

Bearing Fault Diagnosis Based on Time Series Isolation Slicing and Multi-domain Feature Fusion

Wuchu Tang, Guofu Wu*, Jiawei Yang and Wenxin Dong

School of Mechanical Engineering, Dalian Jiaotong University, Dalian 116000, China

*Corresponding author: w52688610@163.com

Received 15 July, 2026; Revised 01 September 2026; Accepted 16 October 2026; Published online 16 October 2026

Abstract: To address the potential data leakage associated with overlapping slicing and the limited generalization capability of rolling bearing fault diagnosis models under strong noise, small-sample, and class-imbalanced conditions, a parallel CNN – BiLSTM – CBAM method based on time-isolated slicing and multi-domain feature fusion is proposed. A time-isolated data construction strategy is adopted, in which continuous vibration records are first partitioned into independent temporal segments and then sliced separately, thereby reducing the risk of cross-subset information leakage caused by overlapping windows. In addition, -2 dB noise and class-imbalance processing are introduced to simulate complex operating conditions. A dual-branch EWT – CWT mechanism is constructed to collaboratively extract multi-domain time – frequency features, and a gated adaptive fusion module is designed to dynamically estimate channel weights, thereby enhancing fault-sensitive features while suppressing noise interference. A parallel network is employed to capture local impulsive characteristics and bidirectional temporal dependencies, while CBAM further refines critical fault information. The parameter sizes and training times of different models are also preliminarily compared. The proposed method is evaluated on a self-developed bearing test rig under multiple coupled speed – load conditions and on the public CWRU dataset. Experimental results indicate that overlap-first sampling followed by random partitioning may lead to overestimated test accuracy, whereas time-isolated partitioning provides a more reliable indication of model performance on temporally independent test data.

Keywords: Rolling bearing fault diagnosis, Imbalanced class distribution, Time-series isolated slicing, Gated adaptive fusion.

1. Introduction

As critical motion-transmitting and load-bearing units within rotating machinery systems, rolling element bearings play a vital role in minimizing frictional loss and guaranteeing stable equipment operation [1][2]. Consequently, they see extensive applications across diverse industrial sectors,

including wind turbines, aeronautics, and rail transport, where their operational integrity dictates overall system safety [3][4]. Industrial statistics indicate that 45%–55% of rotating equipment breakdowns stem from bearing damage, which often causes unscheduled downtime or severe safety accidents[5]. Under harsh working environments characterized by heavy loads, elevated rotational speeds, and corrosive

media, bearings tend to suffer from structural deterioration such as fatigue spalling, surface wear, and cage breakdown. Thus, creating highly precise and dependable fault diagnosis techniques is indispensable for realizing effective preventive maintenance and operational security [6][7]. Nevertheless, raw bearing vibration signals usually suffer from heavy background noise, making single-perspective analysis inadequate for robust fault identification [8]. Consequently, holistic multi-perspective strategies are necessary to extract informative fault signatures while preserving shock-induced impulse characteristics to ensure high diagnostic precision [9].

Over the past few decades, bearing diagnostic paradigms have progressed from classical signal processing to data-driven deep representation learning [10]. Initial research primarily leveraged time-domain statistical metrics, frequency-domain spectrums, and standard time-frequency analysis to capture health signatures [11]. To tackle non-stationary behavior and weak fault responses obscured by heavy noise, adaptive modal decomposition algorithms—such as empirical mode decomposition (EMD), empirical wavelet transform (EWT), and variational mode decomposition (VMD)—were extensively deployed to isolate local frequency bands and highlight transient shock patterns [12]–[14]. Meanwhile, continuous wavelet transform (CWT) and short-time Fourier transform (STFT) were adopted to model transient energy distribution and frequency modulation dynamics [15][16]. Despite their efficacy, these conventional techniques heavily depend on manual preset parameters and frequently fail to simultaneously capture fine-grained local details alongside global time-frequency evolution. This limitation has driven the integration of interpretable mathematical transformations with the

automated feature representation capabilities of deep neural networks.

Consequently, deep learning-driven intelligent representation methods have emerged as a dominant research area [17]. Convolutional neural networks (CNN) are widely utilized to mine localized structural features from raw vibration signals or 2D time-frequency maps. Hybrid architectures like CNN-LSTM and CNN-BiLSTM, along with bidirectional long short-term memory networks (BiLSTM), further improve the capability to model complex temporal dynamics and non-stationary features [18]–[20]. Furthermore, to counteract feature attenuation under intense noise, the convolutional block attention module (CBAM) has been incorporated to emphasize discriminative representations across spatial and channel dimensions [21]–[23]. Although these methodologies yield impressive accuracy in state classification, their robustness and generalization under low signal-to-noise ratios (SNR) and scarce training data require further improvement.

While high classification accuracies are frequently claimed in the literature, the true integrity of model validation relies on whether the sample slicing and dataset partitioning protocols are strictly executed [24]. To overcome sample scarcity, overlapping sliding windows are widely employed for data augmentation by altering the starting index along continuous time series [25][26]. However, in studies utilizing standard benchmarks like the Case Western Reserve University (CWRU) dataset, many works specify window size, overlap ratio, and split proportions without disclosing the exact sequence of dataset creation [27]–[29]. Specifically, it often remains ambiguous whether continuous signals are partitioned into isolated temporal blocks prior to slicing, or if overlapping samples are created first and then randomly shuffled into train/test splits. Multiple studies explicitly execute

"overlapping slicing prior to random division"[30]–[32]. While this strategy rapidly inflates sample quantity, consecutive windows share substantial raw temporal sequences. Shuffling such closely correlated samples across different subsets induces implicit data leakage between training and testing sets, yielding artificially inflated test accuracy that fails to reflect true generalization performance on unseen real-world data [33]. Consequently, recent studies emphasize that continuous vibration records must be partitioned into strictly non-overlapping temporal segments before sample slicing, thereby mitigating leakage risks [34]. Moreover, relying solely on laboratory benchmark test rigs overlooks complex industrial environments where coupled speed-load variations, complex acoustic transmission paths, and severe noise prevail. In practical applications, early-stage fault instances, multi-condition operating data, and minority-class fault samples remain extremely scarce [35]. Therefore, enhancing diagnostic stability, anti-noise capabilities, and generalizability under severe constraints—such as class imbalance, high noise levels, limited samples, and dynamic operating states—remains a major technical challenge.

To address these key challenges, this paper presents a comprehensive study covering data creation, feature learning, architecture design, and experimental validation. During data construction, a rigorous time-series isolated slicing scheme is implemented: raw continuous signals are first split into strictly non-overlapping training, validation, and testing segments prior to slicing, thereby avoiding cross-subset sample overlap and reducing the associated risk of data leakage. Concurrently, a baseline scheme employing "overlapping slicing before random shuffling" is implemented to quantitatively assess how sample division sequence influences evaluated accuracy.

In the feature learning phase, additive white Gaussian noise (SNR = -2 dB) and random sample deletion are intentionally imposed to emulate harsh industrial conditions with class imbalance. Building upon this, an EWT-CWT multi-domain feature fusion mechanism powered by a gated adaptive module is established. By leveraging a lightweight bottleneck network, the framework dynamically adjusts channel weights to amplify salient fault features and suppress background noise without expanding feature dimensions, effectively capturing localized spectral details, global time-frequency maps, and transient energy shifts simultaneously. In terms of model architecture, a dual-branch parallel CNN-BiLSTM network is developed to extract localized shock patterns and bidirectional sequence dependencies simultaneously, supplemented by CBAM to refine crucial fault signatures. Finally, comprehensive validation is executed on empirical data collected from a custom bearing platform under coupled variable speeds and loads, alongside generalization benchmark testing on the CWRU dataset. The main contributions of this study are summarized as follows:

- (1) To address the potential cross-subset information correlation introduced by overlapping sliding windows, a time-isolated data construction protocol is adopted. Before sample generation, the continuous vibration records are first partitioned into mutually independent training, validation, and test time segments, and sample slicing is then performed separately within each segment. In addition, an overlap-first random-partitioning scheme is established as a reference to quantitatively evaluate how different data construction orders affect model performance assessment.
- (2) A gated adaptive multi-domain fusion mechanism is developed to integrate the complementary representations extracted by

EWT and CWT. A lightweight bottleneck gating network dynamically estimates channel-wise fault relevance and reweights the multi-domain features, thereby enhancing fault-sensitive information and suppressing noise interference without increasing the fused feature dimensionality.

(3) A parallel CNN-BiLSTM-CBAM framework is constructed to jointly capture local impulsive characteristics and bidirectional temporal dependencies, with CBAM further refining discriminative fault representations. The complete framework is systematically evaluated through multi-seed comparisons, fusion-mechanism ablation, multi-SNR robustness analysis, and cross-dataset transfer from the self-developed platform to the CWRU dataset.

The remainder of this paper is structured as follows: Section 2 provides a detailed description of the gated adaptive EWT - CWT multi-domain feature fusion framework. Section 3 details the architectural design and principles of the parallel CNN-BiLSTM-CBAM diagnostic network; Section 4 describes the experimental platform and data construction procedure; Section 5 presents the experimental results and comprehensive analyses; and Section 6 concludes the paper.

2. Multi-Domain Feature Extraction and Fusion Method

2.1. Continuous Wavelet Transform

Vibration signatures emitted by rolling element bearings are inherently dynamic, non-stationary, and multi-resolution. When localized surface damage occurs, fault-induced impulse excitations trigger pronounced transient fluctuations across the combined time-scale space. To capture such complex time-frequency evolution behaviors, the continuous wavelet transform (CWT) is incorporated into the diagnostic framework

[36]. By performing temporal translation and scale expansion on a specified mother wavelet, CWT evaluates the transient energy distribution alongside frequency shift dynamics within the joint time-scale domain. Mathematically, for a one-dimensional vibration sequence $x(t)$, its continuous wavelet transform is expressed as:

$$CWT(s, \tau) = \int_{-\infty}^{+\infty} x(t) \frac{1}{\sqrt{s}} \psi^* \left(\frac{t - \tau}{s} \right) dt \quad (1)$$

Where s is the scale factor, which controls the dilation and compression of the wavelet; τ is the translation factor, which indicates the time position; $\psi^*(\cdot)$ denotes the complex conjugate of the mother wavelet function.

2.2. Empirical Wavelet Transform

Empirical wavelet transform (EWT) serves as a specialized signal decomposition technique underpinned by data-driven segmentation of the frequency spectrum[37]. It constructs an adaptive empirical wavelet filter bank to resolve the raw vibration signal into a series of intrinsic modal components, each characterized by distinct frequency-domain properties. The procedural execution of EWT primarily encompasses three key stages: spectral segmentation, filter bank construction, alongside signal decomposition and reconstruction.

(1) Adaptive spectral partitioning

For a real-valued discrete signal $x(k)$, its normalized Fourier spectrum is usually considered only over the positive frequency interval $[0, \pi]$. The EWT algorithm first detects the local maxima in the spectrum and then divides the frequency interval into N consecutive sub-intervals according to these local maxima. Let the n -th sub-interval be denoted as Λ_n . The interval boundary is then defined as follows:

$$\begin{aligned} \Lambda_n &= [\omega_{n-1}, \omega_n], \quad n=1, 2, \dots, N \\ \bigcup_{n=1}^N \Lambda_n &= [0, \pi] \end{aligned} \quad (2)$$

The interval boundaries satisfy: $0 = \omega_0 < \omega_1 < \dots < \omega_{N-1} < \omega_N = \pi$. To satisfy the reconstruction condition required for the subsequent construction of the filter bank, a transition band with a width of $2\tau_n$ is introduced around each boundary ω_n , where τ_n represents the transition bandwidth around the boundary, γ is the adjustment parameter controlling the transition bandwidth, and N is the number of sub-intervals after frequency-domain partitioning.

$$\gamma < \min_n \left(\frac{\omega_{n+1} - \omega_n}{\omega_{n+1} + \omega_n} \right) \quad (3)$$

Where Λ_n denotes the n -th frequency sub-interval, ω_n is the boundary frequency of the n -th sub-interval, τ_n represents the transition bandwidth around the boundary, γ is the adjustment parameter controlling the transition bandwidth, and N is the number of sub-intervals after frequency-domain partitioning.

