Pyramid Pooling-Based Vision Transformer for Tool Condition Recognition

ZHENG Kun[1*], *LI Yonglin*[1], *GU Xinyan*[1], *DING Zhiying*[1], *ZHU Haihua*[2]

1. School of Traffic Engineering, Nanjing Institute of Technology, Nanjing 211167, P. R. China;
2. College of Mechanical & Electrical Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P. R. China

(Received 16 April 2025; revised 3 June 2025; accepted 6 June 2025)

Abstract: This study focuses on tool condition recognition through data-driven approaches to enhance the intelligence level of computerized numerical control (CNC) machining processes and improve tool utilization efficiency. Traditional tool monitoring methods that rely on empirical knowledge or limited mathematical models struggle to adapt to complex and dynamic machining environments. To address this, we implement real-time tool condition recognition by introducing deep learning technology. Aiming to the insufficient recognition accuracy, we propose a pyramid pooling-based vision Transformer network (P2ViT-Net) method for tool condition recognition. Using images as input effectively mitigates the issue of low-dimensional signal features. We enhance the vision Transformer (ViT) framework for image classification by developing the P2ViT model and adapt it to tool condition recognition. Experimental results demonstrate that our improved P2ViT model achieves 94.4% recognition accuracy, showing a 10% improvement over conventional ViT and outperforming all comparative convolutional neural network models.

Key words: tool condition recognition; Transformer; pyramid pooling; deep convolutional neural network

CLC number: TN925 **Document code:** A **Article ID:** 1005-1120(2025)03-0322-15

0 Introduction

Tool condition recognition technology holds significant application value in modern manufacturing, particularly in enhancing machining accuracy, reducing cost, and extending tool life, where it plays an irreplaceable role. With the increasing complexity of machining processes, traditional manual inspection methods have struggled to meet the demands for efficiency and precision. Consequently, machine learning and signal processing-based tool condition recognition methods have gradually become a research hotspot[1-2]. By real-time monitoring and analyzing signals such as vibration, acoustic emission, and force during the cutting process, the wear degree of tools can be effectively predicted, thereby providing robust support for smart manufacturing.

Current research on tool condition recognition primarily focuses on data-driven methods, improvements in deep learning techniques, multi-source information fusion, and real-time or online monitoring objectives. Most studies adopt data-driven approaches, utilizing image and sensor data to identify and predict tool wear conditions. Deep learning technologies, especially models like support vector regression (SVR), Transformer, and ResNet, are widely employed to extract features of tool wear and perform predictions[3]. However, despite numerous valuable advancements in tool wear recognition, further optimization of algorithms, diversification of datasets, and enhancement of model adaptability under varying working conditions are still required to improve practical applicability.

In recent years, regarding deep learning model fusion, Ullah et al.[4] proposed a multi-scale feature fusion model, MSWDNet, combining dilated con-

*Corresponding author, E-mail address: KunZheng@njit.edu.cn.

How to cite this article: ZHENG Kun, LI Yonglin, GU Xinyan, et al. Pyramid pooling-based vision Transformer for tool condition recognition[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2025, 42(3): 322-336.

http://dx.doi.org/10.16356/j.1005-1120.2025.03.005

volution blocks and channel attention mechanisms. Trained on a dataset containing 3 351 images of wheat disease in Pakistani fields, the experimental results demonstrated that the model's accuracy significantly surpassed existing methods. Ablation studies confirmed the critical role of multi-scale feature fusion and attention mechanisms in improving disease detection precision. Ren et al.[5] designed a BiLSTM-BiGRU hybrid model, fusing multi-model prediction results using a covariance intersection algorithm. By estimating variance through overlapping data to optimize weight allocation, the model achieved improved prediction accuracy on the Beijing meteorological dataset while outputting prediction fluctuation ranges, offering an interpretable fusion framework for complex time-series data analysis. Shah et al.[6] innovatively fused physical features such as pitch, intensity, and spectral slope with log power spectrum features. Based on a convolutional neural network (CNN) model tested on the TIMIT dataset, recognition accuracy was markedly enhanced. Experiments showed that the statistical feature fusion strategy outperformed raw sequence inputs, and robustness was further improved using majority voting. Haider et al.[7] proposed a model integrating bidirectional long short-term memory (BiLSTM) and gated recurrent unit (GRU), achieving high detection accuracy and low false alarm rates on the CIC-IDS 2018 dataset. Compared to the CNN-LSTM baseline model, it reduced computational latency by 20% and memory consumption by 15%. Through hierarchical batch processing to balance data distribution, the model effectively identified diverse attacks like DDoS and brute-force, demonstrating the comprehensive performance advantages of hybrid architectures in real-time cloud security scenarios.

