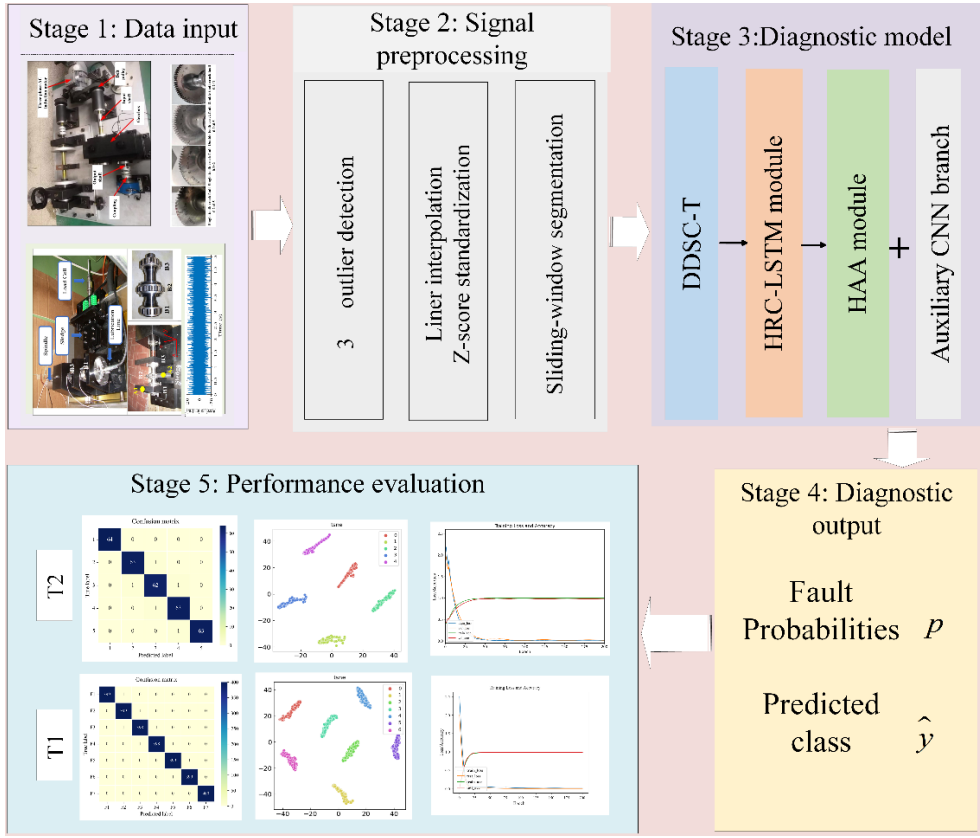


Fig. 4. Overall flowchart of the proposed intelligent diagnosis process

The input data are obtained from the vibration records of the T1 and T2 datasets and contain different health and fault conditions. After preprocessing, each vibration segment (X) is simultaneously delivered to the main branch and the auxiliary CNN branch. The main branch sequentially integrates the DDSC-T, HRC-LSTM, and HAA modules, whereas the auxiliary branch extracts complementary convolutional features. Their outputs are adaptively fused and finally mapped to the fault-probability vector (P) by the classification layer. The framework is implemented in Python using the PyTorch deep-learning framework; the detailed hardware and training settings are provided in Section 4.1.

During model development, the preprocessing parameters estimated from the training subset are retained and applied consistently to the validation and test subsets. This procedure avoids using information from the test set during model construction and ensures that the reported performance reflects the generalization ability of the trained diagnostic framework.

3.1. Vibration signal preprocessing and distribution alignment

To improve signal quality and facilitate stable model optimization, the raw one-dimensional vibration signals are processed through a fixed preprocessing pipeline. The pipeline includes abnormal-sample detection, local signal restoration, leakage-controlled dataset partitioning, and Z-score standardization. These operations are performed before the vibration sequences are segmented into model inputs.

First, possible abnormal observations caused by sensor interference, acquisition disturbances, or temporary signal loss are detected using the Pauta criterion, also referred to as the three-sigma rule. For a vibration sequence $r(t)$ with mean μ_r and standard deviation σ_r, a sample is marked as an outlier when:

$$|r(t) - \mu_r| > 3\sigma_r. \tag{20}$$

The detected abnormal points are repaired by local linear interpolation using the nearest valid samples on both sides. This operation is intended to reduce isolated acquisition artifacts while preserving the local trend and continuity of the vibration waveform. It does not generate additional fault samples or alter the fault labels.

After outlier restoration, the data are divided into training, validation, and test subsets at the original recording or experiment-run level before sliding-window segmentation. Samples originating from the same continuous recording are kept within the same subset whenever possible, thereby preventing highly overlapping windows from appearing simultaneously in different subsets. A stratified sampling strategy without replacement is used to approximately preserve the class distribution of the three subsets. This partitioning procedure is particularly important for vibration signals because adjacent samples in a continuous recording are strongly correlated.

Subsequently, Z-score standardization is applied to reduce the influence of physical scale differences among signal channels and operating conditions. For a sample $x(t)$, the standardized value is calculated as:

$$x^{std}(t) = \frac{x(t) - \mu_{tr}}{\sigma_{tr}}, \tag{21}$$

where μ_{tr} and σ_{tr} are the mean and standard deviation estimated from the training subset. The same training-derived parameters are then applied to the validation and test signals, rather than recalculating the statistics separately for each subset. This arrangement prevents information leakage and provides a consistent feature scale during training and inference.

The preprocessing procedure therefore preserves the essential temporal characteristics of fault vibration while reducing the effect of isolated measurement errors and amplitude-scale variation.

The standardized signals are subsequently passed to the dynamic observation-window construction stage.

3.2. Dynamic observation window construction

A sliding window is used to segment the preprocessed continuous vibration signal into fixed-length temporal segments. Let $x^{std}(t)$ denote the standardized signal, N the number of sampling points in one window, and S the sliding stride. The i-th input segment is defined as:

$$X_i = [x^{std}(iS), x^{std}(iS+1), \ldots, x^{std}(iS+N-1)]. \tag{22}$$

Each segment inherits the label of its original recording or operating condition. The value of N determines the temporal context available to the diagnostic model and directly affects both the observable fault-response pattern and the computational latency of edge inference. A window that is too short may truncate periodic impacts or omit the local resonance response, whereas an excessively long window increases memory use and inference time without necessarily providing a proportional accuracy improvement.

For the T2 dataset, the sampling frequency is $f_s = 12.8$ kHz and the rotational speed is 1500 r/min. The observation time corresponding to a window of N points is:

$$T_N = \frac{N}{f_s}. \tag{23}$$

Thus, 300, 400, and 500 sampling points correspond to 23.44, 31.25, and 39.06 ms, respectively. Under the same data-partitioning, preprocessing, training, and inference settings, the three window lengths were compared to evaluate the trade-off between diagnostic performance and computational cost, as summarized in Table 1.

Table 1. Comparison of diagnostic performance and computational overhead for different sampling-window lengths

Sampling points (N)	Observation time window (ms)	Diagnostic accuracy (%)	Single-inference latency (ms)	Interpretation
300	23.44	92.4	7.6	The temporal context is insufficient to represent some periodic and resonance-related fault responses
400	31.25	98.3	9.8	Provides a suitable balance between temporal information and inference latency
500	39.06	98.4	13.5	Provides only a marginal accuracy gain while introducing considerable computational overhead

As shown in Table 1, the 300-point window leads to a substantial performance decrease because the available temporal context is insufficient to represent the complete local evolution of some fault responses. Increasing the window length to 400 points improves the accuracy to 98.3 % with a single-inference latency of 9.8 ms. Although the 500-point window achieves a slightly higher accuracy of 98.4 %, the improvement is only 0.1 percentage point, while the latency increases from 9.8 to 13.5 ms, corresponding to an increase of approximately 37.8 %.

Therefore, $N = 400$ is selected as the default observation-window length. This selection is not based solely on the highest accuracy, but on the joint consideration of temporal-context completeness, diagnostic performance, and real-time deployment requirements. The dynamic-kernel and HAA configurations are fixed according to the model settings reported in Section 4.1 and are not tuned using the test set. Table 1 provides the window-length sensitivity analysis

available in this study; a complete factorial sensitivity analysis of every convolution-kernel and attention parameter is beyond the present experimental scope and is acknowledged as a possible direction for further investigation.

3.3. Overall fault-diagnosis execution steps

After preprocessing and window construction, each input segment (X) is processed by the proposed framework. According to the model architecture shown in Fig. 5, the inference procedure can be summarized into four sequential operations, followed by an independent performance-evaluation stage.

Step 1: Adaptive local temporal-feature extraction.

The input segment (X) is first delivered to the DDSC-T module. In each parallel branch, depth-wise Conv1D extracts channel-specific local temporal patterns using a branch-specific kernel (K_i), and point-wise Conv1D performs cross-channel feature projection. The GAP-fully connected network shown in Fig. 1 generates the branch weights ($\alpha_1, \alpha_2, \ldots, \alpha_M$) from the current input feature. These weights modulate the outputs of the parallel branches rather than directly replacing the convolution kernels. The weighted branch outputs are then aggregated and average-pooled to obtain F_{DDSC-T}. This mechanism enables the model to select an appropriate temporal receptive field for transient impacts and resonance-related patterns under different signal conditions.

Step 2: Hierarchical temporal-dependency modeling.

The feature sequence (F_{DDSC-T}) is provided to the HRC-LSTM module. The stacked LSTM layers model the temporal evolution of the extracted features, while the projection paths (P_l) establish cross-layer residual connections. The residual connections allow shallow temporal information to be reused by deeper recurrent layers and provide an additional route for gradient propagation. Consequently, HRC-LSTM can learn both short-term impact relationships and relatively long-range dependencies within the selected observation window without relying on an excessively deep recurrent structure.

Step 3: Channel-temporal feature recalibration.

The final hidden sequence ($H^{(L)}$) from HRC-LSTM is then input into the HAA module. The channel-attention path first applies temporal global average pooling, followed by fully connected reduction, ReLU activation, fully connected expansion, and Sigmoid normalization to generate the channel mask (M_c). The channel-recalibrated feature is subsequently processed by the temporal-attention path. Average pooling and maximum pooling are performed along the channel dimension, and the two descriptors are concatenated and passed through a ($k = 3$) Conv1D layer and a Sigmoid function to generate the temporal mask (M_t).

The final main-branch representation is therefore:

$$F_m = F_{HAA} = M_t \odot \left(M_c \odot H^{(L)}\right), \tag{24}$$

where the multiplication operations are broadcast along the compatible feature dimensions. This operation forms a joint channel-temporal soft mask. When the learned channels encode multi-scale resonance-related responses, the mask can also be interpreted as a frequency-sensitive-temporal recalibration mechanism. It should not be understood as an explicit Fourier-domain filter; rather, it is a learned feature-domain weighting operation.

Step 4: Cross-branch fusion and classification.

In parallel with the main branch, the original input segment (X) is processed by the lightweight CNN auxiliary branch to obtain complementary feature representation (F_a). As shown in Fig. 5, the main-branch and auxiliary-branch features are first aligned to compatible dimensions and concatenated along the channel dimension:

$$F_{cat} = [F_m; F_a]. \tag{25}$$

The concatenated feature is then passed through a learnable 1×1 convolution for adaptive channel recombination:

$$F_{fus} = \phi\left(W_{fus} * F_{cat} + b_{fus}\right), \tag{26}$$

where W_{fus} and b_{fus} denote the trainable fusion weights and bias, respectively. Because the 1×1 convolution contains trainable parameters, this operation is referred to as lightweight adaptive feature fusion rather than parameter-free dynamic soft attention. Finally, Global Average Pooling compresses F_{fus} along the temporal dimension, and the resulting vector is mapped to the fault-probability distribution:

$$P = \mathrm{Softmax}\left(W_o GAP_T\left(F_{fus}\right) + b_o\right), \tag{27}$$

where P is the probability vector of all fault categories. The predicted fault class is obtained by:

$$\hat{y} = \arg\max_j P_j. \tag{28}$$

Step 5: Performance evaluation.