(2) Construction of empirical wavelet filters

Following the construction principle of Meyer wavelets, EWT defines an empirical scaling function $\hat{\phi}_n(\omega)$ and an empirical wavelet function $\hat{\psi}_n(\omega)$ on each partitioned sub-interval Λ_n .

For the low-frequency band, namely when $n=1$ the empirical scaling function $\hat{\phi}_1(\omega)$ is defined as follows:

$$\hat{\phi}_1(\omega) = \begin{cases} 1, & |\omega| \leq (1-\gamma)\omega_1 \\ \cos \left[\frac{\pi}{2} \beta \left(\frac{1}{2\gamma\omega_1} (|\omega| - (1-\gamma)\omega_1) \right) \right], & (1-\gamma)\omega_1 < |\omega| \leq (1+\gamma)\omega_1 \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

For the other high-frequency bands, namely when $n>1$, the empirical wavelet function $\hat{\psi}_n(\omega)$ is defined as follows:

$$\hat{\psi}_n(\omega) = \begin{cases} 1, & (1+\gamma)\omega_{n-1} \leq |\omega| \leq (1-\gamma)\omega_n \\ \cos \left[\frac{\pi}{2} \beta \left(\frac{1}{2\gamma\omega_n} (|\omega| - (1-\gamma)\omega_n) \right) \right], & (1-\gamma)\omega_n < |\omega| \leq (1+\gamma)\omega_n \\ \sin \left[\frac{\pi}{2} \beta \left(\frac{1}{2\gamma\omega_{n-1}} (|\omega| - (1-\gamma)\omega_{n-1}) \right) \right], & (1-\gamma)\omega_{n-1} < |\omega| \leq (1+\gamma)\omega_{n-1} \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

Where $\hat{\phi}_1(\omega)$ denotes the empirical scaling function corresponding to the lowest

frequency band, $\hat{\psi}_n(\omega)$ denotes the empirical wavelet function corresponding to the n -th high-frequency sub-band, and $\beta(x)$ is a smooth transition function, which is expressed as follows:

$$\beta(x) = x^4(35 - 84x + 70x^2 - 20x^3), x \in [0, 1] \quad (6)$$

This function has smooth and monotonic properties, and is used to ensure smooth transitions between the empirical scaling function and the empirical wavelet function at adjacent frequency-band boundaries.

(3) Signal decomposition and reconstruction

By performing filtering operations in the frequency domain, the corresponding modal components can be extracted. The low-frequency approximation component $f_1(t)$ and the high-frequency detail component $f_n(t)$ of the signal are calculated as follows:

$$\begin{aligned} f_1(t) &= F^{-1} \{ \hat{x}(\omega) \hat{\phi}_1(\omega) \} \\ f_n(t) &= F^{-1} \{ \hat{x}(\omega) \hat{\psi}_n(\omega) \}, \quad n = 2, 3, \dots, N \end{aligned} \quad (7)$$

Therefore, the original signal $x(t)$ processed by EWT can be completely reconstructed from all modal components as:

$$x(t) = \sum_{n=1}^N f_n(t) \quad (8)$$

Where $\hat{x}(\omega)$ denotes the Fourier transform of the original signal; $F^{-1}\{\cdot\}$ represents the inverse Fourier transform; $f_1(t)$ is the low-frequency approximation component of the signal; $f_n(t)$ is the n -th high-frequency modal component; and N is the total number of decomposed modes.

2.3. EWT-CWT Multi-Channel Adaptive Feature Fusion

Synthesizing the complementary advantages of Continuous Wavelet Transform (CWT) and Empirical Wavelet Transform (EWT) in processing non-stationary signals, and addressing the challenge that bearing vibration features are easily submerged by strong noise under

complex variable operating conditions, this section constructs a preprocessing framework featuring dual-branch parallel feature extraction and gated adaptive fusion. This framework aims to enhance the fault discriminability and anti-noise robustness of

the fused features without increasing the feature dimensions. The specific implementation procedure is as follows (in conjunction with the flowchart shown in Fig. 1):

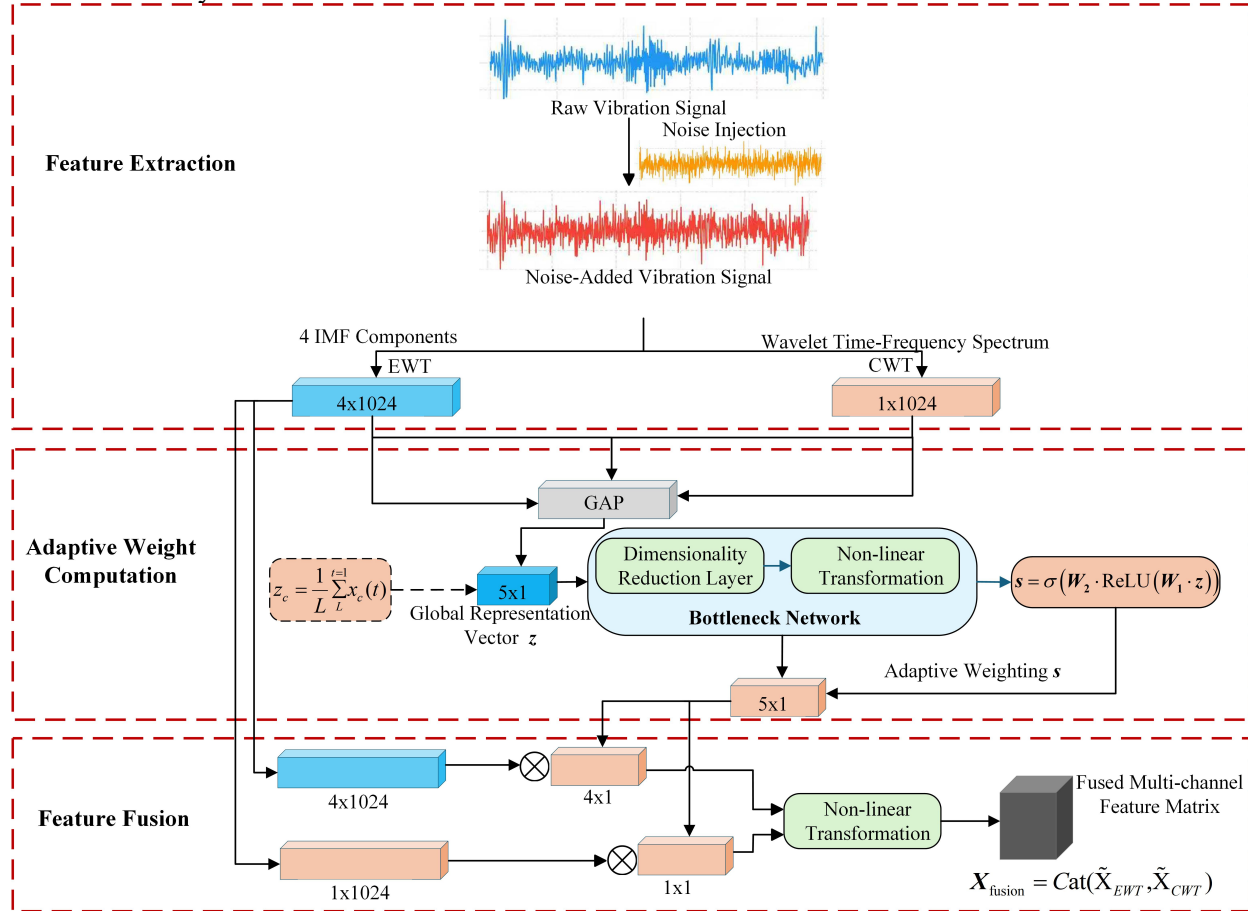


Fig. 1. Framework of EWT-CWT multi-channel adaptive dynamic weighted feature fusion

(1) Noise Contamination and Class Imbalance Synthesis

Following the time-series isolated slicing protocol, one-dimensional vibration segment samples of length $L = 1024$ undergo stochastic sample dropouts across specified target classes to synthesize a skewed class distribution. Subsequently, additive white Gaussian noise with a signal-to-noise ratio (SNR) of -2dB is superimposed onto these samples to construct a noise-contaminated, class-imbalanced dataset. This benchmark dataset serves to rigorously assess the diagnostic robustness

and noise immunity of the proposed framework under combined noise and data scarcity constraints.

(2) Dual-Branch Parallel Feature Extraction Paradigm

A dual-branch architecture combining EWT and CWT is established for multi-scale feature generation. In the EWT branch, the noise-contaminated vibration signal is adaptively decomposed into four adaptive frequency components (each with a dimension of 1024), aimed at extracting localized impulse features within different frequency bands. Concurrently, the CWT

branch processes the signal via continuous wavelet transform to generate a 2D time-frequency representation. This representation is subsequently processed by global average pooling (GAP) to obtain a 1D feature vector of 1024 dimensions. The extracted features from both branches are then forwarded to the adaptive weighting module for dynamic feature recalibration.

(3) Gated Adaptive Multi-Domain Feature Fusion

Conventional multi-channel fusion methods relying on uniform channel concatenation fail to account for varying fault sensitivities across multi-domain representations, frequently introducing redundant spectral noise under low-SNR regimes. To overcome this limitation, a gated adaptive fusion mechanism is designed to perform dynamic, weighted channel concatenation across the dual-branch outputs. Inspired by channel attention principles, this module incorporates a lightweight bottleneck network to autonomously determine the fault-relevance weight for each channel, thereby selectively magnifying informative fault signatures while suppressing irrelevant noise artifacts. The mathematical formulation proceeds as follows:

Let the EWT branch yield four IMF feature channels and the CWT branch produce one global energy envelope feature, all possessing a temporal sequence length of $L = 1024$. Initially, global average pooling (GAP) is executed along the temporal dimension for each individual feature channel. This operation condenses the temporal sequence into a scalar global representation z_c , capturing the overall fault information density of the c -th channel:

$$z_c = \frac{1}{L} \sum_{t=1}^L x_c(t) \quad (9)$$

Where, x_c is the amplitude of the c -th feature channel at the t -th time step, and z_c is the global aggregated value of the c -th channel.

The global representation vector of the 5 channels is input into a bottleneck fully connected gated network. It first undergoes a non-linear feature transformation through a dimensionality reduction layer, and is then mapped to the $(0, 1)$ interval via a dimensionality expansion layer and a Sigmoid function, ultimately yielding the adaptive weights for each channel:

$$s = \sigma(W_2 \cdot \text{ReLU}(W_1 \cdot z)) \quad (10)$$

Where $z \in \mathbb{R}^{5 \times 1}$ is the global aggregated vector, $W_1 \in \mathbb{R}^{2 \times 5}$ and $W_2 \in \mathbb{R}^{5 \times 2}$ are the weight matrices of the dimensionality-reduction and dimensionality-expansion layers, respectively; and $\sigma(\cdot)$ denotes the Sigmoid activation function. The resulting adaptive weight vector satisfies $s \in (0, 1)^{5 \times 1}$, with each element corresponding to one feature channel.

The learned channel weights are multiplied element-wise with their corresponding original time-series features to accomplish adaptive weighting:

$$\tilde{x}_c(t) = s_c \cdot x_c(t) \quad (11)$$

Following dynamic weighting, the four adjusted EWT modal features and the single scaled CWT envelope feature are channel-wise concatenated, culminating in a synthesized multi-channel feature tensor with dimensions of 5×1024 . By dynamically adapting individual channel contribution ratios in response to the operational noise level and underlying fault characteristics of the input signal, this fusion strategy significantly boosts the discriminative sensitivity of the multi-domain signatures while strictly maintaining feature dimensionality.