Inspired by these model fusion approaches, in the field of tool condition recognition, Yang et al.[8] proposed a tool wear condition recognition method combining wavelet packet transform and a 1D CNN. This method uses wavelet packet decomposition to denoise spindle vibration signals and selects the energy features of each frequency band after decomposition as inputs to the CNN, achieving accurate identification of tool wear states. Experimental results showed that the proposed model significantly outperformed traditional BP neural networks, energy spectrogram-AlexNet, and LSTM models in recognition accuracy. Wei et al.[9] addressed tool wear recognition under small-sample conditions by proposing an innovative Inception-BiLSTM model. This method extracted time-frequency features of signals using continuous wavelet transform (CWT), then employed an inception network for feature extraction, followed by global average pooling for dimensionality reduction, and finally utilized BiLSTM for state recognition. Results indicated higher accuracy under limited data conditions. Han et al.[10] introduced a tool wear recognition method based on an improved Hunter-Prey optimization (HPO) algorithm, variational mode decomposition (VMD), and a GRU. By optimizing VMD layers and penalty factors via HPO, the signal decomposition effect was enhanced. After feature extraction and kernel principal component analysis (KPCA) for dimensionality reduction, the GRU network achieved state recognition. Experiments confirmed that the proposed model exhibited superior accuracy, efficiency, and generality compared to traditional methods. Guo et al.[11] proposed a tool wear recognition method using a stacked sparse denoising autoencoder (SSDAE). By decomposing raw vibration signals into intrinsic mode functions (IMFs) and selecting optimal IMFs via Pearson correlation coefficients, the SSDAE adaptively extracted deep features for state recognition. Results demonstrated the method's effectiveness in handling non-stationary vibration signals with strong generalization and reliability.

Pyramid pooling has been widely applied to computer vision for feature extraction[12]. Originating prior to the rapid development of deep CNNs, pyramid pooling was frequently used in natural scene recognition. He et al.[13] integrated spatial pyramid pooling into deep CNNs for image classification and object detection. By introducing multi-scale pooling operations, they converted the final convolutional feature maps into fixed-size representations through pooling at different levels, enhancing the model's robustness and generalization. Zhao et

al.[14] adapted pyramid pooling for semantic segmentation. Unlike He et al., they upsampled the pooled fixed-size features to the original dimensions and used them for prediction, validating pyramid pooling's effectiveness in dense prediction tasks.

Inspired by these studies, this paper proposes a fusion of pyramid pooling and Transformer to improve model robustness and generalization. By leveraging an image classification-based approach, the accuracy of tool condition recognition is enhanced. The contributions of this paper are described as follows: (1) First, based on the fundamental principles of pyramid pooling and the complementary strengths of CNN and Transformer, the P2ViT model integrating pyramid pooling and Transformer is proposed. (2) Second, the improved P2ViT model incorporates a multi-level spatial pooling mechanism (pyramid pooling) into the vision Transformer framework. By constructing a hierarchical feature compression module, the model optimizes the self-attention computation topology while preserving the multi-scale representation capability of the backbone network. Additionally, depthwise (DW) operations are embedded within the feed-forward network (FFN) to enhance 2D local feature learning.

1 Data Description and Dataset

1.1 PHM2010 dataset description

Tool wear has long been a critical research topic in the mechanical field. Over decades of development, standardized research protocols have been established, particularly with the widespread adoption of the PHM2010 (Product Health Management Society 2010) dataset[15] by scholars globally. This study also employs the publicly available PHM2010 dataset as the primary data source based on the following considerations. (1) The dataset is extensively utilized in the industry due to its high reliability, diverse data types, and strong interpretability. (2) Its frequent use in numerous journal articles ensures that experimental results in this study are highly comparable, which helps highlight the innovative contributions of our proposed learning model and facilitates further research by scholars in related fields.

The dataset is collected using a Roders TechRFM760 high-speed CNC milling machine, equipped with a 6 mm solid carbide ball-end milling tool. The tool operates at a spindle speed of 10 400 r/min and a feed rate of 1 555 mm/min. The schematic diagram of the original data acquisition setup is illustrated in Fig.1.

1.2 Dataset

Tool condition recognition is a classification problem. As mentioned earlier, tool wear undergoes three stages: Initial wear, normal wear, and severe wear. Therefore, category labels need to be assigned to each data sample before training. In this study, the labels for initial wear, normal wear, and