The predicted labels and probability vectors are retained for subsequent evaluation. As illustrated in Fig. 4, the evaluation stage includes the analysis of confusion matrices, t-SNE feature distributions, and training loss–accuracy curves for the T1 and T2 datasets. The confusion matrix is used to identify class-specific recognition errors, t-SNE is used to visualize the separability of learned features, and the training curves are used to examine convergence behavior and possible overfitting. These visualizations complement quantitative indicators such as accuracy, recall, precision, Macro-F1, and inference latency, but t-SNE visualization itself is not treated as an independent classification metric.

Overall, the proposed framework establishes a progressive information-processing chain: DDSC-T extracts adaptive multi-scale local temporal patterns, HRC-LSTM models their hierarchical sequential dependencies, HAA suppresses irrelevant channel and temporal responses, and the auxiliary CNN branch preserves complementary local texture information. The adaptive 1×1 fusion layer then recombines these heterogeneous representations before classification. This division of functional roles explains how the framework integrates local signal processing, temporal feature learning, attention-based recalibration, and lightweight edge-oriented inference in a single diagnostic pipeline.

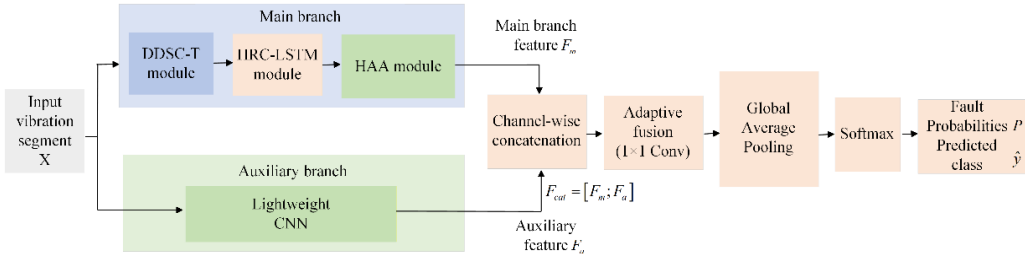


Fig. 5. Overall architecture of the proposed DDSC-T-HRC-LSTM-HAA framework with the auxiliary CNN branch

4. Experimental setup and datasets

This section presents the experimental setup used to evaluate the proposed DDSC-T-HRC-LSTM-HAA diagnostic framework. To respond more clearly to the requirements of reproducibility, fair comparison, and cross-condition generalization, the experimental description is reorganized into six parts: implementation environment, hyperparameter configuration,

evaluation protocol, T1 dataset and cross-condition validation, T2 dataset, and baseline-model comparison.

Specifically, Section 4.1 reports the computing environment, programming framework, and implementation procedure. Section 4.2 explains the main hyperparameter settings and their selection rationale. Section 4.3 defines the evaluation metrics and statistical protocol used to assess the robustness of the reported results. Section 4.4 introduces the Turin Polytechnic University rolling-bearing dataset and the newly added cross-condition generalization protocol based on unused speed–load working conditions. Section 4.5 describes the QPZZ-II rotating-machinery dataset. Section 4.6 summarizes the baseline models and the fair-comparison strategy.

The overall objective of this experimental design is not only to report classification accuracy under standard test conditions, but also to examine whether the proposed model can maintain stable diagnostic performance under small-sample settings, noisy environments, and operating-condition shifts. In particular, the additional cross-condition protocol on T1 is designed to evaluate the generalization ability of the model without introducing artificial or manually synthesized fault categories.

4.1. Experimental environment and implementation details

All diagnostic models were implemented in Python using the PyTorch deep-learning framework. The complete experimental pipeline includes vibration-signal loading, abnormal-point processing, Z-score standardization, sliding-window segmentation, model training, validation, testing, and result visualization. The implementation was organized as modular scripts so that the proposed DDSC-T, HRC-LSTM, HAA, auxiliary CNN branch, and baseline networks could be trained and evaluated under the same data-processing and optimization pipeline.

The training experiments were conducted on a workstation equipped with an Intel Core i9 CPU and an NVIDIA RTX 2080Ti GPU. The GPU was used for model training and acceleration of tensor operations, while the CPU was mainly used for data preprocessing, dataset partitioning, and result statistics. To evaluate practical deployment potential, edge-inference tests were further conducted on an NVIDIA Jetson Xavier NX platform. The Jetson Xavier NX was used only for inference-time evaluation, whereas model training was performed on the workstation.

For each dataset, the raw vibration signals were first preprocessed according to the procedure described in Section 3.1. The training-set statistics were used for Z-score standardization, and the same parameters were applied to the validation and test sets to avoid information leakage. The segmented samples were then fed into the diagnostic models in mini-batches during training. During inference, each fixed-length vibration segment was independently mapped to a fault-probability vector and a predicted fault class.

The proposed network consists of a main branch and an auxiliary branch. In the main branch, DDSC-T extracts adaptive multi-scale local temporal features, HRC-LSTM models hierarchical temporal dependencies, and HAA performs channel-temporal feature recalibration. In parallel, the auxiliary CNN branch extracts complementary shallow convolutional features. The outputs of the two branches are concatenated and recombined through a lightweight learnable 1×1 convolutional fusion layer, followed by Global Average Pooling and Softmax classification. This implementation is consistent with the network architecture described in Sections 2.1-2.4 and Fig. 5.

For a fair comparison, all baseline models were implemented or reproduced under the same PyTorch environment and were trained using the same input samples, data splits, preprocessing strategy, and evaluation metrics as the proposed model. No test-set information was used during model training, hyperparameter selection, or feature standardization. Unless otherwise stated, the same training configuration was maintained across different models to ensure that performance differences mainly reflected the model architectures rather than inconsistent experimental settings.

The main experimental environment is summarized in Table 2.

Table 2. Experimental environment and implementation settings

Item	Setting
Programming language	Python
Deep-learning framework	PyTorch
Training CPU	Intel Core i9
Training GPU	NVIDIA RTX 2080Ti
Edge-inference platform	NVIDIA Jetson Xavier NX
Input type	One-dimensional vibration segments
Main model branch	DDSC-T + HRC-LSTM + HAA
Auxiliary branch	Lightweight CNN branch
Fusion operation	Channel-wise concatenation + learnable 1×1 convolution
Classification layer	Global Average Pooling + fully connected layer + Softmax
Main output	Fault-category probability vector and predicted class
Implementation principle	Same preprocessing, data split, and evaluation protocol for all compared models

In the edge-inference evaluation, the trained model was deployed on the Jetson Xavier NX to measure the single-segment inference latency under the selected input-window length. This setting was used to verify whether the proposed lightweight architecture can satisfy real-time diagnostic requirements on embedded hardware. The detailed hyperparameter configuration, including the input-window length, batch size, optimizer, training epochs, and learning-rate strategy, is provided in Section 4.2.

4.2. Hyperparameter configuration and selection rationale

The hyperparameters were configured according to three considerations: preservation of fault-related temporal information, stable optimization under limited training samples, and computational efficiency for edge deployment. Dataset-independent training parameters were kept unchanged across operating conditions, whereas dataset-dependent segmentation parameters were selected according to the corresponding sampling frequency and observation-window analysis. No target-condition test sample was involved in hyperparameter selection.

The maximum number of training epochs was set to 200, the mini-batch size was 128, and Adam was employed as the optimizer. A common training budget of 200 epochs was adopted to provide sufficient convergence time while maintaining the same optimization budget for comparative models. The batch size of 128 offers a practical compromise between gradient stability, GPU-memory consumption, and training efficiency. Adam was selected because its adaptive first- and second-moment estimates are suitable for jointly optimizing the convolutional, recurrent, and attention-based components of the proposed network.

Instead of maintaining a constant learning rate throughout training, a Cyclical Learning Rate (CLR) policy was employed. The learning rate was periodically varied within a predefined interval during optimization, enabling the model to explore a wider parameter region at the beginning of each cycle and perform finer updates at lower learning-rate phases. The same learning-rate policy was maintained for repeated runs of a given model, and its parameters were not adjusted according to test-set performance. This strategy reduces dependence on a single manually selected constant learning rate and improves optimization stability under the small-sample setting.

The input-window length was treated as a key dataset-dependent parameter because it jointly determines temporal coverage and online latency. For T2, candidate lengths of 300, 400, and 500 points were evaluated under the same training and inference conditions, as reported in Table 1. A 300-point window achieved only 92.4 % accuracy because its temporal coverage was insufficient, whereas a 500-point window increased the accuracy from 98.3 % to only 98.4 % but increased the latency from 9.8 to 13.5 ms. Therefore, the 400-point window was retained because it provided the best accuracy-latency trade-off. The segmentation and condition-partition settings used for T1

are specified separately in Section 4.4 to account for its sampling frequency and cross-condition protocol.

The architecture-related settings follow the functional designs introduced in Sections 2.1-2.4. DDSC-T employs multiple parallel depth-wise temporal-convolution branches with branch-specific kernel sizes $(K_1, K_2, \ldots, K_M)$. Their outputs are modulated by input-dependent weights generated through GAP, two fully connected layers, ReLU, and Sigmoid, and are subsequently aggregated. The branch configuration is held fixed across source and target operating conditions; hence, no kernel is selected using target-condition labels. The point-wise convolution uses a 1×1 kernel to mix channel information without enlarging the temporal receptive field.

HRC-LSTM vertically stacks recurrent layers and introduces residual projections (P_l) between adjacent hierarchical representations. An identity mapping is used when the input and output dimensions agree; otherwise, a learnable projection performs dimensional alignment. The recurrent depth and hidden dimensions are kept unchanged across all operating conditions within the same experimental protocol, ensuring that cross-condition results are not obtained by enlarging the model for the target domain.

Table 3. Main hyperparameter configuration and selection rationale

Category	Configuration	Selection rationale
Training epochs	200	Provides a common and sufficient optimization budget for all experimental iterations
Mini-batch size	128	Strikes an optimal balance among gradient stability, GPU memory consumption, and overall training efficiency
Optimizer	Adam	Facilitates the stable joint optimization of the convolutional, recurrent, and attention modules
Learning-rate policy	Cyclical learning rate (CLR)	Mitigates reliance on a fixed learning rate, promoting robust exploration of the parameter space
T2 input-window length	400 sampling points	Determined based on the optimal accuracy-latency trade-off (cf. Table 1 for the 300/400/500-point ablation study)
DDSC-T temporal extraction	Parallel depth-wise branches with diverse kernels (K1, K2, ..., KM)	Captures multiscale temporal patterns across varying receptive fields; configurations remain invariant across diverse operating conditions
DDSC-T dynamic weighting	GAP-FC-ReLU-FC-Sigmoid	Generates adaptive, input-dependent branch weights with marginal computational overhead
Point-wise projection	1×1 Conv1D	Executes cross-channel feature mixing while strictly preserving temporal resolution
HRC-LSTM structure	Stacked LSTM layers with residual projections (Pl)	Integrates hierarchical temporal modeling to enhance both information flow and gradient propagation
HAA channel path	GAP-FC reduction-ReLU-FC expansion-Sigmoid	Adaptively recalibrates fault-sensitive channels while suppressing redundant feature responses
HAA temporal path	AvgPool/MaxPool-Concat-Conv1D ($k = 3$)-Sigmoid	Efficiently extracts local temporal saliency with minimal computational burden
Cross-branch fusion	Channel-wise concatenation followed by a learnable 1×1 convolution	Adaptively recombines heterogeneous features extracted from the main and auxiliary branches
Output layer	GAP–Fully Connected Layer-Softmax	Yields a compact global representation to output the final fault-class probability distribution

For HAA, the channel-attention path adopts temporal GAP followed by fully connected reduction, ReLU, fully connected expansion, and Sigmoid normalization. The temporal-attention path combines channel-wise average and maximum pooling and applies a Conv1D layer with a kernel size of 3. The value $(k = 3)$ was selected as the smallest odd kernel that captures the immediate temporal neighborhood while preserving the sequence length and introducing limited

computational overhead. Finally, the main-branch and auxiliary-branch features are concatenated and adaptively recombined by a learnable 1×1 convolution before GAP and Softmax classification.

The principal training and architecture configurations are summarized in Table 3.