3. Theoretical Basis of the Model

3.1. Convolutional Neural Network

Owing to their local receptive field geometry and weight-sharing topology, convolutional neural networks (CNNs) exhibit inherent efficacy in spatial feature learning and parameter compactness [38]. Within the proposed framework, the CNN branch operates as a core feature abstraction module tasked with isolating localized transient shocks and inter-channel spatial correlations directly from the multi-channel fused input. Mathematically, let the input feature tensor F_{rec} be defined as:

$$F_{rec} \in \mathbb{R}^{B \times C \times 1024} \quad (12)$$

where F_{rec} denotes the input multi-channel fused feature tensor, $\mathbb{R}$ represents the real-valued domain, B is the batch size, C is the number of fused feature channels, and 1024 is the sequence length. For a single sample, the input size can be expressed as $(C, 1024)$.

During the forward propagation phase, the multi-channel feature tensor traverses an alternating sequence of 1D convolutional operations, ReLU activation functions, and 1D max-pooling layers. Specifically, 1D convolution captures localized temporal signatures, the ReLU function injects non-linear expressiveness, and max-pooling condenses spatial dimensionality while retaining essential salient features. Through the hierarchical cascading of these convolutional and pooling blocks, the raw multi-channel time-series signal is progressively transformed into rich, high-level abstraction representations. The detailed layer-wise feature mapping pipeline for an individual input sample is illustrated in Fig.2.

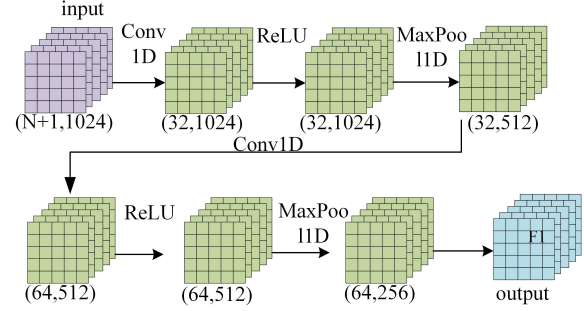


Fig. 2. CNN feature extraction architecture

3.2. Bidirectional Long Short-Term Memory Network

The bidirectional long short-term memory (BiLSTM) network integrates dual concurrent processing streams—a forward LSTM layer and a backward LSTM layer—to extract sequential signatures along both past and future temporal trajectories, respectively. By synthesizing past context with upcoming sequence trends, this architecture effectively maximizes temporal information utilization, thereby bolstering the network's capacity to represent non-stationary sequential dynamics [39].

Formally, let an input sequence be defined as $X = (x_1, x_2, \dots, x_L)$, where L denotes the sequence length. At any discrete time step t , the standard LSTM cell operation is governed by a forget gate f_t , an input gate i_t , a candidate cell state $\tilde{c}_t$, an updated cell state c_t , an output gate o_t , and a hidden state h_t . The step-by-step mathematical formulation is expressed as follows:

$$\begin{aligned} f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f) \\ i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i) \\ \tilde{c}_t &= \tanh(W_c x_t + U_c h_{t-1} + b_c) \\ c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \\ o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o) \\ h_t &= o_t \odot \tanh(c_t) \end{aligned} \quad (13)$$

where σ and $\tanh$ are activation functions; $\odot$ denotes element-wise multiplication; W , U , and b represent the weight matrices and bias terms of each gate, respectively; and c_t and

h_t denote the cell state and hidden state at time step t , respectively.

Specifically, the forward LSTM stream captures historical sequence dependencies along the forward timeline, whereas the backward LSTM stream extracts future contextual trends in reverse chronological order, as illustrated in the schematic diagram of the BiLSTM architecture in Fig. 3. By feature-wise concatenating the hidden state representations from both directions at each identical time step, a unified output vector containing comprehensive bidirectional global context is established. The joint state updates and output predictions are expressed as follows:

$$\begin{aligned} h_t^f &= \text{LSTM}_f(x_t, h_{t-1}^f, c_{t-1}^f) \\ h_t^b &= \text{LSTM}_b(x_t, h_{t+1}^b, c_{t+1}^b) \\ h_t &= [h_t^f; h_t^b] \\ \hat{y}_t &= \text{softmax}(W_y h_t + b_y) \end{aligned} \quad (14)$$

where h_t^f and h_t^b denote the hidden states of the forward LSTM and backward LSTM at time step t , respectively; c_{t-1}^f and c_{t+1}^b denote the cell states of the forward LSTM and backward LSTM at adjacent time steps, respectively; $[\cdot; \cdot]$ represents the vector concatenation operation; W_y and b_y denote the weight matrix and bias term of the output layer, respectively; and $\hat{y}_t$ represents the prediction output normalized by the softmax function.

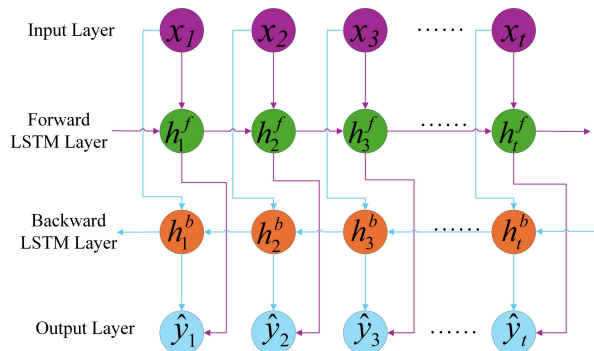


Fig. 3. Schematic diagram of the BiLSTM structure

3.3. Convolutional Block Attention Module

The convolutional block attention module (CBAM) represents a computationally efficient and widely applicable dynamic attention framework tailored for feed-forward convolutional architectures. As illustrated in the structural layout in Fig. 4, CBAM sequentially infers spatial and channel attention weight maps across two decoupled structural axes [40]. This dual-stage feature refinement paradigm adaptively directs feature allocation toward critical "what" (channel-wise semantic content) and "where" (spatial-wise target locations) representations, thereby significantly boosting the network's discriminative representation capacity with minimal computational overhead.

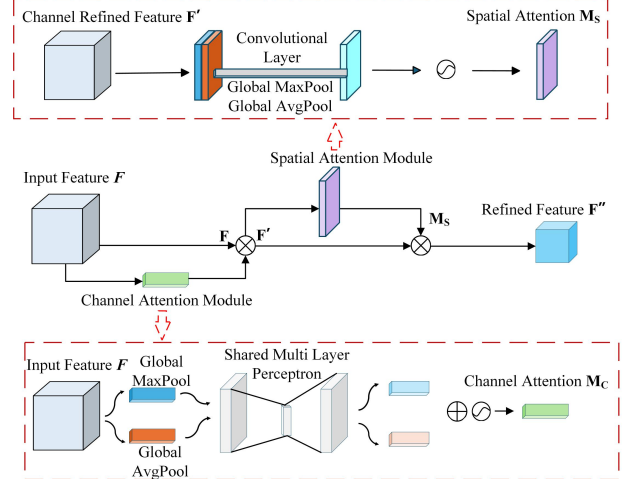


Fig. 4. Schematic diagram of the CBAM structure

Specifically, the channel attention branch dynamically isolates channel-wise salient features by applying dual global pooling operations—namely global average pooling and global max pooling—to the input feature tensor along its spatial dimensions, thereby yielding dual spatial context vectors. Subsequently, these vectors are fed into a

shared multi-layer perceptron (MLP) featuring a bottleneck architecture. The resulting outputs are element-wise aggregated and normalized via a Sigmoid activation function to generate the final 1D channel attention weight vector. The corresponding mathematical formulation proceeds as follows:

$$\begin{cases} F_{\text{avg}}^{\text{channel}} = \text{AvgPool}(F) \\ F_{\text{max}}^{\text{channel}} = \text{MaxPool}(F) \\ M_c = \sigma \left(\text{MLP} \left(F_{\text{avg}}^{\text{channel}} \right) + \text{MLP} \left(F_{\text{max}}^{\text{channel}} \right) \right) \\ F' = F \otimes M_c \end{cases} \quad (15)$$

Concurrently, the spatial attention mechanism is tailored to prioritize information-rich spatial regions while dampening task-irrelevant background noise. Operating on the channel-refined feature map F' , average pooling and max pooling operations are executed across the channel axis, producing dual single-channel 2D spatial feature descriptors. These two descriptors are subsequently concatenated along the channel dimension and processed by a standard 1D/2D convolutional layer to generate the spatial attention weight map M_s . The step-by-step mathematical formulation is expressed as follows:

$$\begin{cases} F_{\text{avg}}^{\text{spatial}} = \frac{1}{C} \sum_c F_c \\ F_{\text{max}}^{\text{spatial}} = \max_c F_c \\ M_s = \sigma \left(\text{Conv} \left(\left[F_{\text{avg}}^{\text{spatial}}; F_{\text{max}}^{\text{spatial}} \right] \right) \right) \\ F'' = F' \otimes M_s \end{cases} \quad (16)$$

where F denotes the initial input feature map; C is the total number of channels; $F_{\text{avg}}^{(\cdot)}$ and $F_{\text{max}}^{(\cdot)}$ represent the feature descriptors generated by average pooling and max pooling, respectively; MLP denotes the shared multilayer perceptron; σ is the sigmoid activation function; and $\otimes$ and $[\cdot; \cdot]$ denote element-wise multiplication and channel concatenation, respectively. M_c and

M_s are the inferred channel and spatial attention weight maps, respectively; F' and F'' represent the output features after channel refinement and dual refinement, respectively.

3.4. Parallel CNN-BiLSTM-CBAM Diagnosis Framework

Aiming to tackle the data leakage issue caused by overlapping sliding window slicing, as well as the insufficient diagnostic capability under harsh conditions involving strong background noise, limited fault samples and imbalanced class distribution, this paper proposes a parallel CNN-BiLSTM-CBAM fault diagnosis framework based on a strict temporally isolated partitioning strategy (As shown in Fig. 5).

First, the original continuous vibration signals are split into three non-overlapping subsets, namely the training, validation and test segments, in strict chronological order. After the partitioning process, random sample elimination is applied to specific fault categories to simulate the real-world scenarios of insufficient samples and class imbalance. For feature construction, a dual-branch EWT-CWT extraction mechanism is established: the EWT branch decomposes the input signal into four IMF components to characterize fine-grained fault details in local frequency bands, while the CWT branch derives one global energy envelope feature to reflect the overall time-frequency variation trend of the vibration signal.

Subsequently, a gated adaptive multi-domain feature fusion mechanism is devised. For the five feature channels generated from the above dual branches, global average pooling is performed along the temporal dimension first, so as to condense the fault information richness of each channel into a global aggregated vector. This vector is then fed into a bottleneck fully connected gating network, which sequentially performs non-linear feature transformation through a

dimensionality reduction layer and a dimensionality expansion layer, and maps the transformed values into the (0, 1) interval via a Sigmoid function to obtain the adaptive weight for each channel. The learned weights are multiplied element-wise with their corresponding original time-series features to fulfill adaptive channel weighting, and all weighted features are finally concatenated along the channel dimension to form a multi-channel fused feature matrix with the dimension of 5×1024 . Without increasing the total feature dimensionality, this mechanism can dynamically adjust the contribution ratio of each channel according to the noise intensity and fault type, thereby achieving the amplification of effective fault features and the suppression of noise components.

In the parallel recognition network, the CNN branch extracts local impulsive features

and inter-channel correlation information, while the BiLSTM branch implements bidirectional temporal dependency modeling. After fusing the output features from the two branches, the CBAM module further enhances critical fault features and suppresses noise interference through the joint effect of channel attention and spatial attention. Finally, fault diagnosis results are output through a global average pooling layer and a Softmax classifier. By integrating temporally isolated data partitioning, gated adaptive multi-domain feature extraction and parallel deep network architecture, the proposed method effectively improves the diagnostic accuracy, anti-interference robustness and generalization performance under complex operating conditions.