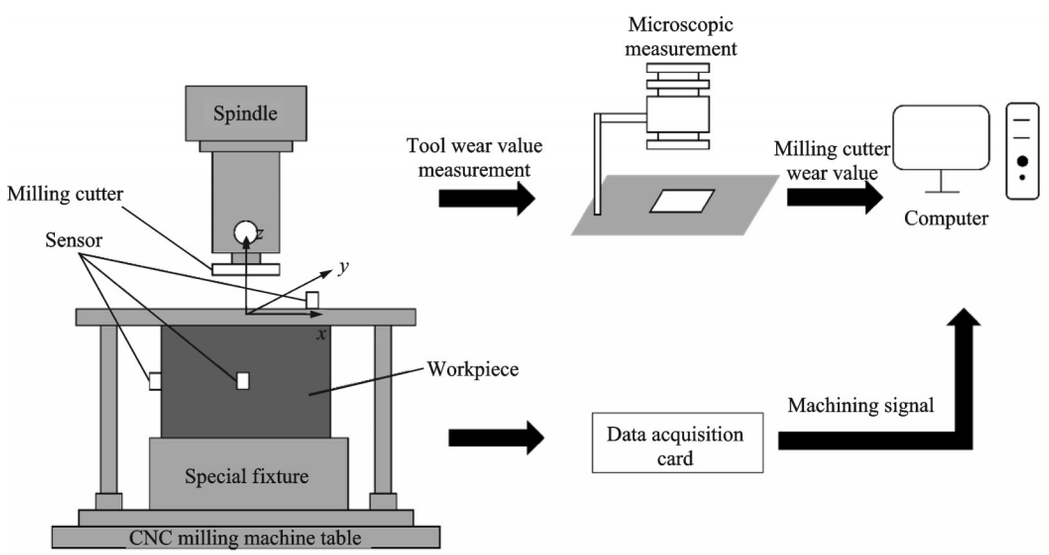


Fig.1 Schematic diagram of tool signal data acquisition

severe wear are set as "initial" "normal", and "severe", respectively. The dataset includes wear values for three tools, corresponding to datasets C1, C4, and C6. Each tool has three flute wear values: flute_1, flute_2, and flute_3. Wei et al.[16] employed the expectation-maximization (EM) algorithm[17] to determine the wear labels for each machining pass of the three tools, as shown in Table 1. The classification principle involves taking the average wear value of the three flutes for each tool as its overall wear value, and then using EM clustering to identify the machining pass intervals corresponding to each wear stage. The training-to-testing ratio is set at 7∶3.

Table 1 Wear state classification in PHM2010 dataset

Dataset	Initial wear	Normal wear	Severe wear
C1	1—47	48—146	147—315
C4	1—135	136—204	205—315
C6	1—81	82—188	189—315

The creation process of the tool condition image classification dataset is shown in Fig.2. The one-dimensional time-series signals are converted into two-dimensional time-frequency images using CWT. The time-frequency characteristics provide instantaneous changes of tool signals during the milling process, enabling the model to capture more detailed data features, which improves the recognition accuracy to some extent and supports non-offline tool condition monitoring. In this study, the CWT uses the cgau8 wavelet basis function with a scale range of 1—256. The sampling frequency is 50 kHz, and the signal length is 5 000 points. For C1, C4 and C6, a total of 237 918 image datasets are created and divided into a training set (214 128 images) and a test set (23 790 images) in a ratio of 9∶1.

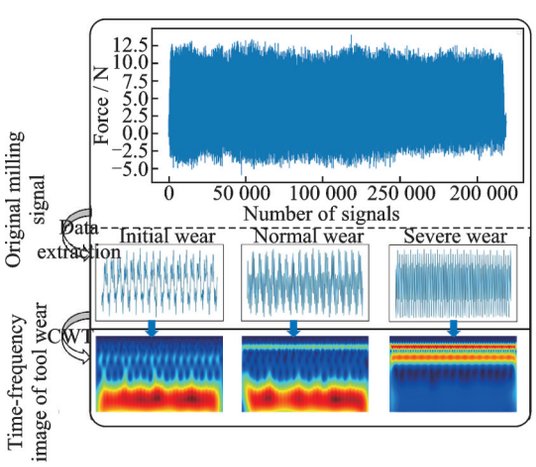


Fig.2 Flowchart of dataset creation process

2 Methodology

2.1 Pyramid pooling

The principle of pyramid pooling is illustrated in Fig.3. For feature inputs of arbitrary sizes, the process involves three key steps, with the operational procedure detailed in Fig.4. Assuming the input image has a channel depth of 256, the first stage performs global pooling to generate a 1×256 feature vector. In the second stage, the feature matrix is divided into 2×2 grid cells, and local pooling is applied to each sub-region, producing four 256-dimensional vectors. The final stage employs a 4×4 grid division to extract 16 local feature vectors. By concatenating the pooled results from these three spatial scales, a final feature of dimension 5 376 ((1+4+16)×256 = 21×256) is constructed. As evident from the pooling mechanism, the whole operation is