A dedicated sensitivity comparison was conducted for the observation-window length because it directly affects both diagnostic information and inference latency. By contrast, an exhaustive grid search over all dynamic-kernel, recurrent, and attention configurations was not performed. These architecture settings were fixed before cross-condition testing and were evaluated through the module-level ablation experiments in Section 5.2. This protocol limits the risk of overfitting the architecture to a particular test condition while keeping the computational cost of model selection compatible with the lightweight objective of the study.

4.3. Evaluation metrics and statistical protocol

The models were evaluated in terms of classification performance, prediction-probability error, and computational efficiency. Let TP_c, TN_c, FP_c, and FN_c denote the true-positive, true-negative, false-positive, and false-negative predictions for class c, respectively. Accuracy, precision, recall, and F1-score are defined as:

$$\begin{aligned} \text{Accuracy} &= \frac{\sum_{c=1}^{C} TP_c}{N}, \quad \text{Precision}_c = \frac{TP_c}{TP_c + FP_c}, \\ \text{Recall}_c &= \frac{TP_c}{TP_c + FN_c}, \quad \text{F1}_c = \frac{2\,\text{Precision}_c\,\text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}, \end{aligned} \tag{29}$$

where N and C are the numbers of test samples and fault categories, respectively. Precision, recall, and F1-score were macro-averaged across all categories so that each fault class contributed equally to the final result. Accuracy and Macro-F1 were used as the primary metrics, while per-class recall and confusion matrices were examined to identify recognition failures involving weak defects and cross-condition shifts.

Because the present problem is a nominal multi-class classification task, directly calculating Root Mean Square Error (RMSE) from integer class labels would impose an artificial numerical distance between fault categories. Therefore, probability-based RMSE (P-RMSE) was adopted as a supplementary metric:

$$P\text{-RMSE} = \sqrt{\frac{1}{NC}\sum_{n=1}^{N}\sum_{c=1}^{C}\left(\hat{p}_{n,c} - y_{n,c}\right)^2}, \tag{30}$$

where $\hat{p}_{n,c}$ is the predicted probability for class c, and $y_{n,c}$ is the corresponding one-hot target. A lower P-RMSE indicates that the predicted probability distribution is closer to the ground truth.

Computational efficiency was evaluated using the number of trainable parameters, floating-point operations (FLOPs), peak memory consumption, and single-segment inference latency. All latency comparisons were conducted on the NVIDIA Jetson Xavier NX using the same input length and execution settings. Model loading and data-reading time were excluded from the latency measurement.

To reduce the influence of random initialization and mini-batch ordering, each principal comparison was independently repeated ten times using a shared set of random seeds. Results are reported as mean ± standard deviation (SD). Statistical differences between the proposed method and each baseline were evaluated using a two-sided paired Wilcoxon signed-rank test on the run-level Accuracy and Macro-F1 results. Holm-Bonferroni correction was applied when multiple baselines were compared simultaneously, and an adjusted $p < 0.05$ was considered statistically

significant.

For the cross-condition experiment, models were trained exclusively on the designated source conditions and directly evaluated on previously unseen target speed-load conditions. Target-condition samples were not used for training, hyperparameter selection, or standardization-parameter estimation. The results of individual target conditions were macro-averaged to prevent conditions containing more segmented samples from dominating the overall generalization result.

Table 4 summarizes the principal evaluation criteria.

Table 4. Evaluation criteria used in this study

Evaluation aspect	Metrics	Reporting form
Classification performance	Accuracy, macro-precision, macro-recall, and macro-F1 score	Mean ± SD (%)
Probability-level error	Probabilistic root mean square error (P-RMSE)	Mean ± SD
Diagnostic failure analysis	Per-class recall and confusion matrix	Class-Wise Results
Computational efficiency	Model parameters, FLOPs, memory footprint, and inference latency	Millions (M), GFLOPs, MB, and ms
Statistical significance	Paired Wilcoxon signed-rank test with Holm-Bonferroni correction	Adjusted p-value

4.4. T1 dataset and cross-condition generalization protocol

The T1 dataset was collected using the public rolling-bearing test rig developed by Politecnico di Torino. The experimental platform consists of a high-speed spindle, three supporting roller bearings (B1-B3), a loading mechanism, and two triaxial accelerometers installed at positions A1 and A2. The six synchronized acceleration channels measure the axial and two radial vibration components at the two sensor positions (the details could be referred to Fig. 6). The signals were acquired at a sampling frequency of 51.2 kHz. Because the spindle was controlled using the nominal inverter setting without closed-loop tachometer feedback, the speed values in the dataset represent nominal rotational frequencies; the actual speed can be slightly lower, particularly under increased load.

Artificial defects were introduced into the bearing installed at position B1 by producing conical indentations on either the inner race or a single roller. The dataset contains one healthy condition and six fault conditions with three defect sizes. The original dataset labels (0A)-(6A) were mapped to the diagnostic labels F1-F7 used in this study (the corresponding time-domain waveforms are shown in Fig. 7), as summarized in Table 5.

Table 5. Fault categories used in the T1 dataset

Diagnostic label	Original label	Bearing health state	Defect size (μm)
F1	0A	Normal (Baseline)	–
F2	1A	Inner-race fault	450
F3	2A	Inner-race fault	250
F4	3A	Inner-race fault	150
F5	4A	Roller fault	450
F6	5A	Roller fault	250
F7	6A	Roller fault	150

For the original within-condition diagnosis experiment, the nominal rotational frequency was 100 Hz (6000 r/min) under the unloaded condition. A total of 500 non-overlapping samples were retained for each category, of which 100 were used for training and 400 for testing. Additive White Gaussian Noise (AWGN) with a randomly sampled Signal-to-Noise Ratio (SNR) between –5 and 5 dB was applied only to the training samples. The augmented samples increased the training set to 200 samples per category, while the test data remained unaltered.

To provide stronger evidence of operating-condition generalization without introducing a new fault-label system, previously unused speed-load records from T1 were employed to construct an additional direct-transfer experiment. The model was trained only at the source condition of 100 Hz and 0 N. It was then evaluated without fine-tuning under three unseen target conditions selected to isolate different forms of domain shift: 300 Hz and 0 N for a speed-only shift, 100 Hz and 1400 N for a load-only shift, and 300 Hz and 1400 N for a combined speed-load shift. These conditions are available for all seven bearing states and use the same sensor arrangement, enabling the effect of operating-condition variation to be evaluated without changing the fault categories or measurement channels.

For this additional experiment, each six-channel signal was divided into non-overlapping windows of 512 sampling points. At 51.2 kHz, each window covers 10 ms, corresponding to one nominal shaft revolution at the 100-Hz source condition. The same window length was retained for all target conditions so that the input dimensionality and model architecture remained unchanged. One hundred clean source windows per category were used for training, and the source-only AWGN augmentation increased this number to 200 per category. A fixed set of 400 clean windows per category was retained for each target condition. Windows were sampled without replacement, and identical source and target samples were supplied to all compared models.

Table 6. Cross-condition generalization protocol on T1

Domain	Nominal speed	Nominal load	Condition shift scenario	Samples per class	Experimental phase
Source (S)	100 Hz (6,000 r/min)	0 N	Model training	100 clean + 100 AWGN-augmented	Training only
Target (TS)	300 Hz (18,000 r/min)	0 N	Speed shift	400 clean	Testing only
Target (TL)	100 Hz (6,000 r/min)	1,400 N	Load shift	400 clean	Testing only
Target (TSL)	300 Hz (18,000 r/min)	1,400 N	Combined speed-load shift	400 clean	Testing only

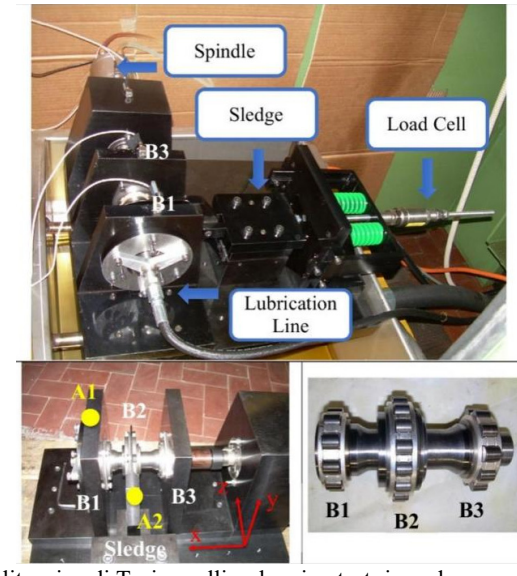


Fig. 6. Politecnico di Torino rolling-bearing test rig and sensor arrangement

All normalization statistics were estimated exclusively from the clean source-domain training samples and then applied unchanged to the target domains. No target-condition label, sample, or

statistical parameter was used for model training, hyperparameter selection, early stopping, or feature normalization. Consequently, this experiment evaluates direct cross-condition generalization rather than target-domain adaptation.

To keep the additional experiment computationally economical, only the proposed DDSC-T-HRC-LSTM-HAA model and three representative baselines – WDCNN, CNN-GRU, and ResNet-CBAM – were evaluated. The target conditions were reported separately to distinguish speed-induced, load-induced, and combined degradation. Their condition-level Accuracy and Macro-F1 values were also averaged with equal weights to obtain an overall cross-condition score. For every independent run, the same source samples, target samples, and random seed were shared by the four models, following the statistical protocol in Section 4.3.

This protocol differs from a random within-condition split because the source and target samples originate from physically distinct operating conditions. It therefore provides a stricter test of whether the learned fault representation remains discriminative when rotational frequency, vibration amplitude, and load-dependent resonance characteristics change simultaneously.

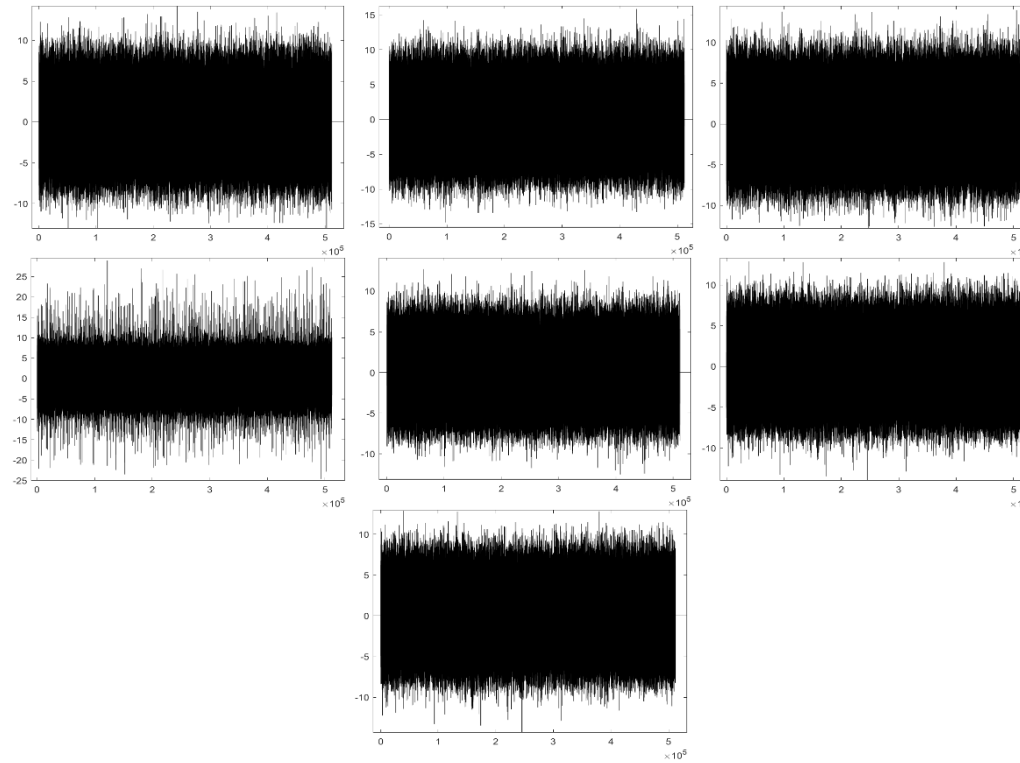


Fig. 7. Representative time-domain vibration signals for the seven T1 bearing conditions

4.5. T2: QPZZ-II rotating-machinery dataset

The T2 dataset was collected using the QPZZ-II rotating-machinery vibration test bench in our laboratory. As shown in Fig. 8 (the Fig. 8 was taken by H. C. Wang in the laboratory of Zhengzhou University of Light Industry, On May.18, 2025), the test system contains a gearbox, driving motor, loading device, coupling, and vibration-acquisition system. The gearbox consists of a 55-tooth pinion and a 75-tooth gear, both with a module of 2. During data acquisition, the motor operated at 1500 r/min, corresponding to a shaft rotational frequency of 25 Hz, and the vibration signals were sampled at 12.8 kHz.