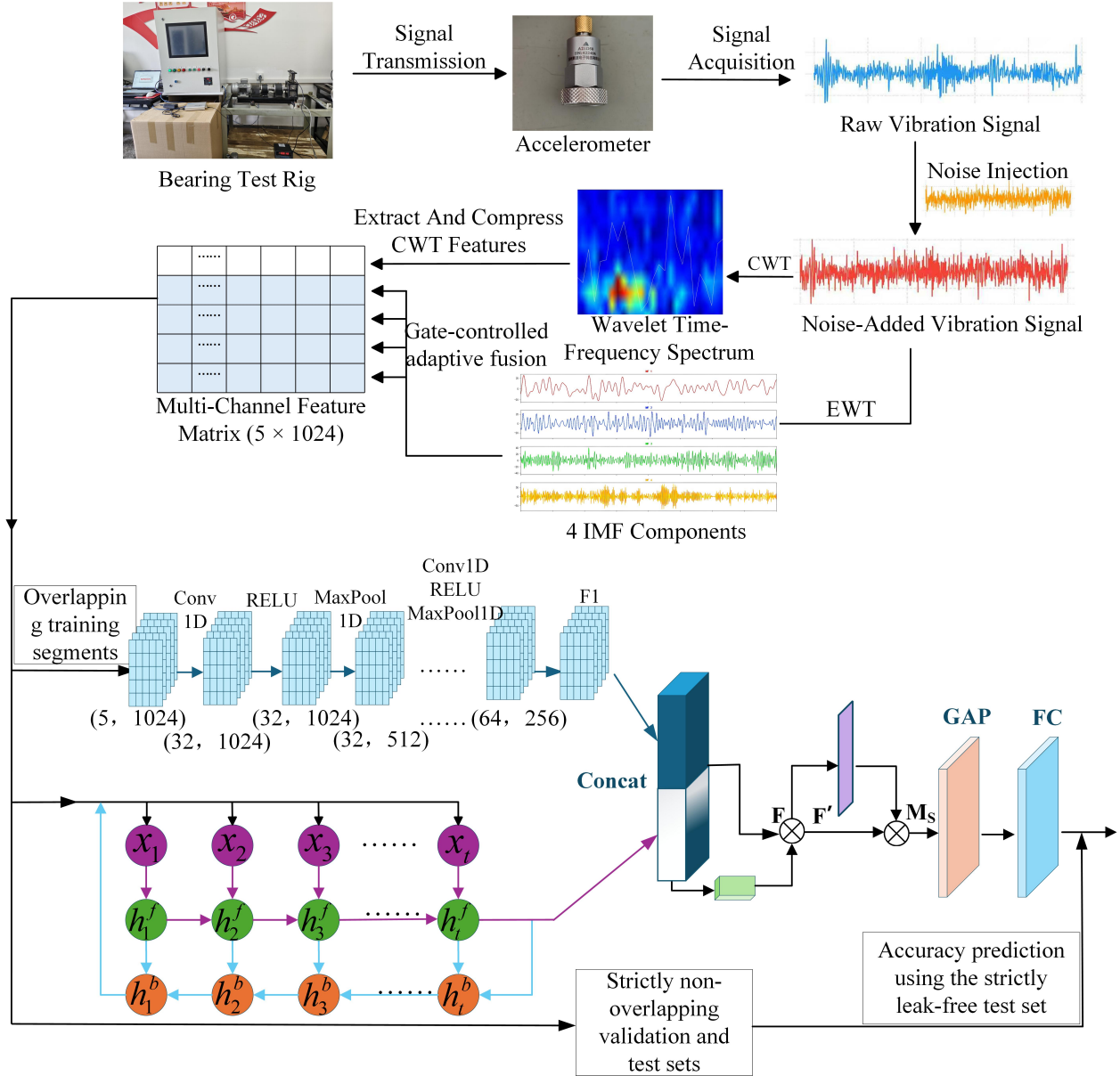


Fig. 5. Parallel CNN-BiLSTM-CBAM fault diagnosis process

To guarantee the reproducibility of the proposed fault diagnosis approach and clarify its complete architectural design, the core structural configurations of the parallel CNN-BiLSTM-CBAM network are elaborated as follows.

The CNN branch follows a VGG-inspired one-dimensional architecture with a gradually increasing number of channels, which consists of three consecutive convolutional blocks. Each convolutional layer utilizes small-size convolution kernels,

which helps cut down the total number of trainable parameters while enabling a deeper network structure. Both the stride and padding are uniformly configured as 1, so as to maintain the original temporal resolution of the input features. A max-pooling layer is attached right after each convolutional layer to perform $2\times$ downsampling on the feature sequences. For the temporal feature extraction branch, a single bidirectional long short-term memory (BiLSTM) layer is deployed, and the number of hidden units is

set to 120. In addition, a one-dimensional convolutional block attention module (1D-CBAM) is embedded after the spatial feature layer, with a channel reduction ratio of 16 and a spatial convolution kernel size of 7. The output dimension of the final fully connected layer is set to 12, which matches the classification requirement of 12 distinct bearing operating conditions.

4. Bearing Fault Experiment Design

4.1. Structure of the Bearing Test Rig

The experiment is conducted on a self-developed bearing fault simulation test platform, and the physical layout and core components of the platform are shown in Fig. 6. In terms of physical architecture, the platform is supported by a rigid welded base, on which the drive motor, coupling, main shaft, and test bearing housing are connected in sequence. During the experiment, the target rotational speed is set by the host computer and then transmitted to the PLC control cabinet to drive the rotor system. A

Table 1.Geometric specifications for the 31306 bearing model

Specification	Inside Diameter/mm	Outside Diameter/mm	Width/mm	Width of Inner Race/mm	Outer Ring Width/mm	Contact Angle/°
31306	30	72	20.5	19	14	28.8

In practical applications, vital parts of rolling bearings — including the inner ring, outer ring, and rolling bodies — frequently experience localized damage due to prolonged alternating stresses, wear, and contact fatigue. To accurately replicate authentic operating failures, defects on both the inner ring and rolling components were introduced utilizing wire-cut electrical discharge machining. Conversely, an angle grinder was applied to create spalling on the outer ring. Furthermore, a healthy bearing free from visible wear served as the baseline control. The actual appearances of the healthy bearing alongside the three

radial load is applied to the test bearing seat through a bolt-loading mechanism and monitored by a pressure sensor. Vibration signals are captured by a high-sensitivity accelerometer, then converted from analog signals to digital signals through an ART data acquisition card, and finally recorded by the computer for data storage and analysis.

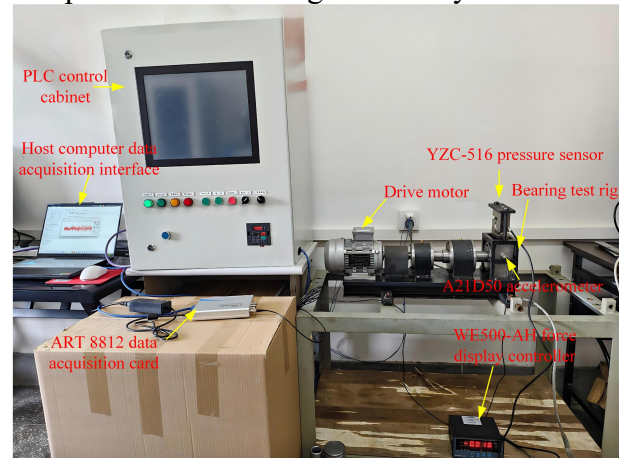


Fig. 6. Bearing Test Rig

The test bearing installed in the bearing housing is a 31306 tapered roller bearing, and its structural parameters are listed in Table 1.

distinctively damaged models are illustrated in Fig. 7.



Fig. 7. Physical photographs of the normal bearing and three typical faulty bearings

4.2. Setup of the Experiment and Signal Collection

The dataset utilized in this study comprises vibratory responses acquired from 31306 tapered roller bearings. We evaluated four distinct structural conditions: a healthy baseline, alongside localized defects on the inner ring, outer ring, and rolling components. To simulate the operating characteristics of bearings under different industrial conditions, three speed–load combinations are designed: 800 r/min + 3000 N, corresponding to low-speed heavy-load operation; 1100 r/min + 2000 N, corresponding to medium-speed medium-load operation; and 1400 r/min + 1000 N, corresponding to high-speed light-

load operation. As a result, 12 operating-condition labels are formed.

An ART 8812 data acquisition (DAQ) board was employed to digitize the analog outputs from the sensor. These digitized signals were subsequently forwarded to a master PC for real-time monitoring, archiving, and further analytical processing. Throughout this procedure, data was recorded at a continuous sampling rate of 25.6 kHz.

Samples were constructed using a time-series window of 1024 points. The original continuous vibration signals were first divided into mutually non-overlapping training, validation, and test segments at a ratio of 7:2:1. Within the training segment, samples were generated using a sliding window with 50% overlap, whereas the validation and test segments were sequentially partitioned without overlap. After time-isolated slicing, the initial sample size was kept consistent across all operating-condition classes, with each class containing 279 training samples, 80 validation samples, and 40 test samples. On this basis, random undersampling was applied to selected classes to simulate class imbalance under practical industrial conditions, using a fixed random seed of 7. Samples from each class were removed according to predefined proportions, and the corresponding reduction ratios together with the resulting numbers of training, validation, and test samples are listed in Table 2.

Table 2. Class-imbalance processing results for the self-developed bearing fault dataset

Operational Scenario	Failure Mode	Label	Removal ratio (%)	Training set	Validation set	Test set
Low-speed heavy-load	Inner-race fault	0	12.28	245	70	35
	Outer-race fault	1	64.91	98	28	14
	Rolling-element fault	2	44.86	154	44	22
	Normal condition	3	42.36	161	46	23

Medium-speed medium-load	Inner-race fault	4	7.27	259	74	37
	Outer-race fault	5	42.36	161	46	23
	Rolling-element fault	6	14.79	238	68	34
	Normal condition	7	7.27	259	74	37
High-speed light-load	Inner-race fault	8	9.77	252	72	36
	Outer-race fault	9	12.28	245	70	35
	Rolling-element fault	10	12.28	245	70	35
	Normal condition	11	47.37	147	42	21

4.3. Data Construction Protocols and Overlapping-Window Leakage Analysis

4.3.1. Problem Statement and Motivation

In deep learning-based rolling bearing fault diagnosis, the reliability of performance evaluation depends not only on the network architecture but also on how the dataset is constructed and partitioned. Many existing studies report the window length, overlap ratio, and train/validation/test split proportions, while the order of sample generation and subset partitioning is often not explicitly specified. Specifically, continuous vibration records may either be partitioned into temporally isolated subsets before sample generation, or first segmented into overlapping windows that are subsequently shuffled and randomly assigned to different subsets. These two procedures are not equivalent. In the latter case, highly overlapping samples derived from the same raw signal segment may appear in both the training and test sets, introducing potential cross-subset information leakage and consequently inflating the measured test performance.

Let a continuous vibration record of length N be denoted as:

$$\mathbf{x} = [x_1, x_2, \dots, x_N] \quad (17)$$

Using a sliding window of length L and step size s , the k -th sample is generated as:

$$\mathbf{x}^{(k)} = [x_1 + (k-1)s, x_2 + (k-1)s, \dots, x_L + (k-1)s] \quad (18)$$

$k = 1, \dots, K$

The overlap ratio is defined as:

$$\rho = \frac{L-s}{L} = 1 - \frac{s}{L} \quad (19)$$

Based on the sequence of data segmentation and subset partitioning, two data construction protocols are formally defined as follows.

Protocol A — Overlapping slicing followed by random partitioning.

Overlapping windows are first generated from the complete continuous record. The resulting samples are then randomly shuffled and divided into the training, validation, and test subsets. Since two adjacent windows share $L-s$ original sampling points, highly correlated samples may be assigned to different subsets.

Protocol B — Time-isolated slicing.

The continuous record is first divided into mutually non-overlapping temporal blocks corresponding to the training, validation, and test subsets. Sample windows are then generated independently within each block. Consequently, samples from different subsets do not share raw sampling points through window overlap, thereby avoiding the overlap-induced cross-subset leakage pathway considered in this study.