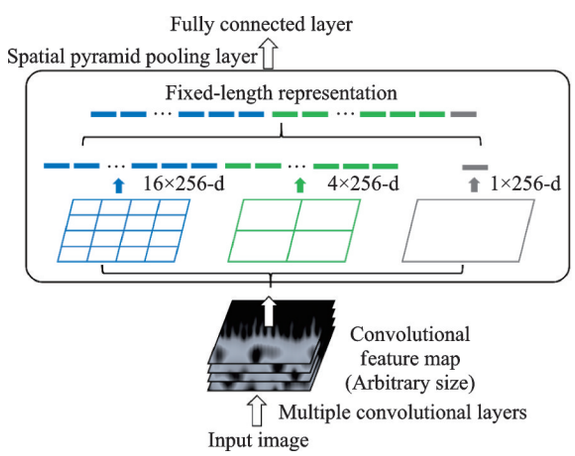


Fig.3 Pyramid pooling schematic diagram

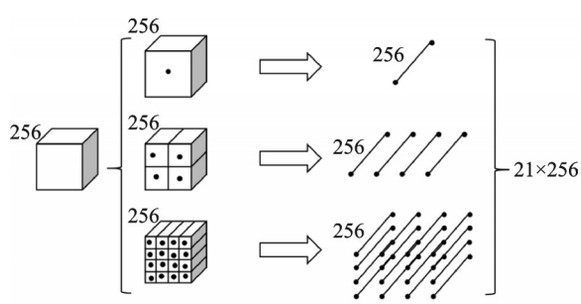


Fig.4 Schematic diagram of three-step pyramid pooling process

entirely independent of the input dimensions, and always yields a fixed-channel output (21 channels). This enables the theoretical capability to process input images of any arbitrary size.

Inspired by the application of pyramid pooling in CNN architectures, this paper proposes to integrate the concept of pyramid pooling into the vision Transformer (ViT) backbone network. By embedding pyramid pooling into the basic pooling attention module of the ViT architecture, we achieve both sequence length reduction and enhanced learning of robust contextual feature representations.

2.2 Vision Transformer

Vision Transformer originates from the sequence modeling Transformer architecture. Its core mechanism relies on multi-head self-attention (MHSA) to establish global interaction networks among input elements. This architecture first demonstrates the effectiveness of self-attention-based sequence modeling in natural language processing, particularly achieving dynamic capture of non-local dependencies in machine translation tasks. Given the need for modeling long-range feature relationships in computer vision tasks, researchers attempt to adapt this architecture to image processing. However, since standard Transformer operations are designed for one-dimensional sequential data, directly processing two-dimensional image matrices presents dimensional compatibility challenges. This necessitates feature reorganization to map spatial pixels into processable sequential representations. By employing CNNs to extract 2D representations, which are then flattened and fed into the Transformer, image classification tasks can be effectively addressed.

Theoretically, the Transformer has no limitation on memory length, with its greatest advantage being parallel processing capability. The self-attention mechanism is the core component of the Transformer, enabling the model to consider information from other positions in the sequence when processing the output at a given position. As illustrated in Fig.5, the self-attention mechanism operates as follows: The input vectors $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are first mapped to higher-dimensional representations a_1 and a_2

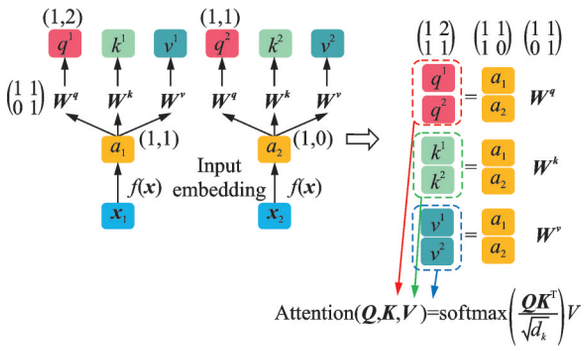


Fig.5 Schematic diagram of self-attention mechanism principles

through embedding. These embeddings are then transformed by shared weight matrices $\boldsymbol{W}^q$, $\boldsymbol{W}^k$, and $\boldsymbol{W}^v$ to generate corresponding query ($\boldsymbol{q}$), key ($\boldsymbol{k}$), and value ($\boldsymbol{v}$) vectors, where the same weight matrices are applied to all input embeddings. The computation of $\boldsymbol{q}$, $\boldsymbol{k}$, and v follows Eq.(1), where each query vector $\boldsymbol{q}$ is matched against all key vectors $\boldsymbol{k}$, while $\boldsymbol{v}$ represents the meaningful information extracted from the original embeddings. Leveraging the parallel processing capability of Transformer, the column vectors of embeddings can be concatenated into matrices and multiplied by $\boldsymbol{W}^q$, $\boldsymbol{W}^k$, and $\boldsymbol{W}^v$ to obtain q^1, q^2, k^1, k^2, v^1, v^2, which are then concatenated to form the complete query ($\boldsymbol{Q}$), key ($\boldsymbol{K}$), and value ($\boldsymbol{V}$) matrices. The matching process between $\boldsymbol{Q}$ and $\boldsymbol{K}$ is demonstrated in Fig.6 and calculated according to Eq.(2). The resulting attention scores α are normalized through softmax operation to produce the final attention weights $\hat{\alpha}$, which determine the relative importance of each value vector. The higher weights, the more attention is paid. The computation of the weighted value matrix $\boldsymbol{V}$ follows Eq.(3) and is illustrated in Fig.7. The MHSA mechanism extends this principle by computing multiple sets of $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ matrices in parallel, each capturing different aspects of the input information. Each attention head learns distinct subspace representations, and their outputs are concatenated before undergoing a final linear transformation to produce the combined result. Additionally, the Transformer incorporates positional encoding to prevent potential errors that might arise from positional variations among input elements.