To simulate gear-root damage, cracks with prescribed dimensions were manufactured by

Electrical Discharge Machining (EDM), as illustrated in Fig. 9 (the Fig.9 was taken by H. C. Wang in the laboratory of Zhengzhou University of Light Industry, On May.18, 2025). The dataset contains five health states: normal, two single-tooth crack severities, and two double-tooth crack severities. Compared with the bearing-defect categories in T1, T2 provides a structurally different diagnostic task involving the weak vibration-response variation caused by crack-induced Time-Varying Mesh Stiffness (TVMS) attenuation. The detailed fault configurations are listed in Table 7.

Table 7. Fault categories and sample configuration of the T2 dataset

Label	Fault condition	Nominal Defect Dimensions (mm)	Window Length (Points)	Base Samples (Per Class)	Dataset Split (Train / Val / Test)	Effective Training Samples
1	Normal (Baseline)	–	400	320	224 / 32 / 64	298
2	Single-tooth crack	0.2×0.5	400	320	224 / 32 / 64	298
3	Single-tooth crack	0.2×2.0	400	320	224 / 32 / 64	298
4	Double-tooth crack	0.2×0.5	400	320	224 / 32 / 64	298
5	Double-tooth crack	0.2×1.0	400	320	224 / 32 / 64	298

Each health condition contains 128,000 original sampling points. Based on the window-length analysis in Section 3.2, the signal was first divided into 320 non-overlapping segments of 400 points, corresponding to an observation duration of 31.25 ms per segment. The resulting samples were allocated to mutually exclusive training, validation, and test portions at a ratio of 7:1:2, yielding 224, 32, and 64 base samples per category, respectively.

To moderately increase the number of training samples without contaminating the evaluation data, overlapping segmentation was applied exclusively within the continuous signal portion assigned to the training set. A window length of 400 points and a stride of 300 points were used, corresponding to an overlap ratio of 25 %. This procedure increased the training set from 224 to 298 samples per category. The validation and test sets retained their original 32 and 64 non-overlapping samples, respectively. No sliding window was allowed to cross the boundaries between the training, validation, and test portions.

All preprocessing and normalization parameters were estimated using the training data and then applied unchanged to the validation and test sets. The validation set was used for model configuration and convergence monitoring, whereas the test set was reserved for the final performance evaluation. The same data partitions, input windows, and preprocessing procedure were used for the proposed method and all baseline models. This protocol ensures that the reported differences are attributable primarily to the diagnostic architectures rather than inconsistent sample construction or access to test information.

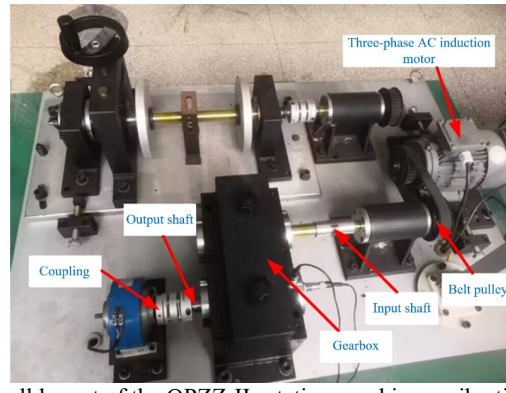


Fig. 8. Overall layout of the QPZZ-II rotating-machinery vibration test bench

Together, T1 and T2 provide complementary evaluation scenarios. T1 focuses on bearing defects of different severities under limited samples, noise interference, and operating-condition shifts, whereas T2 evaluates gear-crack recognition from weak stiffness-related vibration variations. This cross-structural configuration enables the proposed framework to be examined on two distinct rotating-machinery fault mechanisms.

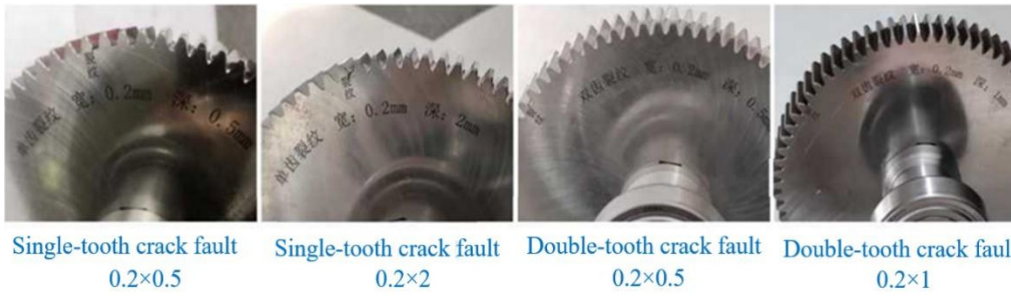


Fig. 9. Gear-root cracks manufactured by electrical discharge machining

4.6. Baseline models and fair-comparison protocol

To objectively evaluate the effectiveness of the proposed DDSC-T-HRC-LSTM-HAA framework, several representative baseline models were selected for comparison. The baseline group covers convolution-based, recurrent-based, attention-enhanced, and Transformer-style diagnostic architectures. For the conventional within-dataset experiments on T1 and T2, six baseline models were considered: CGRU, CNN-GRU, GRDAU, WDCNN, ResNet-CBAM, and 1D-ViT. For the newly added cross-condition generalization experiment on T1, a reduced but representative baseline group was used, including WDCNN, CNN-GRU, and ResNet-CBAM. This reduced setting was adopted to control the additional computational cost while still covering pure convolutional, convolutional-recurrent, and attention-enhanced convolutional architectures.

The selected baselines are summarized in Table 8.

All baseline models were implemented under the same Python/PyTorch environment described in Section 4.1. To ensure a fair comparison, the models used identical training, validation, and test partitions within each experimental task. The same preprocessing procedure, standardization parameters, sample-window length, batch size, optimizer type, maximum training epochs, and evaluation metrics were applied to all compared models unless a model-specific operation was required by its original structure. No baseline was allowed to access target-condition samples during training or hyperparameter selection in the cross-condition experiment.

For input compatibility, each baseline received the same one-dimensional vibration segment as the proposed model. When a baseline architecture required an input or output adaptation, only the first input layer and final classification layer were adjusted to match the number of signal channels and fault categories. The core feature-extraction and sequence-modeling structures of the baseline models were preserved. This rule prevents unfair improvement caused by redesigning a baseline while still allowing all models to process the same experimental data.

For the T1 cross-condition generalization experiment, all four compared models were trained exclusively on the source condition of 100 Hz and 0 N and then directly tested on unseen target speed-load conditions. The source-domain training samples, target-domain test samples, and random seeds were kept identical across WDCNN, CNN-GRU, ResNet-CBAM, and the proposed model. The target-domain results were reported separately for speed-only, load-only, and combined speed-load shifts, and their condition-level scores were further macro-averaged.

For computational comparison, the number of trainable parameters, FLOPs, memory consumption, and single-segment inference latency were measured under the same input length and hardware setting. Edge-inference latency was evaluated on the NVIDIA Jetson Xavier NX using the same timing protocol for all models. This setting ensures that the reported deployment

advantage reflects architectural efficiency rather than differences in device configuration or input preprocessing.

Table 8. Baseline models and their roles in the comparative experiments

Model	Main characteristic	Role in comparison	Within-dataset tests	T1 cross-condition test
CGRU	Convolutional feature extraction integrated with gated recurrent modeling	Recurrent baseline	Yes	No
CNN-GRU	CNN-based local feature extraction cascaded with GRU temporal modeling	Convolutional-recurrent baseline	Yes	Yes
GRDAU	Deep diagnostic architecture integrating gated and attention-based feature learning	Advanced recurrent/attention baseline	Yes	No
WDCNN	Wide-kernel 1D convolutional neural network	Classical vibration-diagnosis CNN baseline	Yes	Yes
ResNet-CBAM	Residual CNN augmented with channel and spatial attention mechanisms	Attention-enhanced convolutional baseline	Yes	Yes
1D-ViT	1D Vision Transformer-style sequence encoder	Global self-attention baseline	Yes	No
Proposed	DDSC-T + HRC-LSTM + HAA with auxiliary CNN fusion	Proposed lightweight hybrid framework	Yes	Yes

The purpose of this comparison protocol is to make the evaluation both broad and practical. The complete baseline set in the standard experiments demonstrates how the proposed framework performs against diverse diagnostic architectures, whereas the reduced baseline set in the cross-condition experiment provides a focused and computationally economical test of generalization against the most relevant lightweight competitors.

5. Results and discussion

This chapter evaluates the proposed DDSC-T-HRC-LSTM-HAA framework from five aspects: overall diagnostic performance, module-level contribution, cross-condition generalization, edge-deployment efficiency, and robustness under strong noise. First, the within-dataset classification results on T1 and T2 are analyzed using training curves, confusion matrices, and t-SNE feature visualization. Then, ablation experiments are used to quantify the individual and synergistic contributions of DDSC-T, HRC-LSTM, and HAA. The cross-condition experiment on T1 further examines the model's generalization ability under unseen speed-load conditions. Finally, computational cost and noise-robustness results are discussed to assess the practical applicability and limitations of the proposed diagnostic framework.

All comparisons in this chapter follow the evaluation protocol described in Section 4.3. Accuracy and Macro-F1 are used as the primary quantitative indicators, while confusion matrices, per-class recall, and t-SNE visualizations are used to analyze class-specific recognition behavior and feature separability. The t-SNE plots are used only as qualitative visualization tools and are not treated as independent classification metrics.

5.1. Overall performance and t-SNE visualization

Fig. 10 and Fig. 11 present the overall diagnostic results of the proposed framework on the T1 and T2 datasets, respectively. Each figure includes the training process, Cyclical Learning Rate (CLR) schedule, confusion matrix, and t-SNE visualization of the learned features. In the revised

figures, the t-SNE subfigures are labeled with “t-SNE dimension 1” and “t-SNE dimension 2”, and the class legends are retained to clarify the correspondence between feature clusters and fault categories.

For the T1 dataset, the proposed model achieves an overall diagnostic accuracy of 99.7 % under the limited-sample setting. In this experiment, only 100 clean samples per class are used for training, and Additive White Gaussian Noise (AWGN) augmentation is applied only to the training set. The clean test set remains unchanged, which allows the evaluation to reflect the model’s ability to identify weak bearing faults rather than its ability to memorize augmented samples. As shown in Fig. 10(c), most samples are concentrated on the main diagonal of the confusion matrix, indicating that the seven bearing states are correctly distinguished. In particular, the recall of the challenging 150 μm defect categories remains above 99.2 %, showing that the proposed framework can identify early weak defects whose impact components are easily masked by background vibration.