Unless otherwise specified, Protocol B is adopted throughout the subsequent experiments. To quantify the influence of the data construction procedure, the test-accuracy overestimation associated with Protocol A is defined as: $\Delta = A_{CC_A} - A_{CC_B}$.

4.3.2. Overlapping-Window Leakage Analysis

Consider two adjacent overlapping samples, $\mathbf{x}^{(k)}$ and $\mathbf{x}^{(k+1)}$, extracted from the same continuous vibration record. When $s < L$, the two samples share $L - s$ original sampling points, and the proportion of shared information increases with the overlap ratio ρ . Therefore, a smaller step size s produces greater overlap and stronger similarity between neighboring samples.

Under Protocol A, these highly correlated samples are randomly shuffled before subset partitioning. Consequently, samples containing partially identical raw signal segments may be assigned to the training and test sets simultaneously. A deep learning model may therefore exploit signal patterns already encountered during training when classifying such test samples. In this case, the measured test accuracy may partly reflect the memorization of shared signal segments rather than generalization to temporally independent data, leading to an overestimation of diagnostic performance.

In contrast, Protocol B performs subset partitioning before sample generation. Since the temporal blocks assigned to different subsets do not overlap, the overlap-induced sharing of raw sampling points between the training and test sets is avoided. The resulting test performance therefore provides a more reliable indication of model generalization to temporally isolated data.

Accordingly, the test-accuracy overestimation introduced by Protocol A can be attributed to the overlap-induced cross-subset information leakage during the sample construction and partitioning process. When overlapping samples generated from the same continuous vibration record are randomly assigned to different subsets, correlated information may be implicitly shared between the training and test sets, resulting in overly optimistic evaluation results. This analysis provides a theoretical

explanation for the 1.01%–9.03% test-accuracy overestimation experimentally observed in Section 5.1.

To further verify this effect empirically, Protocol B is adopted as the reference data construction procedure, while Protocol A is evaluated using six representative diagnostic architectures under the same experimental conditions.

4.4. CWRU Public Bearing Database

To thoroughly assess the generalizability of the developed algorithm, the open-source CWRU bearing repository [41] was utilized for benchmarking purposes. A visual representation of this test rig is provided in Fig.8.

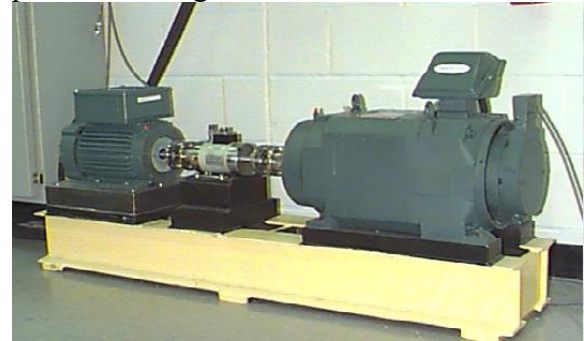


Fig. 8. Physical layout of the CWRU test rig

For this phase of testing, the analysis relied on vibratory data collected from the drive-end bearing at a sampling rate of 12 kHz. The collected records encompass four operational modes: a healthy state, along with localized defects situated on the inner ring, outer ring, and rolling bodies. By examining three distinct severity levels for every defect category, the classification task was structured around a total of ten unique structural health conditions. To maintain strict experimental parity with the aforementioned custom database, identical procedures were applied to the CWRU records regarding the generation of data instances, the allocation of data splits, and the artificial injection of noise. A detailed breakdown of the instance counts per class can be found in Table 3.

Table 3. Dataset partitioning results for the CWRU bearing fault dataset

Fault type	Damage size (mm)	Label	Training set	Validation set	Test set
Normal condition	-	0	140	40	20
Inner-race fault	0.177 8	1	191	34	17
	0.355 6	2	168	48	24
	0.533 4	3	126	36	18
	0.177 8	4	175	50	25
Outer-race fault	0.355 6	5	126	36	18
	0.533 4	6	56	16	8
	0.177 8	7	182	52	26
Rolling-element fault	0.355 6	8	147	42	21
	0.533 4	9	105	30	15

4.5. Experimental Software Environment

To ensure fair evaluations, identical training configurations were applied to both the custom and CWRU databases, with specific hyperparameters detailed in Table 4.

Table 4. Hyperparameter selections for the designed architecture

Hyperparameters	Parameter
Batch size	32
Epoch	20
Activation function	ReLU
Optimize	Adam
Learning rate	0.0003

All models were developed using Python 3.9.21 and the PyTorch framework on a 64-bit Windows 11 machine (CUDA 12.1). The physical workstation featured an Intel Core i7-14650HX CPU, an NVIDIA RTX 4060 Laptop graphics card, and 16 GB of RAM. Additionally, a global random seed of 7 was fixed to guarantee experimental reproducibility.

5. Empirical Results and Discussion

5.1. Performance Evaluation via the Custom Bearing Database

This section assesses the presented algorithm utilizing the custom-built failure database. To comprehensively gauge its classification precision, resilience, and latent feature extraction abilities, the testing framework incorporates four distinct phases: comparative and ablation studies, time-isolated slicing, confusion matrix examination, and visualization of feature clusters.

5.1.1. Ablation and Comparative Experiment Analysis

To substantiate the advantages of the developed architecture in identifying mechanical defects, incremental ablation studies were executed while maintaining strictly identical hardware setups and signal conditioning steps. Taking a basic single CNN model as the baseline, BiLSTM and the CBAM attention mechanism are gradually introduced to quantitatively compare the performance differences among different models and verify the contribution of each introduced module to diagnostic accuracy. Subsequently, horizontal comparative

experiments are carried out between the proposed model and several representative models, including the mainstream 1D-ResNet18 model, the temporal modeling model CNN-Transformer, and the attention-enhanced CNN-Transformer-CA model.

Considering the class-imbalanced setting of the present study, Precision, Recall, and F1-score were calculated using macro averaging, in which the corresponding metric was first calculated independently for each class and then averaged with equal weight across all classes. Accordingly, these metrics are reported as Macro-Precision, Macro-Recall, and Macro-F1. For the single-label multiclass classification task considered in this study, Balanced Accuracy is equivalent to Macro-Recall because both are defined as the arithmetic mean of the class-wise recalls. Therefore, Balanced Accuracy is not listed separately to avoid redundant information. All metrics in Table 5 were independently

calculated for each random seed and are reported as mean $\pm$ standard deviation.

Except for the multi-seed repeated experiments reported in Table 5, the subsequent auxiliary analyses were conducted under consistent experimental conditions using a single controlled run with a fixed random seed of 7. Specifically, Fig. 9, the training-time results in Table 6, Table 7, Fig. 10, and Fig. 11 were obtained from this fixed-seed run. The parameter counts in Table 6 are determined by the network architectures and are therefore independent of the random seed. These single-run results are mainly used to analyze the training process, computational cost, data-partitioning strategy, and class-level visualization, whereas the overall quantitative performance and inter-model comparisons are based on the multi-seed mean $\pm$ standard deviation reported in Table 5.

Table 5. Ablation and comparison on the custom dataset

Model	Accuracy (%)	Macro-Precision (%)	Macro-Recall (%)	Macro-F1
CNN	77.11 $\pm$ 1.87	80.65 $\pm$ 1.76	78.41 $\pm$ 1.56	79.75 $\pm$ 1.12
CNN-BiLSTM	92.23 $\pm$ 1.24	92.85 $\pm$ 1.53	91.85 $\pm$ 1.67	91.98 $\pm$ 1.23
CNN-BiLSTM-CBAM	96.63 $\pm$ 0.53	96.68 $\pm$ 0.21	96.73 $\pm$ 0.42	96.48 $\pm$ 0.45
1D-ResNet18	81.74 $\pm$ 1.78	78.91 $\pm$ 1.34	80.05 $\pm$ 1.51	79.65 $\pm$ 1.22
CNN-Transformer-CA	86.38 $\pm$ 1.12	87.56 $\pm$ 1.39	85.28 $\pm$ 1.98	84.73 $\pm$ 1.48
CNN-Transformer	89.74 $\pm$ 1.45	89.61 $\pm$ 1.62	90.20 $\pm$ 1.10	90.29 $\pm$ 1.03

As shown in Table 5, the evaluation metrics exhibit a clear improvement trend along the ablation sequence from CNN to CNN-BiLSTM and further to CNN-BiLSTM-CBAM. The proposed CNN-BiLSTM-CBAM model achieves a mean accuracy of 96.63 $\pm$ 0.53%, representing improvements of 19.52 and 4.40 percentage points over the baseline CNN and CNN-BiLSTM models, respectively. Meanwhile, its Macro-Precision, Macro-Recall, and Macro-F1 reach 96.68 $\pm$ 0.21%, 96.73 $\pm$

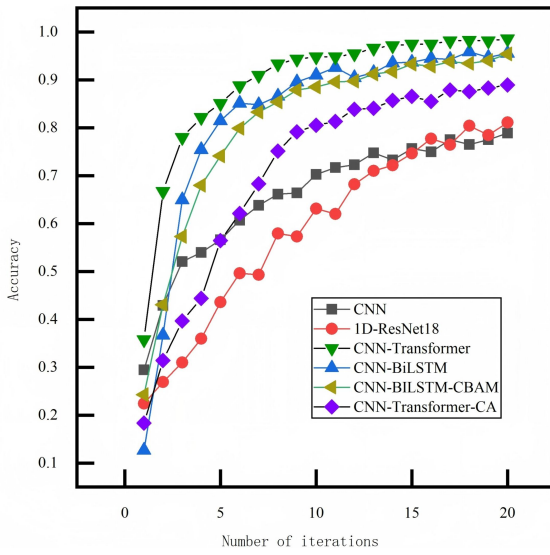
0.42%, and 96.48 $\pm$ 0.45%, respectively. These results indicate that the bidirectional temporal modeling capability of BiLSTM and the key-feature enhancement provided by CBAM further improve the overall classification performance on the basis of the local feature extraction capability of CNN.

In the horizontal comparison, the mean accuracy of the proposed model exceeds those of 1D-ResNet18, CNN-Transformer-CA, and CNN-Transformer by 14.89, 10.25, and 6.89 percentage points, respectively.

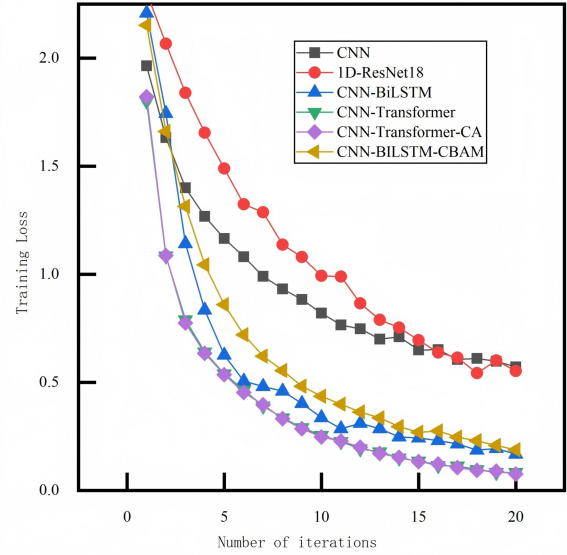
Among these three comparison models, CNN-Transformer achieves the highest mean accuracy of $89.74 \pm 1.45\%$, but it remains lower than that of the proposed model. Meanwhile, the proposed model also achieves higher Macro-Precision, Macro-Recall, and Macro-F1 than the above comparison models. These results indicate that the local impulsive feature extraction capability of CNN, the bidirectional temporal dependency modeling of BiLSTM, and the attention-based enhancement of CBAM provide complementary contributions,

enabling the model to maintain strong overall diagnostic performance under noisy and class-imbalanced conditions.

To additionally substantiate the training consistency and broad applicability of our architecture, we mapped the progression of loss and accuracy trajectories for the evaluated models. This visual representation offers clear insights into the convergence efficiency, training stability, and overfitting risk of the CNN-BiLSTM-CBAM model. The detailed graphical plots of these learning phases are displayed in Fig. 9.