$$\begin{cases} q^i = a^i \boldsymbol{W}^q \\ k^i = a^i \boldsymbol{W}^k \\ v^i = a^i \boldsymbol{W}^v \end{cases} \tag{1}$$

$$\begin{cases} \alpha_{1,i} = q^1 \cdot k^i / \sqrt{d} \\ \alpha_{2,i} = q^2 \cdot k^i / \sqrt{d} \\ \cdots \end{cases} \tag{2}$$

$$\begin{cases} b^1 = \sum_i \hat{\alpha}_{1,i} \times v^i \\ b^2 = \sum_i \hat{\alpha}_{2,i} \times v^i \\ \cdots \end{cases} \tag{3}$$

$$\text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{4}$$

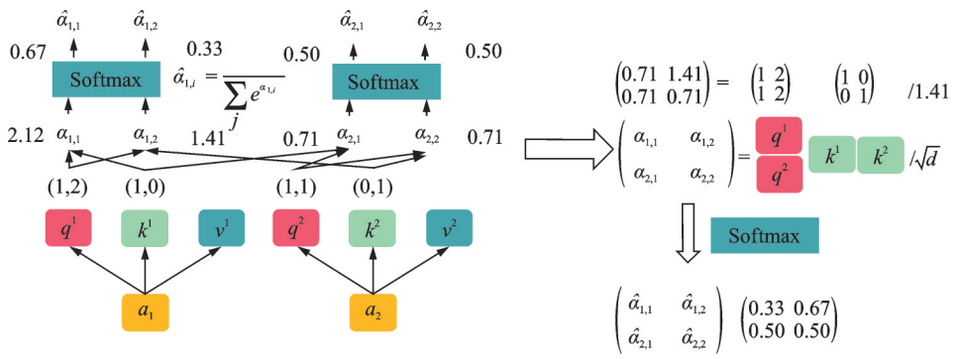


Fig.6 Schematic diagram of query-key matching process

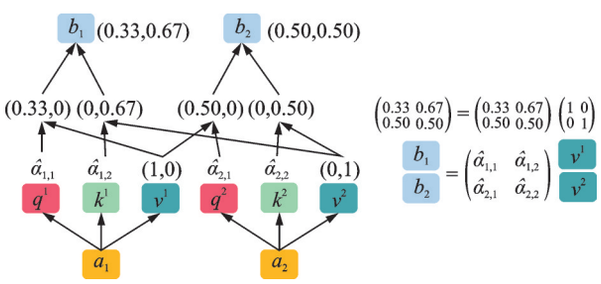


Fig.7 Schematic diagram of value computation process

The architecture of the ViT model is illustrated in Fig.8. An input image is divided into multiple patches, which are fed into the linear projection of flattened patches layer to obtain corresponding vectors, referred to as tokens. A new classification token is prepended to these tokens, with positional encodings assigned as 0, 1, 2, etc. These tokens are then processed through the Transformer encoder layers and the multi-layer perceptron head (MLP Head) module to generate the final classification output[18]. The MLP Head consists of a linear (fully connected) layer, followed by a tanh activation function and another linear layer. The linear projection of flattened patches layer, designed for standard Transformer modules, requires token sequences as input (i. e., a 2D matrix of dimensions [num_token, token_dim]). In practical implementation, this can be achieved using a convolutional layer. For instance, ViT-B/16 employs convolutional kernels of 16×16 with a stride of 16 and 768 kernels. Prior to being fed into the Transformer encoder, both the classification token and positional encodings, which are trainable parameters, must be incorporated.

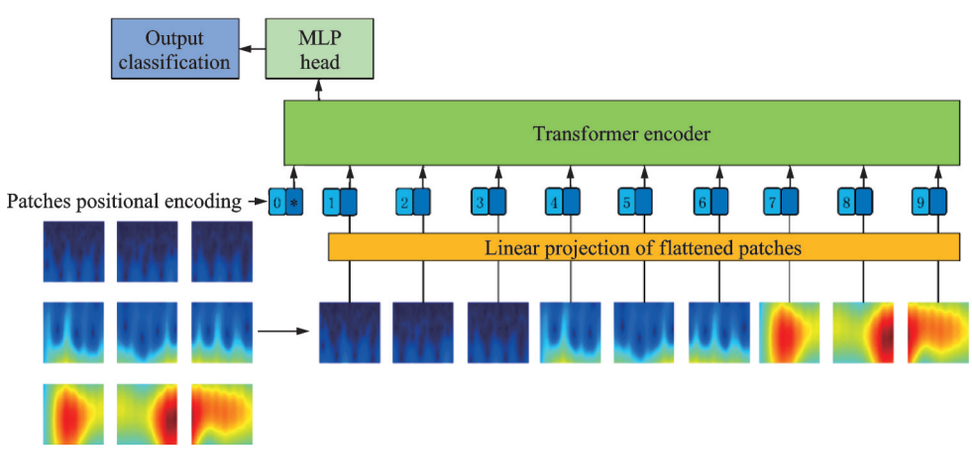


Fig.8 Schematic architecture diagram of ViT

The architecture of the Transformer encoder layer is shown in Fig.9, where the Transformer encoder block is stacked L times sequentially. The processing flow is as follows: The input first passes through a normalization layer, then undergoes multi-head attention computation, followed by a dropout layer. The output from dropout is then added to the original input (residual connection). Before being processed by the MLP block (shown in Fig.10) and dropout operation, this sum subsequently passes through another normalization layer, with the final result again obtained through residual addition.