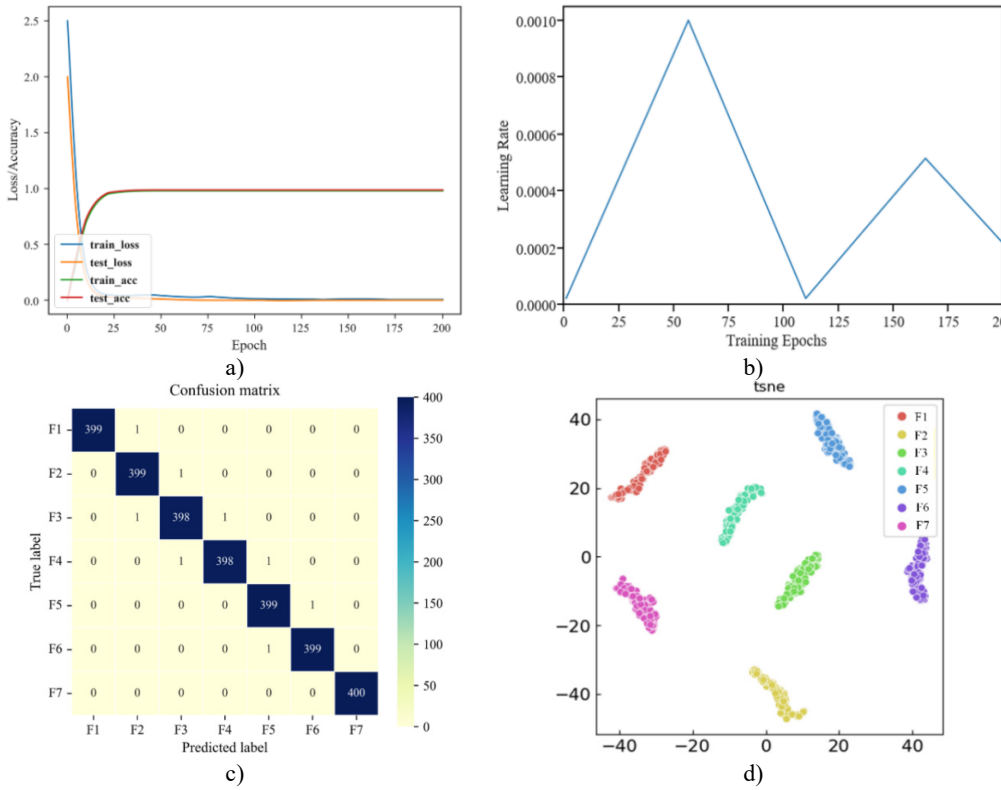


Fig. 10. Comprehensive diagnostic results on the T1 dataset: a) training loss and accuracy curves; b) CLR schedule; c) confusion matrix; and d) t-SNE visualization with labeled axes and class legend

The high performance on T1 can be explained by the complementary roles of the three core modules. DDSC-T extracts multi-scale local temporal features and adapts the branch contribution according to the input response; HRC-LSTM further models the sequential relationship among repetitive impacts; and HAA suppresses redundant channel and temporal responses while enhancing fault-sensitive features. As a result, the model can preserve weak local impulses and maintain discrimination among similar defect sizes.

For the T2 dataset, the proposed framework achieves an overall diagnostic accuracy of 98.3 %. Compared with T1, T2 represents a different fault mechanism, because the gear-root cracks mainly induce weak stiffness-related vibration variations rather than direct bearing-impact responses. Nevertheless, the model maintains stable recognition performance across normal, single-tooth

crack, and double-tooth crack conditions. As shown in Fig. 11(c), the misclassified samples are limited and mainly occur between crack categories with relatively close damage characteristics. This result indicates that the proposed model is not restricted to bearing-defect recognition, but can also extract discriminative features from gear-crack vibration responses.

The t-SNE visualizations in Fig. 10(d) and Fig. 11(d) provide an intuitive view of the feature distributions learned by the proposed framework. On T1, samples from the same category form compact clusters, and different defect types and severities are clearly separated in the two-dimensional embedded space. On T2, the five health states also show distinguishable cluster structures, although the distance between some crack-severity categories is smaller than that between the normal and fault states. This observation is consistent with the physical fact that different crack severities may produce similar weak stiffness variations, making T2 a more subtle classification task.

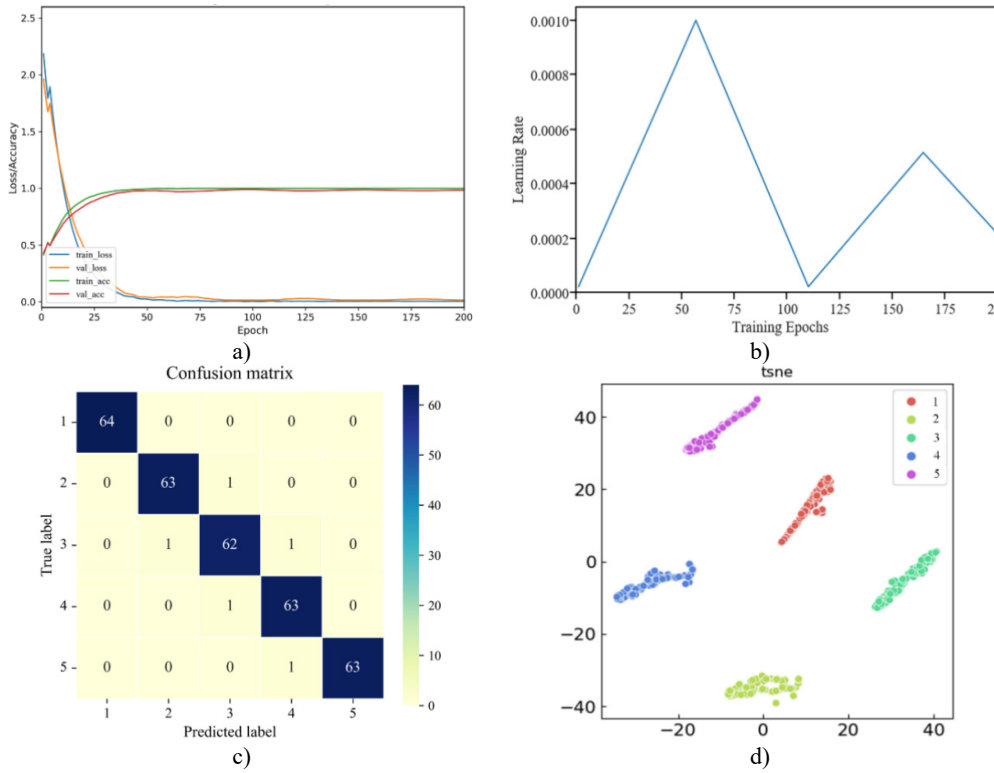


Fig. 11. Comprehensive diagnostic results on the T2 dataset: a) training loss and accuracy curves; b) CLR schedule; c) confusion matrix; and d) t-SNE visualization with labeled axes and class legend

It should be noted that t-SNE is sensitive to visualization parameters and only reflects the local neighborhood structure of high-dimensional features after nonlinear projection. Therefore, the t-SNE results are used as supplementary evidence for feature separability, while the final diagnostic conclusion is based on quantitative metrics such as accuracy, Macro-F1, per-class recall, and confusion-matrix results.

Overall, the results in Fig. 10 and Fig. 11 demonstrate that the proposed DDSC-T-HRC-LSTM-HAA framework achieves high diagnostic accuracy on two rotating-machinery datasets with different mechanical fault mechanisms. The compact intra-class distributions and separated inter-class clusters further indicate that the learned representations are discriminative. However, the remaining misclassifications in weak-defect and close-severity categories also suggest that failure-case analysis under stronger domain shifts and lower Signal-to-Noise Ratio (SNR)

conditions is necessary, which is further discussed in the following sections.

5.2. Ablation study

To evaluate the individual contribution and synergistic effect of DDSC-T, HRC-LSTM, and HAA, eight ablation variants were constructed on the T1 dataset. The same data partition, preprocessing strategy, training protocol, and evaluation metrics were used for all variants. The complete model contains all three core modules, whereas the remaining variants remove one or more modules to examine their independent and combined effects. The results are summarized in Table 9.

Table 9. Ablation results of the proposed framework on the T1 dataset

No.	DDSC-T	HRC-LSTM	HAA	Accuracy (%) (Mean ± SD)	Full model improvement (p.p.)	Statistical significance (p-value)
1	–	–	–	59.4±2.15	+40.3	($p < 0.001$)
2	+	–	–	82.0±1.68	+17.7	($p < 0.001$)
3	–	+	–	71.7±1.82	+28.0	($p < 0.001$)
4	–	–	+	64.4±1.95	+35.3	($p < 0.001$)
5	–	+	+	89.4±1.15	+10.3	($p < 0.01$)
6	+	–	–	91.0±1.02	+8.7	($p < 0.01$)
7	+	+	–	82.0±1.55	+17.7	($p < 0.001$)
8	+	+	–	99.7±0.21	–	Full model (Reference)

The baseline variant without DDSC-T, HRC-LSTM, or HAA achieves only 59.4 % accuracy, indicating that simple feature extraction is insufficient for recognizing weak bearing defects under limited training samples and noise interference. When DDSC-T is used alone, the accuracy increases to 82.0 %, demonstrating that adaptive multi-scale temporal convolution is effective for extracting local impact and resonance-related features. The HRC-LSTM-only variant achieves 71.7 %, suggesting that recurrent temporal modeling is useful but insufficient when the input representation lacks adaptive local feature extraction. The HAA-only variant obtains 64.4 %, which indicates that attention-based recalibration cannot fully compensate for the absence of discriminative front-end feature extraction.

The complete model achieves 99.7 % accuracy, outperforming the HRC-LSTM-only and HAA-only variants by 28.0 and 35.3 percentage points, respectively. This directly shows that the integrated DDSC-T-HRC-LSTM-HAA structure provides a substantial improvement over using HRC-LSTM or HAA alone. Compared with the two-module variants, the complete model also improves the accuracy by 10.3 percentage points over HRC-LSTM+HAA, 8.7 percentage points over DDSC-T+HAA, and 17.7 percentage points over DDSC-T+HRC-LSTM. These results confirm that the performance gain is not produced by a single module, but by the complementary interaction among adaptive convolution, hierarchical temporal modeling, and attention-based feature recalibration.

From a signal-processing perspective, DDSC-T first provides adaptive multi-scale filtering for local transient impulses and resonance-related responses. This is important because the frequency-band distribution of bearing fault vibration may change with defect severity, load variation, and background interference. HRC-LSTM then models the temporal dependence among repeated impacts and modulation components, allowing the network to distinguish random isolated disturbances from fault-related periodic responses. HAA further suppresses redundant channel and temporal responses, thereby enhancing the feature regions that are more closely related to fault-sensitive energy and impact timing.

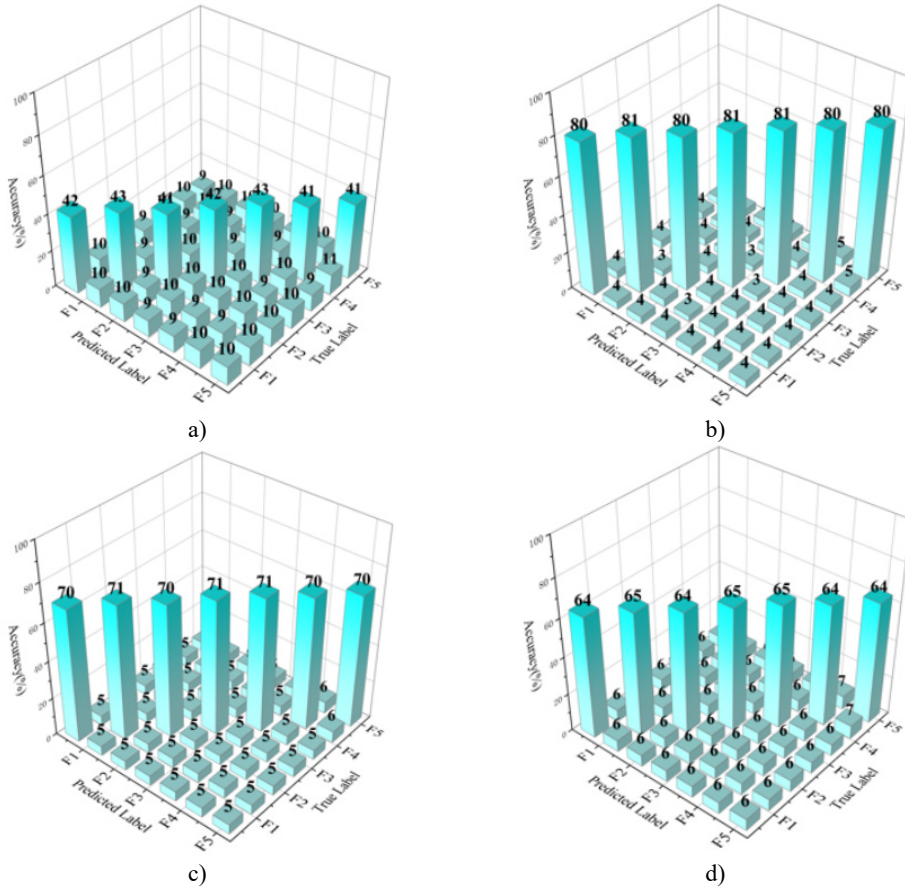
From a feature-learning perspective, the three modules play different but connected roles. DDSC-T improves the quality of the front-end representation, HRC-LSTM transforms local features into a higher-level sequential representation, and HAA adaptively refines the learned feature map before classification. If DDSC-T is removed, the recurrent and attention modules

receive less discriminative local features. If HRC-LSTM is removed, the model loses the ability to organize local impacts into sequential fault patterns. If HAA is removed, fault-sensitive representations are more easily affected by redundant channels and noisy time positions. Therefore, the complete framework forms a progressive feature-learning chain rather than a simple stacking of independent components.