(a) Accuracy



(b) Training Loss

Fig. 9. Training accuracy and loss curves

As shown by the training process, with the increase in the number of iterations, the training accuracy of each model gradually improves, while the training loss continuously decreases. However, the convergence behaviors of different models are not the same. The basic CNN and 1D-ResNet18 models converge relatively slowly. After 20 iterations, their training accuracies are 78.89% and 81.10%, respectively, while their loss values remain at approximately 0.572 and 0.553. This indicates that these two models have relatively weak feature learning capability for noisy and class-imbalanced

samples. After the introduction of BiLSTM or Transformer modules, the convergence performance of the models is significantly improved. Among them, CNN-Transformer achieves the highest training accuracy on the training set, reaching 98.55%, with the training loss reduced to 0.076, indicating its strong fitting capability for the training samples.

The proposed parallel CNN-BiLSTM-CBAM model achieves a final training accuracy of 95.35%, with the training loss decreasing to 0.189, and the overall training process remains relatively stable. Although

its training accuracy is not the highest in this fixed-seed run, the multi-seed test results reported in Table 5 show that the proposed model achieves a higher mean test accuracy than CNN-Transformer. This indicates that the proposed method does not merely pursue a better fit to the training set, but instead provides stronger overall performance in bearing fault diagnosis.

To evaluate the comprehensive effectiveness of the algorithm formulated

herein, and to balance the improvement in fault diagnosis accuracy against the reasonable computational burden, statistics on trainable parameter counts and the cumulative execution duration across 20 epochs were collected for all models involved in the ablation experiments. The comparison results are summarized in Table 6.

Table 6. Comparison of parameter count and total time consumption of 20 cycles for ablation models

Model	Trainable Parameters/k	Total Time for 20 Epochs /s
CNN	82.34	83
CNN-BiLSTM	110.42	96
CNN-BiLSTM-CBAM	144.54	118
1D-ResNet18	83.52	80.25
CNN-Transformer	124.6	104
CNN-Transformer-CA	134.8	115

Table 6 reports the number of trainable parameters and the total training time over 20 epochs for the compared models. The baseline CNN and 1D-ResNet18 have relatively small parameter sizes and lower training costs, whereas the introduction of BiLSTM, Transformer, and attention modules increases both model complexity and computational overhead. The proposed CNN - BiLSTM - CBAM model contains 144.54 k trainable parameters and requires 118 s for 20 training epochs, both of which are the largest among the compared models, indicating that the improvement in diagnostic performance is accompanied by additional computational cost. This overhead mainly arises from the recurrent temporal modeling of BiLSTM and the feature recalibration introduced by the attention module. Therefore, the current experiments primarily demonstrate the effectiveness of the proposed model in terms of diagnostic performance and robustness, but are not sufficient to conclude that it is better suited for resource-

constrained edge devices or highly time-sensitive applications. Future work will focus on reducing computational requirements through model pruning, quantization, and lightweight architectural design, together with evaluations of inference latency and memory consumption on practical edge devices to further assess its engineering deployment potential.

5.1.2. Strict Time-Isolated Slicing

In bearing fault diagnosis studies, many papers only report the overlap ratio of the sliding window, the number of samples, or the partitioning ratio of the training, validation, and test sets, without clearly specifying the order between sample construction and dataset partitioning. In other words, it is often unclear whether the continuous signal is first divided into time segments and then sliced, or whether overlapping slicing is first performed and the generated samples are then randomly divided into different datasets. Since these two processing procedures differ significantly in

terms of sample independence and model evaluation results, it is often difficult to accurately judge the reliability of the test results based only on the overlap ratio or the number of samples. To address this issue, this paper adopts a time-isolated slicing strategy. This strategy can effectively avoid cross-dataset information leakage and make the model evaluation results closer to the diagnostic performance under real operating conditions.

To quantify the influence of different slicing strategies on diagnostic accuracy and further illustrate the effect of different sample

Table 7. Comparison of diagnostic accuracy of different models under time-isolated partitioning and conventional partitioning

Model	Accuracy under time-isolated partitioning (%)	Accuracy under conventional partitioning (%)
CNN	78.98	83.76
CNN-BiLSTM	93.47	96.77
CNN-BiLSTM-CBAM	97.16	98.17
1D-ResNet18	83.52	88.10
CNN-Transformer-CA	87.50	96.53
CNN-Transformer	91.19	96.35

The above comparative experimental results indicate that the conventional partitioning strategy leads to varying degrees of overestimation in the test accuracy of bearing fault diagnosis models. For all six models, the accuracies obtained under conventional partitioning are higher than those obtained under time-isolated partitioning, with the accuracy overestimation ranging from 1.01% to 9.03%. Among them, the CNN-Transformer-CA model shows the most obvious overestimation, reaching 9.03%, whereas the proposed parallel CNN-BiLSTM-CBAM model exhibits the smallest overestimation, at only 1.01% points. This indicates that the data construction procedure of overlapping slicing followed by random partitioning can

construction procedures on the test results, comparative experiments are conducted on six mainstream fault diagnosis models, including the proposed parallel CNN-BiLSTM-CBAM model. Under the condition that the network structure, training parameters, and optimization strategy are kept unchanged, only the sample construction and dataset partitioning methods are changed. The diagnostic results under different sample partitioning strategies are compared, as shown in Table 7.

easily introduce cross-dataset correlations, leading to overly optimistic test results. By contrast, time-isolated partitioning can more realistically reflect the generalization performance of the model on independent data.

5.1.3. Confusion Matrix Analysis

In bearing fault diagnosis, the confusion matrix is a core quantitative evaluation tool. Its value lies in overcoming the limitation of using only overall accuracy by quantifying the diagnostic performance of each individual fault category through the distributions of true positives, false positives, and false negatives. with visual representations provided in Fig. 10.

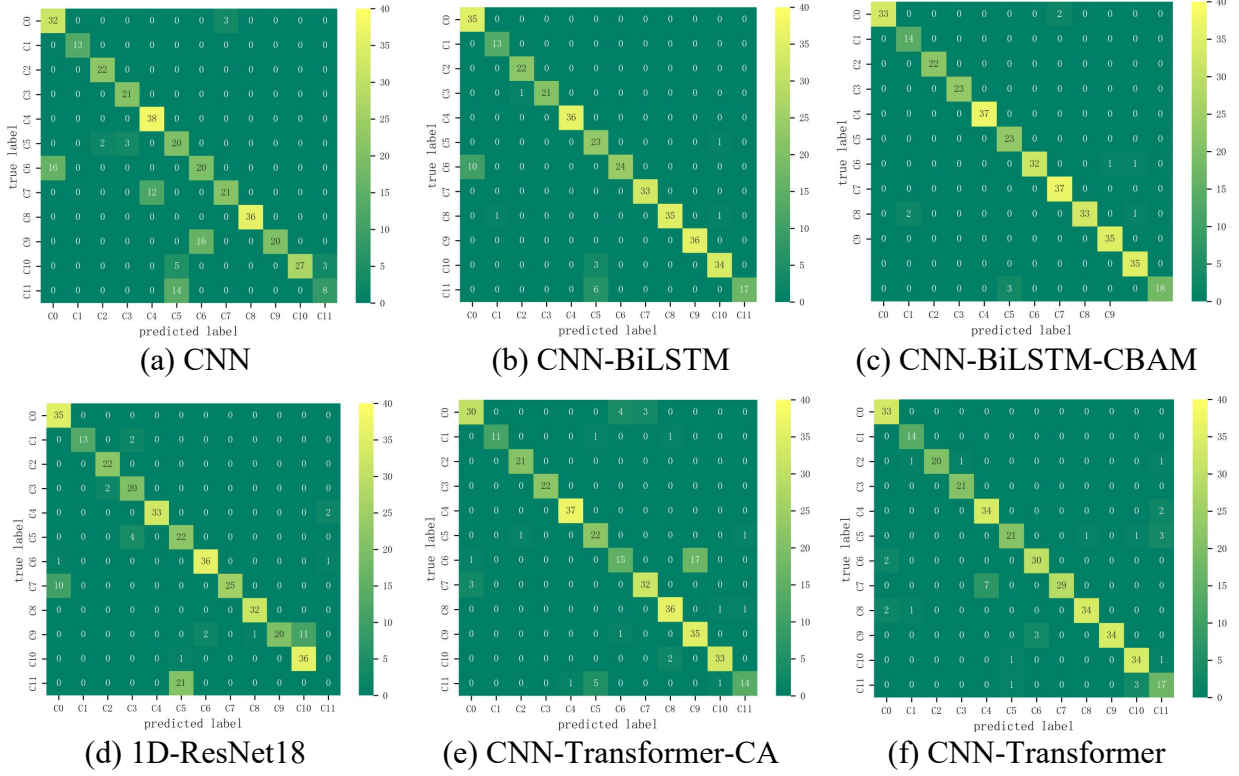


Fig. 10. Visual contrast of the confusion matrices generated by various architectures

According to the visual evidence in Fig. 10, the evaluated algorithms display distinct variations in their predictive capabilities across the class-imbalanced dataset. The basic CNN and 1D-ResNet18 models produce more misclassified samples in the off-diagonal regions, indicating that their ability to distinguish similar fault features remains limited under strong noise and class-imbalanced conditions. After introducing BiLSTM or Transformer modules, the models' ability to capture temporal dependency information is enhanced, and the concentration of samples along the diagonal of the confusion matrix is generally improved. However, the improvement varies across different network structures.

Conversely, the newly constructed parallel CNN-BiLSTM-CBAM architecture demonstrates superior diagnostic efficacy. An analysis of its classification matrix reveals that predictions heavily aggregate along the primary diagonal, yielding the lowest rate of off-diagonal misassignments.

Specifically, the network accurately categorizes 342 test instances, reaching a peak accuracy of 97.16%. The rare prediction errors that do occur are predominantly confined to highly similar defect classes. Such strong performance underscores the synergistic integration of the network's three core components: the CNN module's proficiency in isolating local transient impulses, the BiLSTM's capacity to trace bidirectional sequential patterns, and the CBAM framework's ability to amplify critical diagnostic signatures. Together, these modules significantly bolster the algorithm's robustness, enabling precise defect differentiation even when constrained by limited data availability, severe class skewness, and heavy background acoustic interference.

5.1.4. Feature Clustering Analysis

In bearing fault diagnosis tasks, the t-SNE feature clustering map visually presents the discriminability of different bearing states

through the clustering distribution of feature points. These extracted features are analyzed

below, as visualized in Fig. 11.