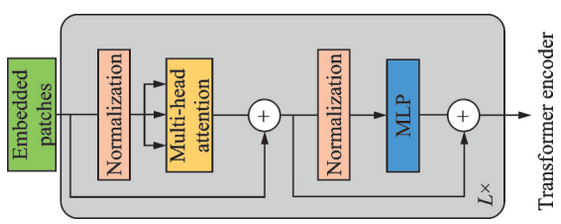


Fig.9 Schematic framework diagram of Transformer encoder

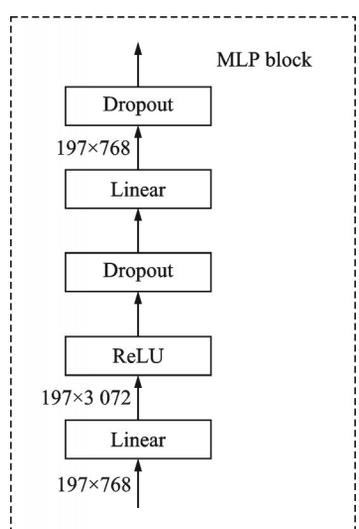


Fig.10 Schematic diagram of MLP block architecture

2.3 P2ViT-Net

2.3.1 Network structure design

Pyramid pooling has been extensively utilized in various scene understanding tasks and has been cooperatively employed with CNNs[19-20]. However, existing studies predominantly apply pyramid pooling to CNN backbone networks for extracting global and contextual information, often focusing on specific tasks[21]. This paper introduces a cross-modal architecture optimization method based on hierarchical feature aggregation, aiming to integrate multi-level spatial pooling mechanisms (pyramid pooling) into the vision Transformer framework. By developing hierarchical feature compression modules, the proposed approach optimizes the self-attention computation topology while preserving the backbone network's multi-scale representational capacity. In practical implementation, spatial resolution reduction is employed to effectively control the computational complexity of MHSA, while cross-receptive-field contextual correlation networks are established to dynamically harmonize local details with global semantics. The enhanced model, designated as P2ViT-Net (Pyramid pooling-based vision Transformer network), is structurally depicted in Fig.11. The input undergoes pooling-based MHSA processing, after which the output is combined with residual connections and normalized via LayerNorm. Following the conventional Transformer block design, a FFN is sequentially appended for feature projection, succeeded by another residual connection and LayerNorm operation. The mathematical formulation of this workflow is expressed as follows

$$\begin{cases} \boldsymbol{X}_{\text{att}} = \text{LayerNorm}(\boldsymbol{X} + \text{P-MHSA}(\boldsymbol{X})) \\ \boldsymbol{X}_{\text{out}} = \text{LayerNorm}(\boldsymbol{X}_{\text{att}} + \text{FFN}(\boldsymbol{X}_{\text{att}})) \end{cases} \tag{5}$$

where $\boldsymbol{X}$, $\boldsymbol{X}_{\text{att}}$ and $\boldsymbol{X}_{\text{out}}$ represent the input, the MHSA output, and the Transformer block output, respectively; P-MHSA (Pooling-based MHSA) denotes the pooling-based multi-head self-attention mechanism.

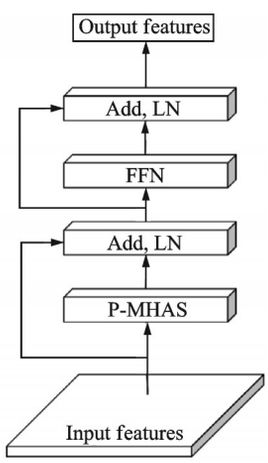


Fig.11 Simplified architecture diagram of P2ViT model

2.3.2 Pooling-based multi-head attention mechanism

The design of P-MHSA is illustrated in Fig.12, where the input $\boldsymbol{X}$ is reshaped into a 2D spatial format and processed through multiple average pooling layers with different pooling ratios to generate pyramid feature maps. The calculation formula is as follows

$$\begin{cases} \boldsymbol{P}_1 = \mathrm{AvgPool}_1(\boldsymbol{X}) \\ \boldsymbol{P}_2 = \mathrm{AvgPool}_2(\boldsymbol{X}) \\ \quad\vdots \\ \boldsymbol{P}_n = \mathrm{AvgPool}_n(\boldsymbol{X}) \end{cases} \tag{6}$$

where $\boldsymbol{P}_1, \boldsymbol{P}_2, \cdots, \boldsymbol{P}_n$ represent the generated pyramid feature maps; n denotes the number of pooling layers. This paper selects $n = 4$, which will be detailed in Section 2.3.3. Furthermore, these pyramid feature maps need to be fed into depthwise convolutional layers for relative position encoding, as expressed in the following formula