The reduced standard deviation of the complete model also indicates better training stability. Its SD is 0.21 percentage points, which is lower than those of all ablated variants. This suggests that the integrated architecture not only improves average accuracy but also reduces sensitivity to random initialization and mini-batch ordering. The statistical comparison further indicates that the full model performs significantly better than each ablated variant under the repeated-run protocol.

Fig. 12 further illustrates the confusion matrices of different ablation variants. In the incomplete variants, misclassifications mainly occur between fault categories with close defect sizes or similar vibration responses, especially for weak 150 μm defects. After DDSC-T, HRC-LSTM, and HAA are combined, the confusion matrix becomes highly concentrated along the main diagonal, indicating that the proposed model improves both inter-class separability and weak-fault recognition. This observation is consistent with the quantitative results in Table 9.

Overall, the ablation study verifies that DDSC-T, HRC-LSTM, and HAA are all necessary for the final diagnostic performance. DDSC-T contributes adaptive local feature extraction, HRC-LSTM contributes hierarchical temporal dependency modeling, and HAA contributes channel-temporal feature enhancement. Their combination yields a clear synergistic gain, which explains why the proposed framework achieves stronger performance than the single-module and two-module variants.



a) b)

c) d)

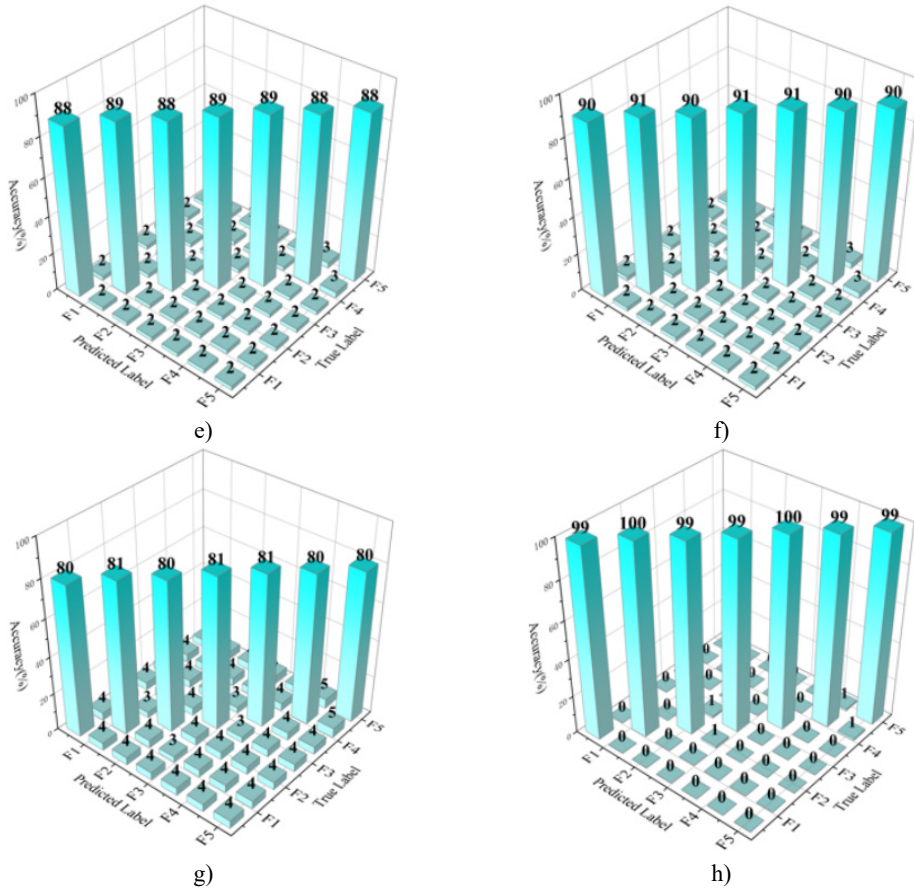


Fig. 12. Confusion matrices of different ablation variants on the T1 dataset: a) baseline, b) DDSC-T only, c) HRC-LSTM only, d) HAA only, e) HRC-LSTM + HAA, f) DDSC-T + HAA, g) DDSC-T + HRC-LSTM, and h) proposed framework

5.3. Cross-domain transfer

To further evaluate the generalization ability of the proposed DDSC-T-HRC-LSTM-HAA framework under operating-condition variation, a direct cross-condition transfer experiment was conducted on the T1 dataset. Unlike the within-condition experiment in Section 5.1, the source and target samples in this experiment were collected under different speed-load conditions. Therefore, this experiment provides a stricter evaluation of whether the learned fault representation remains discriminative when the vibration distribution changes due to rotational-speed and load variation.

Following the protocol described in Section 4.4, all models were trained only on the source condition of 100 Hz and 0 N. the trained models were then directly tested on three unseen target conditions without fine-tuning: 300 Hz and 0 N, 100 Hz and 1400 N, and 300 Hz and 1400 N. These three target conditions correspond to speed shift, load shift, and combined speed-load shift, respectively. No target-domain samples were used for model training, hyperparameter selection, or standardization-parameter estimation.

To maintain a concise but representative comparison, four models were evaluated: WDCNN, CNN-GRU, ResNet-CBAM, and the proposed DDSC-T-HRC-LSTM-HAA framework. WDCNN represents a classical convolution-based diagnostic model, CNN-GRU represents a convolutional-recurrent structure, and ResNet-CBAM represents an attention-enhanced residual convolutional

model. the same source samples, target samples, preprocessing procedure, and random seeds were used for all models. The cross-condition transfer results are summarized in Table 10.

Table 10. Cross-condition transfer results on the T1 dataset

Model	Target domain TS (300 Hz, 0 N) accuracy / macro-F1 Score (%)	Target domain TL (100 Hz, 1400 N) accuracy / macro-F1 Score (%)	Target domain TSL (300 Hz, 1400 N) accuracy / macro-F1 Score (%)	Condition-averaged accuracy / macro-F1 Score (%)
WDCNN	65.3±2.8 / 63.8±3.1	73.5±2.4 / 71.9±2.5	52.4±3.5 / 50.3±3.8	63.7±2.9 / 62.0±3.1
CNN-GRU	78.4±2.1 / 77.5±2.3	84.6±1.9 / 83.1±2.2	58.7±2.9 / 56.4±3.2	73.9±2.3 / 72.3±2.6
ResNet-CBAM	82.5±1.8 / 81.2±2.1	88.1±1.5 / 87.2±1.8	62.3±2.6 / 60.1±2.8	77.6±2.0 / 76.2±2.2
Proposed	96.2±0.6 / 95.8±0.8	97.8±0.4 / 97.4±0.6	91.5±1.4 / 90.8±1.7	95.2±0.8 / 94.7±1.0

As shown in Table 10, all models experience performance degradation when transferred from the source condition to unseen speed-load conditions, but the degradation differs substantially among architectures. Under the speed-shift condition of 300 Hz and 0 N, the proposed model achieves 96.2 % accuracy and 95.8 % Macro-F1, outperforming WDCNN, CNN-GRU, and ResNet-CBAM by 30.9, 17.8, and 13.7 percentage points in accuracy, respectively. This indicates that the proposed dynamic temporal representation can better accommodate changes in fault-impact intervals caused by rotational-speed variation.

Under the load-shift condition of 100 Hz and 1400 N, all models perform better than under the speed-only shift, suggesting that load variation alone is less destructive to temporal alignment than speed variation in this dataset. Nevertheless, the proposed model still achieves the best performance, with 97.8 % accuracy and 97.4 % Macro-F1. Compared with ResNet-CBAM, the strongest baseline in this condition, the proposed model improves accuracy and Macro-F1 by 9.7 and 10.2 percentage points, respectively.

The combined speed-load shift of 300 Hz and 1400 N is the most challenging target condition. in this case, the accuracies of WDCNN, CNN-GRU, and ResNet-CBAM decrease to 52.4 %, 58.7 %, and 62.3 %, respectively, showing that fixed or weakly adaptive feature extractors are sensitive to simultaneous changes in temporal structure and vibration-energy distribution. in contrast, the proposed model maintains 91.5 % accuracy and 90.8 % Macro-F1. Compared with ResNet-CBAM, the proposed model improves accuracy and Macro-F1 by 29.2 and 30.7 percentage points, respectively.

The condition-averaged results further demonstrate the cross-domain advantage of the proposed framework. the proposed model obtains 95.2 % averaged accuracy and 94.7 % averaged Macro-F1, exceeding WDCNN by 31.5 and 32.7 percentage points, CNN-GRU by 21.3 and 22.4 percentage points, and ResNet-CBAM by 17.6 and 18.5 percentage points, respectively. in addition, the standard deviations of the proposed model are consistently smaller than those of the baseline models in most target conditions, indicating more stable training and transfer behavior across independent runs.

From a signal-processing perspective, the superior cross-condition performance can be attributed to the complementary roles of DDSC-T, HRC-LSTM, and HAA. DDSC-T provides dynamically weighted multi-scale temporal convolution, which helps capture fault responses when the impulse interval changes with speed. HRC-LSTM organizes the local responses into sequential fault patterns, reducing the influence of isolated random fluctuations. HAA further recalibrates channel and temporal features, suppressing target-condition responses that are less relevant to fault discrimination. Therefore, the proposed model retains a more stable representation under operating-condition shifts than the baseline models.

Fig. 13 illustrates the cross-condition performance comparison of the four models. the results are reported separately for speed shift, load shift, and combined speed-load shift, making it possible to distinguish the dominant source of performance degradation. the combined speed-load condition produces the largest decline for all models, which indicates that practical cross-domain diagnosis remains challenging when both the temporal fault pattern and vibration-energy

distribution change simultaneously.