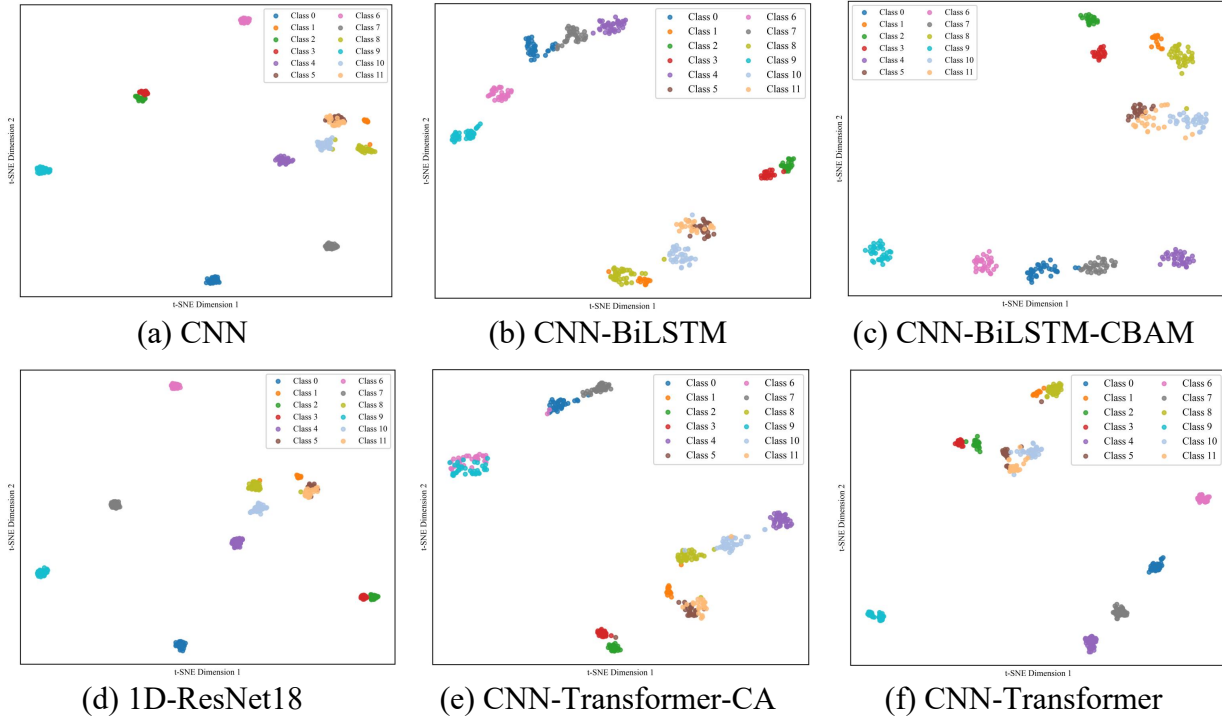


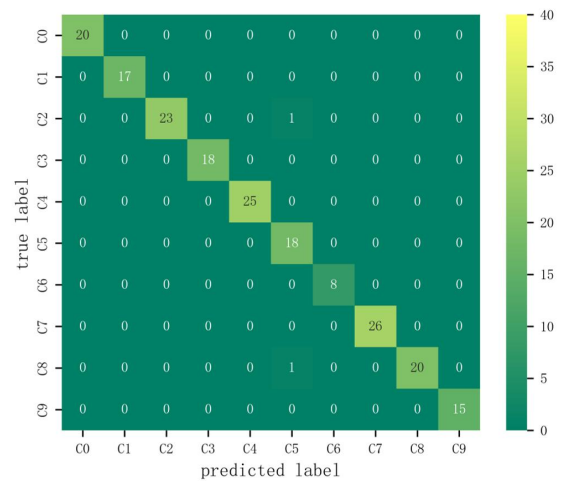
Fig. 11. Comparison of t-SNE feature clustering results of different models

The t-SNE feature clustering results from the ablation experiments show that Class8, Class9, and Class10 exhibit loose feature distributions and severe inter-class overlap in basic models such as CNN and 1D-ResNet18. Although the CNN-Transformer and CNN-Transformer-CA models improve feature compactness, feature overlap still exists among easily confused categories. The comparative experiments further demonstrate that, compared with all mainstream reference models, the proposed parallel CNN-BiLSTM-CBAM model achieves the most compact feature clusters, the clearest inter-class boundaries, and the least category confusion. It effectively resolves the feature mixing problem among similar faults and realizes accurate separation and classification of bearing fault features.

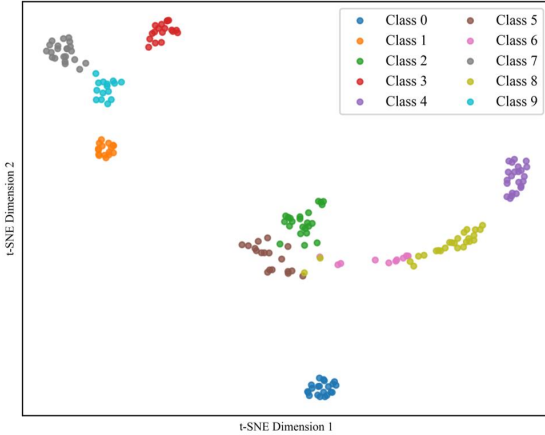
5.2. Application to the CWRU Dataset

The CWRU dataset was processed using the same data partitioning and preprocessing

procedures as the self-developed bearing fault dataset, including noise injection, class-imbalance construction, EWT - CWT feature extraction, and gated adaptive fusion. After multiple independent repeated experiments with different random seeds, the proposed model achieved a mean accuracy of $97.99 \pm 0.87\%$.



(a) Confusion Matrix



(b) Feature Clustering

Fig. 12. Diagnostic results of our architecture: (a) Confusion matrix (b) t-SNE feature clustering map

To further clarify the representativeness of the visualization results, Fig. 12 presents the confusion matrix and t-SNE results obtained from a single representative run, whereas the overall statistical performance of the model across multiple random seeds is reported as the mean $\pm$ standard deviation in

Table 8. As can be seen from Fig. 12, on the CWRU dataset, the parallel CNN-BiLSTM-CBAM model yields very few misclassified samples in the confusion matrix, and the predictions for each category are highly concentrated along the diagonal. In the t-SNE feature clustering results, the boundaries among different fault categories are clear, with no obvious feature overlap.

5.2.1. CWRU Ablation and Comparative Experimental Analysis

This section details comparative and ablation studies performed on the open-source CWRU repository to confirm the transferability of the designed algorithm. The same evaluation metrics and macro-averaging strategy defined in Section 5.1.1 were adopted for the CWRU dataset. Each model was independently trained and evaluated using multiple random seeds, and the results are reported as mean $\pm$ standard deviation in Table 8.

Table 8. Ablation and comparison results on CWRU

Model	Accuracy (%)	Macro-Precision (%)	Macro-Recall (%)	Macro-F1
CNN	59.61 $\pm$ 0.62	62.40 $\pm$ 0.72	62.11 $\pm$ 0.67	60.90 $\pm$ 0.44
CNN-BiLSTM	86.00 $\pm$ 0.12	84.21 $\pm$ 0.77	85.07 $\pm$ 0.14	84.98 $\pm$ 0.89
CNN-BiLSTM-CBAM	97.99 $\pm$ 0.87	96.40 $\pm$ 0.89	97.28 $\pm$ 0.78	96.50 $\pm$ 0.81
1D-ResNet18	50.46 $\pm$ 0.96	50.89 $\pm$ 1.32	46.13 $\pm$ 0.89	50.01 $\pm$ 0.71
CNN-Transformer	90.68 $\pm$ 1.73	91.17 $\pm$ 1.74	91.46 $\pm$ 0.87	91.20 $\pm$ 0.91
CNN-Transformer-CA	94.32 $\pm$ 1.21	93.56 $\pm$ 1.87	94.04 $\pm$ 1.32	94.99 $\pm$ 0.32

Table 8 presents the ablation and comparative results on the CWRU dataset. The overall trends are consistent with those observed on the self-developed dataset. Under the same preprocessing and training protocol, the proposed CNN - BiLSTM - CBAM model achieved relatively strong overall performance among the compared methods, indicating its diagnostic effectiveness on the CWRU public benchmark dataset.

5.3. Cross-Dataset Unsupervised Domain Adaptation Experiment

To further evaluate the adaptability of the proposed method under substantial domain shifts between different experimental platforms, a cross-dataset unsupervised domain adaptation experiment was conducted. It should be clarified that this setting is not a strict domain generalization scenario in which no target-domain information is used. During training, the class labels of the CWRU target-domain samples

were not used; however, a subset of unlabeled target-domain samples was used for CORAL-based feature distribution alignment. Therefore, the present experiment should be regarded as an unsupervised domain adaptation setting. Since the self-developed dataset contains 12 operating-condition classes whereas the CWRU dataset contains 10 classes, the original label spaces cannot be directly matched. Therefore, a shared label space was constructed using the four bearing health states common to both datasets: normal condition (0), inner-race fault (1), outer-race fault (2), and rolling-element fault (3).

For source-domain supervised training, only data from the self-developed dataset under the 800 r/min and 3000 N operating condition were used, with samples constructed according to the time-isolated slicing protocol. The trained model was subsequently evaluated on CWRU drive-end vibration data sampled at 12 kHz, covering the same four health states. To mitigate the domain discrepancy arising from differences in bearing type and sampling configuration, an unsupervised domain adaptation strategy was introduced. Specifically, CORAL loss was employed to align the feature covariance distributions between the source and target domains, while Gaussian-noise augmentation was applied to the source-domain training data to improve model robustness. No target-domain class labels were involved in supervised training, and the final diagnostic performance was evaluated on an independent CWRU target-domain test subset that was not involved in either supervised training or domain alignment.

Under this protocol, the CNN - BiLSTM - CBAM model achieved an accuracy of 98.75% on 800 CWRU test samples. As shown by the confusion matrix in Fig. 13, all normal and inner-race fault samples were correctly classified, while only limited confusion occurred between the

outer-race and rolling-element fault classes, with seven outer-race samples misclassified as rolling-element faults and three rolling-element samples misclassified as outer-race faults. The remaining performance degradation may primarily arise from domain discrepancies in bearing geometry and fault-severity definitions. Overall, without using target-domain class labels for supervised training, the proposed method retained good fault-discrimination capability across different bearing types and acquisition configurations through unsupervised domain alignment.

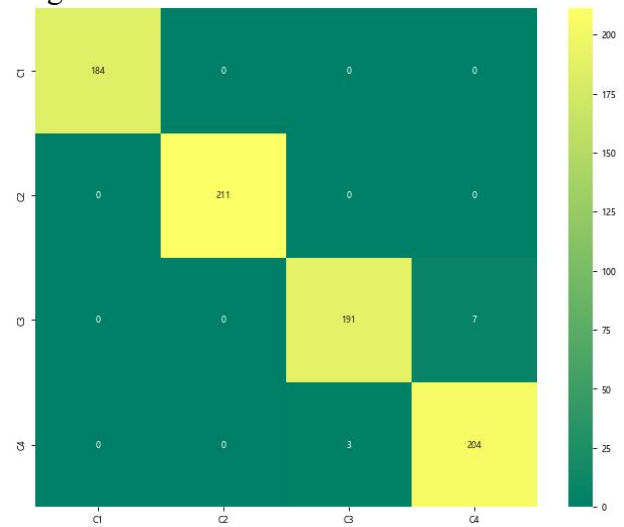


Fig. 13. Confusion matrix for cross-dataset unsupervised domain adaptation on the CWRU dataset

5.4. Fusion Mechanism Ablation Study

To further evaluate the contribution of the proposed gated adaptive fusion mechanism, four fusion strategies were compared under strictly identical experimental settings, including the same dataset, 20 training epochs, optimizer, batch size, and random seed of 7. Only the multi-domain feature fusion/weighting strategy was varied. The four schemes are described as follows:

Concatenation: Multi-domain feature channels are directly concatenated without

any weighting and are used as a naive baseline.

Equal-weight fusion: The same fixed weight is assigned to each feature channel before fusion.

SE-style attention: A conventional Squeeze-and-Excitation (SE) channel-attention mechanism is adopted for adaptive channel recalibration.

Proposed gated fusion: Global average pooling is first applied to the multi-domain

features, after which a lightweight bottleneck gating network generates channel-wise fault-relevance weights. These weights are applied directly to the original feature channels, enabling adaptive feature enhancement without increasing the feature dimensionality.

The ablation results are summarized in Table 9.

Table 9. Ablation results of different fusion mechanisms on the CWRU dataset

Fusion strategy	Accuracy (%)	Precision (%)	Recall (%)	F1/%
Concatenation	94.32	94.13	93.67	93.89
Equal-weight fusion	95.43	95.71	95.21	95.78
SE-style attention	97.34	96.99	97.67	97.24
Proposed gated fusion	98.86	97.29	97.95	97.83

As shown in Table 9, the proposed gated fusion mechanism achieves the best overall performance among the four strategies in terms of accuracy, precision, recall, and F1-score, with particularly clear advantages under the low-SNR condition of -2 dB. Compared with simple concatenation and equal-weight fusion, the proposed mechanism can adaptively emphasize fault-relevant channels while suppressing channels dominated by noise. Compared with SE-style attention, the proposed method performs channel-wise reweighting directly on the original feature representation without increasing the feature dimensionality. This lightweight design may help reduce overfitting under limited-sample and class-imbalanced conditions, thereby contributing to improved generalization performance. These results demonstrate the added value of the proposed gated adaptive fusion module.