$$\boldsymbol{P}_i^{\mathrm{enc}} = \mathrm{DWConv}(\boldsymbol{P}_i) + \boldsymbol{P}_i \quad i = 1, 2, \cdots, n \tag{7}$$

where $\mathrm{DWConv}(\cdot)$ represents a depthwise convolution operation with a 3×3 kernel size and $\boldsymbol{P}_i^{\mathrm{enc}}$ the processed $\boldsymbol{P}_i$ after incorporating relative position encoding. These enhanced pyramid feature maps are subsequently flattened and concatenated, as formulated below

$$\boldsymbol{P} = \mathrm{LayerNorm}(\mathrm{Concat}(\boldsymbol{P}_1^{\mathrm{enc}}, \boldsymbol{P}_2^{\mathrm{enc}}, \cdots, \boldsymbol{P}_n^{\mathrm{enc}})) \tag{8}$$

In the MHSA mechanism, the query, the key, and the value tensors are denoted as $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$, respectively. The conventional computation formula is expressed as follows

$$(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = (\boldsymbol{X}\boldsymbol{W}^q, \boldsymbol{X}\boldsymbol{W}^k, \boldsymbol{X}\boldsymbol{W}^v) \tag{9}$$

When using the approach described in Eq.(8), if the pooling ratio is sufficiently large, the resulting $\boldsymbol{P}$ can form a shorter sequence than the original input $\boldsymbol{X}$. Furthermore, $\boldsymbol{P}$ inherently captures both preceding and succeeding feature information from $\boldsymbol{X}$. This enables $\boldsymbol{P}$ to effectively substitute for $\boldsymbol{X}$ in MHSA computations, leading to the modified formulation, as shown below

$$(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = (\boldsymbol{X}\boldsymbol{W}^q, \boldsymbol{P}\boldsymbol{W}^k, \boldsymbol{P}\boldsymbol{W}^v) \tag{10}$$

where $\boldsymbol{W}^q$, $\boldsymbol{W}^k$, and $\boldsymbol{W}^v$ represent the weight matrices for the linear transformations that generate the query ($\boldsymbol{Q}$), the key ($\boldsymbol{K}$), and the value ($\boldsymbol{V}$) tensors, respectively. These $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ tensors are then fed into the attention module to compute the attention weights $\boldsymbol{A}$, as expressed by the following formula

$$\boldsymbol{A} = \mathrm{Softmax}\left(\frac{\boldsymbol{Q}\times\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_K}}\right)\times\boldsymbol{V} \tag{11}$$

where d_K denotes the channel dimension of $\boldsymbol{K}$ and $\sqrt{d_K}$ the approximate normalization operation.

2.3.3 Deep embedded feature enhancement architecture

This paper proposes a deeply embedded feature enhancement architecture that breaks through the conventional paradigm of using pyramid pooling as an independent module. Unlike traditional approaches that treat hierarchical feature aggregation mechanisms as back-end enhancement modules, our solution deeply integrates them into the core processing pipeline of the ViT architecture. This enables the network to simultaneously perform multi-scale context modeling during the fundamental feature extraction stage, thereby enhancing its learning capacity. Furthermore, the keys ($\boldsymbol{K}$) and the values ($\boldsymbol{V}$) incorporate highly abstracted multi-scale information, endowing the P-MHSA with superior capability in modeling global contextual dependencies. This proves particularly effective for scene understanding, making P-MHSA theoretically more efficient than conventional MHSA. Regarding computational complexity, as shown in Eq.(6), multiple pooling operations are employed to generate pyra-

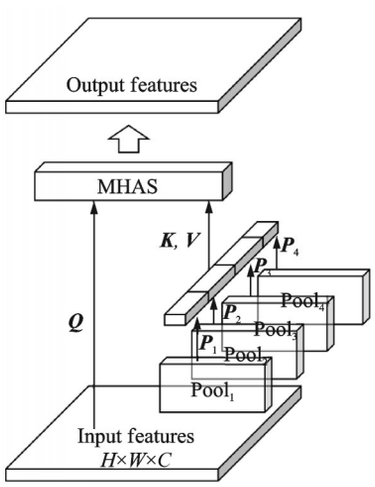


Fig.12 Schematic diagram of P-MHSA design

mid feature maps. The computational overhead of these pyramid pooling operations is negligible, being merely $O(NC)$, where N and C represent the sequence length and the feature dimension, respectively. The total computational complexity for computing the sub-attention is

$$O(\text{P-MHSA})=(N+2M)C^2+2NMC \quad (12)$$

where M represents the concatenated sequence length of all pooled features, with default pooling ratios set to [12, 16, 20, 24].