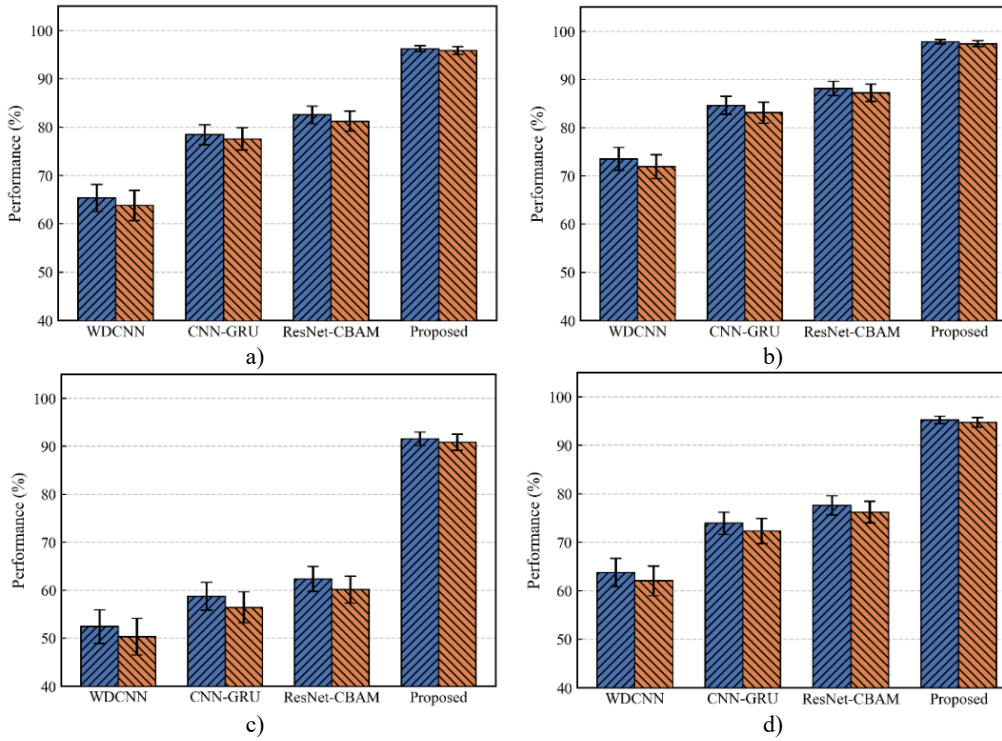


Fig. 13. Cross-condition diagnostic performance comparison on the T1 dataset under unseen speed-load conditions: a) 300 Hz and 0 N (Speed shift); b) 100 Hz and 1400 N (Load shift); c) 300 Hz and 1400 N (Combined Shift); and d) condition-averaged performance

Although the proposed framework achieves the best transfer performance, the result under the combined speed-load shift also reveals a limitation of the current direct-transfer setting. Since no target-domain data are used for model adaptation, the learned representation may still degrade when the target working condition deviates substantially from the source domain. in practical industrial applications, if the equipment operates over a wider speed–load range, a small amount of target-condition calibration data or an explicit domain-adaptation strategy may further improve reliability.

Overall, the cross-domain transfer experiment provides stronger evidence of generalization than a random within-condition split. the source–target separation ensures that the model is evaluated on physically different operating conditions, and the comparison with WDCNN, CNN-GRU, and ResNet-CBAM confirms that the proposed dynamic temporal convolution, residual recurrent modeling, and hierarchical attention mechanism provide a more robust feature representation under operating-condition shifts.

5.4. Edge deployment

To evaluate the practical deployment potential of the proposed framework, the computational overhead of the compared models was measured on a unified NVIDIA Jetson Xavier NX platform. the evaluation considered the number of trainable parameters, computational complexity, peak memory usage, and single-segment inference latency. All models received the same input format and were evaluated under the same embedded-device conditions. The results are listed in Table 11.

The proposed model contains only 0.31 M trainable parameters and requires 0.48 GFLOPs per

input segment. Compared with WDCNN, the parameter count and computational complexity are reduced by approximately 44.6 % and 41.5 %, respectively. Compared with ResNet-CBAM, the reductions reach approximately 63.5 % and 66.9 %, respectively. Compared with 1D-ViT, the proposed model reduces the parameter count by approximately 74.2 % and the computational complexity by approximately 77.1 %. These results demonstrate that the combination of depth-wise separable convolution, residual recurrent modeling, and lightweight attention provides a favorable balance between representation capability and computational cost.

Table 11. Computational overhead and edge-inference performance on the NVIDIA Jetson Xavier NX

Model	Model parameters (M)	Computational complexity (GFLOPs)	Peak memory footprint (MB)	Inference latency (ms/segment)	Theoretical throughput (segments/s)
CGRU	0.02	0.05	650	3.2	312.5
WDCNN	0.56	0.82	1,250	6.5	153.8
ResNet-CBAM	0.85	1.45	1,850	11.4	87.7
1D-ViT	1.20	2.10	2,600	18.6	53.8
Proposed	0.31	0.48	1,050	9.8	102.0

The proposed model requires 1050 MB of peak memory and achieves a single-segment inference latency of 9.8 ms on the Jetson Xavier NX. Its corresponding theoretical throughput, calculated as (1000/9.8), is approximately 102.0 segments/s. Therefore, the model can process more than 100 fixed-length segments per second under the reported measurement condition. This result indicates that the proposed framework has the potential to satisfy real-time segment-level diagnosis requirements in embedded monitoring systems. the actual system-level monitoring rate may also depend on sensor acquisition, data buffering, preprocessing, and communication overhead, which are not included in the network-only latency reported here.

CGRU has the lowest computational cost and latency among the compared models, but its small model capacity should be considered together with its diagnostic performance. WDCNN provides lower latency than the proposed model but requires more parameters and higher computational complexity. ResNet-CBAM and 1D-ViT require substantially more memory and computation, and their latencies increase to 11.4 and 18.6 ms, respectively. in particular, the latency of 1D-ViT is approximately 1.90 times that of the proposed model, which limits its suitability for resource-constrained edge devices.

The reported comparison provides a practical runtime comparison with representative lightweight and hybrid diagnostic architectures, including CGRU, WDCNN, ResNet-CBAM, and 1D-ViT. However, the original deployment experiment did not include a calibrated power meter or an energy-monitoring module. Accordingly, energy consumption per inference is not reported or estimated in Table 11. FLOPs, memory usage, and latency should not be interpreted as direct measurements of electrical energy, because actual power consumption also depends on the device power mode, clock frequency, thermal condition, utilization, and measurement interval. Direct power and energy-per-inference measurements using a calibrated external power monitor will be included in future deployment experiments.

Overall, the proposed framework does not achieve the minimum latency among all models, but it provides a better compromise between diagnostic representation capability, computational complexity, memory usage, and real-time inference performance. This trade-off is important for early weak-fault diagnosis, where an excessively small model may reduce feature-extraction capability, whereas a high-capacity Transformer or attention-heavy model may exceed the resource limits of embedded platforms.

5.5. Extreme-noise robustness

To evaluate the robustness boundary of the proposed DDSC-T-HRC-LSTM-HAA framework under severe noise interference, Additive White Gaussian Noise (AWGN) was injected into the

test signals at different Signal-to-Noise Ratio (SNR) levels. the SNR was gradually decreased from 20 dB to –10 dB to simulate increasingly harsh electromagnetic and mechanical background interference. in this experiment, the trained models were directly tested on the noise-contaminated test sets without retraining or noise-level-specific fine-tuning. Therefore, the degradation curves reflect the inherent noise tolerance of each diagnostic model.

Fig. 14 shows the diagnostic accuracy degradation of different models under varying SNR levels. When the SNR is relatively high, the added noise only weakly disturbs the fault-related impact and modulation components. Under this condition, the proposed model maintains stable recognition performance. in particular, when SNR ≥ 5 dB, the diagnostic accuracy remains above 98 %, indicating that the proposed framework can effectively suppress moderate background noise while preserving fault-sensitive vibration features.

As the SNR decreases, all models show different degrees of performance degradation. This trend is physically reasonable because strong broadband noise can obscure weak transient impulses, reduce the contrast between adjacent defect severities, and distort the temporal regularity of repeated fault responses. the degradation becomes more evident at 0 dB, where the noise energy is approximately equal to the signal energy. Under this extremely challenging condition, the proposed framework still achieves an accuracy of 61.08 %, whereas the baseline models approach a near-failure state. This result indicates that the proposed model has a stronger ability to extract residual discriminative features from heavily contaminated vibration signals.

The robustness of the proposed model can be attributed to the cooperative effect of its three core modules. DDSC-T provides adaptive multi-scale temporal filtering, which helps retain local fault-related impact components under noise interference. HRC-LSTM models the temporal dependence among repeated responses, making the model less sensitive to isolated random fluctuations. HAA further recalibrates channel and temporal features, suppressing redundant responses dominated by broadband noise and strengthening salient fault-sensitive regions. Therefore, the proposed model does not rely solely on high-amplitude local impulses, but combines local temporal patterns, sequential regularity, and attention-based feature enhancement.

Nevertheless, the result at 0 dB also reveals a practical limitation. Although the proposed model performs better than the compared baselines, its accuracy decreases markedly compared with the clean or moderate-noise conditions. This indicates that when random noise energy becomes comparable to the useful fault signal, part of the weak defect information is inevitably masked. in such cases, misclassification is more likely to occur between fault categories with similar defect sizes or similar vibration-response patterns, especially for early weak defects. Therefore, the present framework improves the effective noise-tolerance boundary, but it cannot completely eliminate the influence of extremely low-SNR conditions.

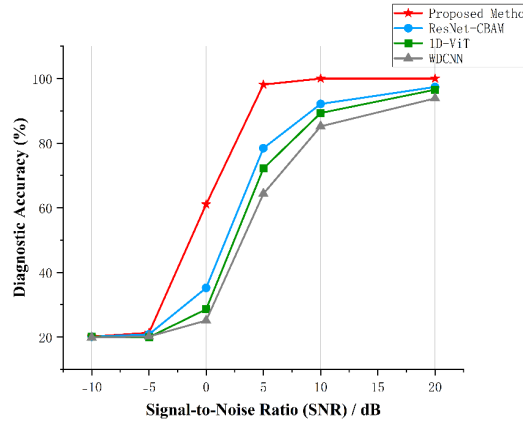


Fig. 14. Diagnostic accuracy degradation curves of different models under AWGN interference with SNR decreasing from 20 dB to –10 dB

ISSN PRINT 1392-8716, ISSN ONLINE 2538-8460

For SNR levels below 0 dB, the diagnostic task becomes even more difficult because the fault-related components are no longer dominant in the observed signal. the corresponding results in Fig. 14 should therefore be interpreted as stress-test evidence rather than as normal operating performance. in practical deployment, when the monitoring environment is expected to contain such extreme noise, additional signal-quality control, sensor shielding, denoising preprocessing, or multi-sensor fusion may be required to ensure reliable diagnosis.

Overall, the extreme-noise experiment demonstrates that the proposed DDSC-T-HRC-LSTM-HAA framework has better robustness than the compared baseline models under AWGN interference. the model maintains high accuracy under moderate noise and still preserves partial diagnostic capability at 0 dB. at the same time, the significant degradation under extremely low-SNR conditions provides a clear boundary for the practical application of the current framework.

6. Conclusions

This study proposed a lightweight DDSC-T-HRC-LSTM-HAA framework for early fault diagnosis of rotating machinery under limited-sample, noisy, and variable-operating-condition scenarios. the proposed framework integrates dynamic depth-wise separable temporal convolution, hierarchical residual-connected LSTM modeling, hierarchical adaptive attention, and an auxiliary CNN branch. DDSC-T extracts adaptive multi-scale local temporal features, HRC-LSTM models hierarchical dependencies among repetitive fault responses, and HAA recalibrates the learned representations along the channel and temporal dimensions. The auxiliary CNN branch further provides complementary local feature information through adaptive feature fusion.

The experimental results demonstrate that the proposed framework achieves 99.7 % accuracy on the T1 bearing dataset and 98.3 % accuracy on the T2 gearbox dataset. For the challenging 150 μm bearing defects in T1, the recall remains above 99.2 %. The ablation results further show that the complete model substantially outperforms the HRC-LSTM-only and HAA-only variants, with accuracy improvements of 28.0 and 35.3 percentage points, respectively. These results indicate that the final performance is derived from the complementary interaction of adaptive local feature extraction, hierarchical temporal modeling, and channel-temporal feature recalibration rather than from a single module.

The direct cross-condition experiment on the T1 dataset provides additional evidence of operating-condition generalization. When the model is trained at 100 Hz and 0 N and directly tested under unseen speed-load conditions, the proposed framework achieves condition-averaged Accuracy and Macro-F1 values of 95.2 % and 94.7 %, respectively. Under the most challenging combined condition of 300 Hz and 1400 N, it maintains 91.5 % Accuracy and 90.8 % Macro-F1, outperforming WDCNN, CNN-GRU, and ResNet-CBAM. These results suggest that the dynamic temporal representation and hierarchical recalibration mechanism can reduce, but cannot completely eliminate, the influence of speed and load shifts.

The edge-deployment evaluation further confirms the lightweight characteristics of the proposed framework. The model contains 0.31 M trainable parameters, requires 0.48 GFLOPs, uses 1050 MB of peak memory, and achieves a single-segment inference latency of 9.8 ms on the NVIDIA Jetson Xavier NX. The corresponding network-level throughput is approximately 102 fixed-length segments per second. These properties make the framework suitable for embedded condition-monitoring systems in which computational resources and response time are constrained.