5.5. Noise Robustness Analysis

To comprehensively evaluate the robustness of the proposed method under different noise conditions, experiments were conducted at four signal-to-noise ratio (SNR) levels: -2 dB, -4 dB, -6 dB, and -8 dB. Accuracy, precision, recall, and F1-score

were calculated at each SNR level, and the results are presented in Fig. 14.

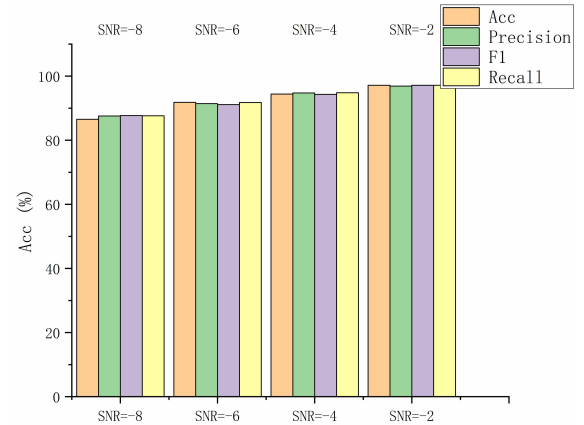


Fig. 14. Classification performance of the proposed method under different signal-to-noise ratio (SNR) levels. The orange, green, purple, and yellow bars represent accuracy, precision, F1-score, and recall, respectively.

As shown in Fig. 14, model performance decreases progressively as the SNR decreases. The four evaluation metrics reach their highest values at -2 dB, at approximately 98%. At -4 dB and -6 dB, the performance decreases to approximately 95% and 92%, respectively. Even under the strongest noise condition of -8 dB, the model still maintains a performance level of approximately 87%.

These results indicate that the EWT–CWT multi-domain feature extraction and gated adaptive fusion mechanisms can effectively suppress noise interference and maintain a relatively gradual and controlled performance degradation under increasingly severe noise conditions.

It is also worth noting that accuracy, precision, recall, and F1-score remain at comparable levels across all SNR settings, without showing substantial divergence among the metrics. This suggests that the proposed model maintains relatively balanced classification performance across different fault categories even under strong noise. The results further support the overall noise robustness of the proposed method across a range of SNR conditions, rather than only at a single noise level.

6. Conclusion

This paper proposes a rolling bearing fault diagnosis method based on time-isolated data construction, gated adaptive EWT–CWT multi-domain feature fusion, and a parallel CNN–BiLSTM–CBAM architecture. By partitioning continuous vibration records before sample generation, the method reduces the risk of cross-subset information leakage caused by overlapping windows, while the gated mechanism dynamically integrates local frequency-band and global time–frequency features.

Experimental results show that overlap-first random partitioning can lead to overestimated test performance. Under identical experimental conditions, the proposed gated fusion outperforms direct concatenation, equal-weight fusion, and SE-style attention, while maintaining stable diagnostic performance across different noise levels. The cross-dataset experiment further shows that the proposed method can retain effective fault-discriminative capability across different bearing types and acquisition conditions without using target-domain

labels for supervised training, indicating good noise robustness and cross-dataset adaptability.

Several limitations remain. The current validation is mainly based on laboratory datasets, the cross-dataset experiment is limited to four shared health states and uses unlabeled target-domain data for CORAL alignment, and the complete framework still introduces a certain computational cost. Future work will focus on more complex industrial noise and compound-fault conditions, domain-generalization strategies requiring less or no target-domain information, and lightweight implementation for online diagnosis.

References

- [1] Sun B, Sheng Z, Song P, Sun H, Wang F, Sun X and Liu J 2025 State-of-the-art detection and diagnosis methods for rolling bearing defects: a comprehensive review Appl. Sci. 15 1001
- [2] Zhang X, Zhao B and Lin Y 2021 Machine learning-based bearing fault diagnosis using the Case Western Reserve University data: a review IEEE Access 9 155598–155608
- [3] Chen X, Yang R, Xue Y, Huang M, Ferrero R and Wang Z 2023 Deep transfer learning for bearing fault diagnosis: a systematic review since 2016 IEEE Trans. Instrum. Meas. 72 1–21
- [4] Yan G, Chen J, Bai Y, Yu C and Yu C 2022 A survey on fault diagnosis approaches for rolling bearings of railway vehicles Processes 10 724.
- [5] Long W, Guang Y, Longxin H, Chunsheng Y and Ke F 2024 A new unsupervised health index estimation method for bearings early fault detection based on Gaussian mixture model Eng. Appl. Artif. Intell. 128 107562.

- [6] Bo P, Ying B, Bing X, Mengjie Z and Shuting W 2022 A survey on fault diagnosis of rolling bearings *Algorithms* 15 347.
- [7] Pandiyan M and Babu T N 2024 Systematic review on fault diagnosis on rolling-element bearing *J. Vib. Eng. Technol.* 12 8249–8283
- [8] Raj K K, Kumar S and Kumar R R 2025 Systematic review of bearing component failure: strategies for diagnosis and prognosis in rotating machinery *Arab. J. Sci. Eng.* 50 5353–5375
- [9] Ma J, Huang J, Liu S, Luo J and Jing L 2024 A self-attention Legendre graph convolution network for rotating machinery fault diagnosis *Sensors* 24 5475
- [10] Hakim M, Omran A A B, Ahmed A N, Al-Waily M and Abdellatif A 2023 A systematic review of rolling bearing fault diagnoses based on deep learning and transfer learning: taxonomy, overview, application, open challenges, weaknesses and recommendations *Ain Shams Eng. J.* 14 101945.
- [11] Bao Z, Liu C, Yang H, Zhang J and Li Y 2026 From theory to industry: a survey of deep learning-enabled bearing fault diagnosis in complex environments *Eng. Appl. Artif. Intell.* 163 113068
- [12] Song Y, Zeng S, Ma J and Guo J 2018 A fault diagnosis method for roller bearing based on empirical wavelet transform decomposition with adaptive empirical mode segmentation *Measurement* 117 266–276
- [13] Abdelkader R, Kaddour A and Derouiche Z 2018 Enhancement of rolling bearing fault diagnosis based on improvement of empirical mode decomposition denoising method *Int. J. Adv. Manuf. Technol.* 97 3099–3117
- [14] Li H, Wu X, Liu T, Li S, Zhang B, Zhou G and Huang T 2022 Composite fault diagnosis for rolling bearing based on parameter-optimized VMD *Measurement* 201 111637
- [15] Kankar P K, Sharma S C and Harsha S P 2011 Fault diagnosis of ball bearings using continuous wavelet transform *Appl. Soft Comput.* 11 2300–2312
- [16] Gao H, Liang L, Chen X and Xu G 2015 Feature extraction and recognition for rolling element bearing fault utilizing short-time Fourier transform and non-negative matrix factorization *Chin. J. Mech. Eng.* 28 96–105
- [17] Zhang S, Zhang S, Wang B and Habetler T G 2019 Machine learning and deep learning algorithms for bearing fault diagnostics: a comprehensive review *arXiv:1901.08247*
- [18] Lv Y, Liu Y, Li S, Liu J and Wang T 2025 Enhancing marine shaft generator reliability through intelligent fault diagnosis of gearbox bearings via improved Bidirectional LSTM *Ocean Eng.* 337 121860
- [19] Guo Y, Mao J and Zhao M 2023 Rolling bearing fault diagnosis method based on attention CNN and BiLSTM network *Neural Process. Lett.* 55 3377–3410
- [20] Wang D, Zhu W and Zhu Y 2026 Intelligent recognition and fault prediction of bearing vibration signals based on the CNN-LSTM hybrid model *Eng. Res. Express* 8 035502
- [21] Liu W, Liu S, He Y, Wang J and Gu Y 2025 Rolling bearing fault diagnosis based on MTF encoding and CBAM-LCNN mechanism *Comput. Mater. Contin.* 82 4863–4880

- [22] Xu S, Yuan R, Lv Y, Hu H, Shen T and Zhu W 2023 A novel fault diagnosis approach of rolling bearing using intrinsic feature extraction and CBAM-enhanced InceptionNet Meas. Sci. Technol. 34 105111
- [23] Zhang Q, Wei X, Wang Y and Hou C 2024 Convolutional neural network with attention mechanism and visual vibration signal analysis for bearing fault diagnosis Sensors 24 1831
- [24] Rauber T W, da Silva Loca A L, de Assis Boldt F, Rodrigues A L and Varejao F M 2021 An experimental methodology to evaluate machine learning methods for fault diagnosis based on vibration signals Expert Syst. Appl. 167 114022
- [25] Jiang X, Zhang Z, Yuan H, He J and Tong Y 2025 A DAE-BiLSTM-based fault diagnosis method for principal drive shaft bearings: a case study using Case Western Reserve University bearing data Processes 13 202
- [26] Cai C, Xu T, Ren J and Xue Y 2025 Bearing fault diagnosis based on the Markov transition field and SE-IShufflenetV2 model Struct. Durab. Health Monit. 19 125–144
- [27] Chen Q, Zhang F, Wang Y, Yu Q, Lang G and Zeng L 2025 Bearing fault diagnosis based on efficient cross space multiscale CNN transformer parallelism Sci. Rep. 15 12344
- [28] Song B, Liu Y, Fang J, Liu W, Zhong M and Liu X 2024 An optimized CNN-BiLSTM network for bearing fault diagnosis under multiple working conditions with limited training samples Neurocomputing 574 127284
- [29] Chang Y and Bao G 2024 Enhancing rolling bearing fault diagnosis in motors using the OCSSA-VMD-CNN-BiLSTM model: a novel approach for fast and accurate identification IEEE Access 12 78463–78479
- [30] Yuan B, Lei L and Chen S 2025 Optimized variational mode decomposition and convolutional block attention module-enhanced hybrid network for bearing fault diagnosis Machines 13 320
- [31] Gong L, Pang C, Wang G and Shi N 2024 Lightweight bearing fault diagnosis method based on improved residual network Electronics 13 3749
- [32] Zou Y, Zhang X, Liu T, Zhang Y, Li L and Zhao W 2025 Rolling bearing fault diagnosis based on cross-attention fusion WDCNN and BiLSTM Comput. Mater. Contin. 83 4699–4723
- [33] Varejao I M D S, de Oliveira Costa L G, Silva L H P D, Rodrigues A, Ribeiro M P, Varejao F M and Santos T O 2025 The similarity bias problem: what it is and how it impacts vibration based intelligent fault diagnosis Mech. Syst. Signal Process. 235 112822
- [34] Wang Y, Jiang H, Tong B and Song S 2025 Rolling bearing fault diagnosis via Meta-BOHB optimized CNN–Transformer model and time-frequency domain analysis Sensors 25 6920
- [35] Matania O, Dattner I, Bortman J, Kenett R S and Parmet Y 2024 A systematic literature review of deep learning for vibration-based fault diagnosis of critical rotating machinery: limitations and challenges J. Sound Vib. 590 118562
- [36] Kumar H S and Upadhyaya G 2023 Fault diagnosis of rolling element bearing using continuous wavelet transform and K-nearest neighbour Mater. Today Proc. 92 56–60

- [37] Gilles J 2013 Empirical wavelet transform IEEE Trans. Signal Process. 61 3999–4010
- [38] Wen L, Li X, Gao L and Zhang Y 2018 A new convolutional neural network-based data-driven fault diagnosis method IEEE Trans. Ind. Electron. 65 5990–5998
- [39] Cui Z, Ke R, Pu Z and Wang Y 2018 Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction arXiv:1801.02143
- [40] Woo S, Park J, Lee J-Y and Kweon I S 2018 CBAM: convolutional block attention module Proc. Eur. Conf. Comput. Vis. (ECCV) pp 3–19
- [41] Case Western Reserve University Bearing Data Center Bearing data center Available at: <https://engineering.case.edu/bearingdatacenter>