The extreme-noise experiment shows that the proposed model maintains an accuracy above 98 % when the SNR is no lower than 5 dB. At 0 dB, where the noise energy is approximately equal to the signal energy, the accuracy decreases to 61.08 % but remains superior to those of the compared models. This result confirms the improved noise tolerance of the proposed framework, while also revealing that extremely low-SNR conditions remain a practical limitation because weak fault impulses can be physically masked by broadband noise.

Several limitations should be acknowledged. First, the current validation is based on two rotating-machinery datasets, and the cross-condition experiment is conducted within the T1 dataset rather than across different machines or sensing systems. Second, the direct-transfer protocol does not use target-domain adaptation, so performance may decrease when the target operating condition lies outside the source-domain range. Third, the noise analysis is based on AWGN and does not fully represent non-stationary industrial interference, structural resonance coupling, or sensor-specific noise. Finally, the reported edge-deployment measurements characterize network inference time; calibrated power consumption and energy per inference were not measured in the present study.

Future work will focus on four directions. First, more public datasets and field measurements involving different machines, loads, speeds, and sensor configurations will be introduced to further examine cross-machine generalization. Second, domain-adaptation or domain-generalization strategies will be investigated to reduce the dependence on source-domain operating conditions. Third, real industrial noise, multi-sensor fusion, and online incremental learning will be considered to improve reliability under changing environments. Fourth, calibrated power monitoring and energy-per-inference measurements will be conducted on embedded platforms to establish a more complete assessment of deployment efficiency.

In summary, the proposed DDSC-T-HRC-LSTM-HAA framework provides a compact and interpretable solution for small-sample rotating-machinery fault diagnosis. Its combination of adaptive temporal feature extraction, hierarchical sequence modeling, attention-based noise suppression, and lightweight edge inference offers practical potential for online bearing and gearbox condition monitoring, while the identified limitations define the boundary for further validation and engineering improvement.

Acknowledgements

This work was supported in part by the Key Science and Technology Research project of the Henan province (Grant No. 242102220013), the Henan Province Intelligent Manufacturing Industry College (Grant No. YYSFZD2501) and the Henan Province Key Scientific and Technological Project (Grant No. 252102220121).

Data availability

The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Author contributions

Hongchao Wang is the theoretical researcher, Caixia Fan is the writer of the paper and the program programmer in the paper.

Conflict of interest

The authors declare that they have no conflict of interest.

References

[1] Y. Lei, B. Yang, X. Jiang, F. Jia, N. Li, and A. K. Nandi, "Applications of machine learning to machine fault diagnosis: a review and roadmap," *Mechanical Systems and Signal Processing*, Vol. 138, p. 106587, Jan. 2020, https://doi.org/10.1016/j.ymssp.2019.106587

[2] D. Neupane and J. Seok, "A review on deep learning techniques for the diagnosis of novel coronavirus (COVID-19)," *Neurocomputing*, Vol. 627, p. 129588, 2025.

[3] S. Zhang, S. Zhang, B. Wang, and T. G. Habetler, "Deep learning algorithms for bearing fault diagnostics-a comprehensive review," *IEEE Access*, Vol. 8, pp. 29857–29881, Feb. 2020, https://doi.org/10.1109/access.2020.2972859

[4] Z. Zhao et al., "Deep learning algorithms for rotating machinery intelligent diagnosis: an open source benchmark study," *ISA Transactions*, Vol. 107, pp. 224–255, Aug. 2020, https://doi.org/10.1016/j.isatra.2020.08.010

[5] X. Li, W. Zhang, Q. Ding, and J.-Q. Sun, "Intelligent rotating machinery fault diagnosis based on deep learning using data augmentation," *Journal of Intelligent Manufacturing*, Vol. 31, No. 2, pp. 433–452, Nov. 2018, https://doi.org/10.1007/s10845-018-1456-1

[6] Z. Wang, "A review of deep learning-based intelligent fault diagnosis methods for rotating machinery under small-sample conditions," *Exploring Science Academic Conference Series*, Vol. 14, pp. 162–167, Jun. 2026, https://doi.org/10.70267/ic-aimees.202602162167

[7] M. Umar, M. F. Siddique, J. Kim, and J.-M. Kim, "A distribution-level statistical framework for reliable pipeline leak detection using multi-domain signal analysis," *Scientific Reports*, Vol. 16, No. 1, Jun. 2026, https://doi.org/10.1038/s41598-026-56335-6

[8] W. Zhang, G. Peng, C. Li, Y. Chen, and Z. Zhang, "A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals," *Sensors*, Vol. 17, No. 2, p. 425, Feb. 2017, https://doi.org/10.3390/s17020425

[9] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mechanical Systems and Signal Processing*, Vol. 115, pp. 213–237, Jun. 2018, https://doi.org/10.1016/j.ymssp.2018.05.050

[10] Q. Zhang, Y. Yao, Y. Huang, Y. Liu, and L. Wu, "A bearing fault diagnosis model based on a simplified wide convolutional neural network and random forrest," *Sensors*, Vol. 25, No. 3, p. 752, 2025, https://doi.org/10.3390/s25030752

[11] X. Zeng and X. Feng, "Bearing fault diagnosis with small sample sizes based on CNN-SE-LSTM transfer learning," in *2025 40th Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pp. 2617–2622, 2025, https://doi.org/10.1109/yac66630.2025.11149878

[12] J. Wang, D. Wang, S. Wang, W. Li, and K. Gryllias, "Fault diagnosis of planetary gearbox based on RTSWMFE, IS-GSE and COOT-SVM," *Journal of Vibration, Measurement and Diagnosis*, Vol. 44, No. 6, pp. 1121–1130, 2024.

[13] Z. Cao, "Application of improved CNN-LSTM model in fault diagnosis of rolling bearings," *Computer Systems and Applications*, Vol. 33, No. 5, pp. 121–129, 2024.

[14] H. Wang, H. Wang, and X. Tang, "A review of deep learning in rotating machinery fault diagnosis and its prospects for port applications," *Applied Sciences*, Vol. 15, No. 21, p. 11303, Oct. 2025, https://doi.org/10.3390/app152111303

[15] Z. Liu, Y. Wang, and K. Feng, "Lightweight residual network for rolling bearing fault diagnosis under variable working conditions," *IEEE Transactions on Instrumentation and Measurement*, Vol. 72, p. 3516812, 2023.

[16] B. Zhang, H. Li, and S. Ma, "PDSResNet-AM: a lightweight attention-enhanced residual network for rotating machinery fault diagnosis," *Measurement*, Vol. 221, p. 113543, 2023, https://doi.org/10.1016/j.measurement.2024.115498

[17] M. F. Siddique, W. Zaman, M. Khalid, B. Hamdan, and J.-M. Kim, "A multistage transfer learning framework for intelligent fault diagnosis of rotating machinery under variable operating conditions," *Scientific Reports*, Vol. 16, No. 1, Apr. 2026, https://doi.org/10.1038/s41598-026-47377-x

[18] M. H. Alsharif, A. H. Kelechi, A. Jahid, R. Kannadasan, and M. K. Singla, "A comprehensive review of attention mechanisms in deep learning for rotating machinery fault diagnosis," *IEEE Access*, Vol. 12, pp. 34520–34540, 2024.

[19] Y. Ding, M. Jia, Q. Miao, and Y. Cao, "A multi-modal domain adaptation method for bearing fault diagnosis under variable working conditions," *Reliability Engineering and System Safety*, Vol. 241, p. 109672, 2024.

[20] Y. Xu, Z. Li, S. Wang, and W. Du, "A two-stage attention-based multi-scale CNN with prediction interval estimation for bearing fault diagnosis," *Knowledge-based Systems*, Vol. 278, p. 110845, 2023.

[21] H. Wang, J. Xu, R. Yan, and R. X. Gao, "Multi-branch temporal attention network for intelligent bearing fault diagnosis under variable speed conditions," *IEEE Transactions on Industrial Informatics*, Vol. 20, No. 3, pp. 4567–4576, 2024.

[22] Z. Chen, J. Wu, C. Deng, and X. Wang, "Frequency-domain self-attention with residual connections for rolling bearing fault diagnosis," *Measurement Science and Technology*, Vol. 35, No. 5, p. 056107, 2024, https://doi.org/10.1109/tim.2024.3436093

[23] X. Ma, S. Xu, and P. Shang, "Fault diagnosis of gearbox under open set and cross working condition based on transfer learning," (in Chinese), *Journal of Beijing University of Aeronautics and Astronautics*, Vol. 50, No. 5, pp. 1753–1760, 2024, https://doi.org/10.13700/j.bh.1001-5965.2022.0719

[24] Z. Wang, J. Zhou, and Y. Shao, "Parallel convolution module-based intelligent fault diagnosis network for rotating machinery under time-varying speeds," *Mechanical Systems and Signal Processing*, Vol. 200, No. 24, p. 110579, 2023, https://doi.org/10.1109/jsen.2023.3326112

[25] J. Yao, H. Jing, and C. Zhao, "A multimode coupled input neural network for rotating machinery fault diagnosis in noisy environment," *Control and Decision*, Vol. 38, No. 7, pp. 1687–1694, 2023.

[26] Z. Ma and Y. Zhang, "A study on rolling bearing fault diagnosis using RIME-VMD," *Scientific Reports*, Vol. 15, No. 1, p. 89161, Feb. 2025, https://doi.org/10.1038/s41598-025-89161-3

[27] A. Taibi et al., "Diagnosis of misalignment faults using the DTCWT-RCMFDE and LSSVM algorithms," *Scientific Reports*, Vol. 15, No. 1, p. 32128, Sep. 2025, https://doi.org/10.1038/s41598-025-12407-7

[28] G. San Martin, E. López Droguett, V. Meruane, and M. Das Chagas Moura, "Deep variational auto-encoders: a promising tool for dimensionality reduction and ball bearing elements fault diagnosis," *Structural Health Monitoring*, Vol. 18, No. 4, pp. 1092–1128, Jul. 2018, https://doi.org/10.1177/1475921718788299

[29] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, 2018, https://doi.org/10.1109/cvpr.2018.00745

[30] A. Khan et al., "Calibrated deep-learning risk indexing and latent behavioural profiling for occupational mental-health risk assessment," *Bioengineering*, Vol. 13, No. 6, p. 626, May 2026, https://doi.org/10.3390/bioengineering13060626

[31] S. P. Parida, S. Sahoo, and P. C. Jena, "Prediction of multiple transverse cracks in a composite beam using hybrid RNN-mPSO technique," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, Vol. 238, No. 16, pp. 7977–7986, Mar. 2024, https://doi.org/10.1177/09544062241239415

[32] B. B. Rout, S. P. Parida, and R. R. Dash, "Dynamic analysis of multi-cracked hollow rectangular beams," in *Lecture Notes in Mechanical Engineering*, Singapore: Springer Nature Singapore, 2024, pp. 369–380, https://doi.org/10.1007/978-981-97-1080-5_30

[33] S. P. Parida, S. Sahoo, and P. C. Jena, "A hybrid RNN-fuzzy-PSO model for forecasting multiple transverse cracks in laminated composite beam-like structures," (in Korean), *Smart Structures and Systems*, Vol. 36, No. 2, pp. 123–137, 2025, https://doi.org/10.12989/sss.2025.36.2.123

[34] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, Vol. 9, No. 8, pp. 1735–1780, Nov. 1997, https://doi.org/10.1162/neco.1997.9.8.1735



Caixia Fan received Master's degree in Lanzhou University of Technology, Lanzhou, China, in 2007. Associate Professor. Now she works at Department of Engineering, Huanghe Science and Technology College, Zhengzhou, China. Her current research interests include digital manufacturing of complex parts, fault diagnosis and reliability, dynamic testing and intelligent instruments.



Hongchao Wang received Ph.D. degree in Shanghai Jiaotong University, Shanghai, China, in 2015. Now he works at Zhengzhou university of light industry. His current research interests include signal processing and rotating machinery fault diagnosis.