FFN serves as a crucial component in Transformers for feature enhancement. Conventional Transformers typically employ MLPs as their FFN and rely entirely on attention mechanisms to capture dependencies between pixels. Although this architecture demonstrates relatively high efficiency, it shows limited effectiveness in learning 2D local features, which plays a vital role in scene understanding. To address this limitation, inspired by existing CNNs, this paper integrates depthwise convolution into the FFN. This enhancement enables the improved model to simultaneously inherit the Transformer's capability for long-range dependency modeling while incorporating the advantages of CNNs in capturing 2D local patterns. Specifically, we adopt the inverted residual block (IRB) proposed in MobileNetV2[22] as the FFN structure.

To adapt IRB for ViT, the input sequence $\boldsymbol{X}_{\text{att}}$ is first reshaped into a 2D feature map $\boldsymbol{X}_{\text{att}}^{\text{I}}$, as follows

$$\boldsymbol{X}_{\text{att}}^{\text{I}}=\text{Seq2 Image}(\boldsymbol{X}_{\text{att}}) \quad (13)$$

where Seq2Image(·) denotes the function that reshapes a 1D sequence into a 2D feature map. For the input $\boldsymbol{X}_{\text{att}}^{\text{I}}$, the IRB can be directly used as follows

$$\begin{cases}\boldsymbol{X}_{\text{IRB}}^{1}=\text{Act}(\boldsymbol{X}_{\text{att}}^{\text{I}}\boldsymbol{W}_{\text{IRB}}^{1})\\ \boldsymbol{X}_{\text{IRB}}^{\text{out}}=\text{Act}(\text{DWConv}(\boldsymbol{X}_{\text{IRB}}^{1}))\boldsymbol{W}_{\text{IRB}}^{2}\end{cases} \quad (14)$$

where $\boldsymbol{W}_{\text{IRB}}^{1}$ and $\boldsymbol{W}_{\text{IRB}}^{2}$ represent the weight matrices of the 1×1 convolutional layers; Act denotes the nonlinear activation function and $\boldsymbol{W}_{\text{IRB}}^{\text{out}}$ the output feature map from the IRB. Since $\boldsymbol{W}_{\text{IRB}}^{\text{out}}$ is a 2D spatial feature map, it ultimately needs to be transformed back into a 1D sequence representation, as expressed as

$$\boldsymbol{X}_{\text{IRB}}^{\text{S}}=\text{Image 2Seq}(\boldsymbol{X}_{\text{IRB}}^{\text{out}}) \quad (15)$$

where Image2Seq(·) denotes the operation that reshapes a 2D feature map back into a 1D sequence. The final output $\boldsymbol{X}_{\text{IRB}}^{\text{S}}$ corresponds to the FFN output and maintains identical dimensions with $\boldsymbol{X}_{\text{att}}$.

In summary, the improved P2ViT model proposed in this paper is illustrated in Fig.13. The input consists of three-channel RGB color images. P2ViT first partitions each image into $(H/4)\times(H/4)$ patches, with each patch flattened into 48 elements ($4\times4\times3$). These flattened patches are then processed by a patch embedding module composed of a linear projection layer, followed by the addition of learnable positional encodings. The patch embedding module expands the feature dimension from 48 to C_1. Subsequently, the features pass through the proposed pyramid-pooling ViT blocks. The entire network is divided into four stages, with feature dimensions denoted as C_i ($i=1,2,3,4$). Between every two stages, groups of 2×2 patches are concatenated and linearly projected, transforming the feature dimension from $4\times C_i$ to C_{i+1} ($i=1, 2, 3$). Through this approach, the four stages progressive-

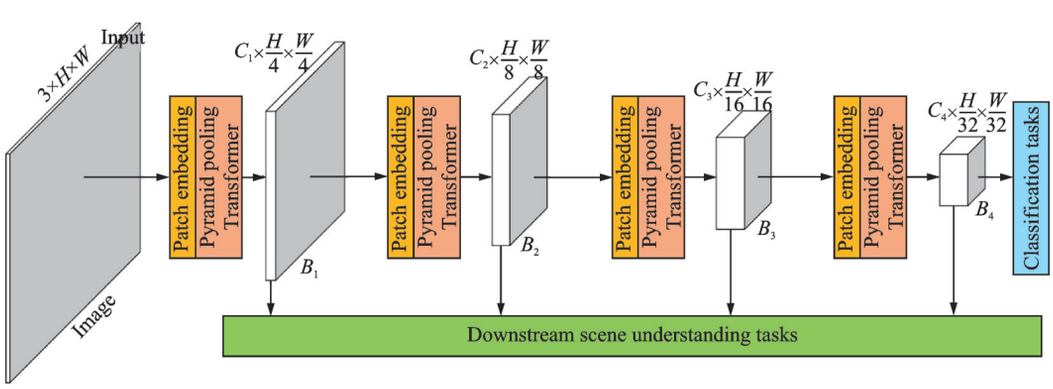


Fig.13 Schematic diagram of the proposed P2ViT model architecture

ly achieve scales of $(H/4)\times(H/4)$, $(H/8)\times(H/8)$, $(H/16)\times(H/16)$ and $(H/32)\times(H/32)$, respectively. From these stages, four distinct feature representations (B_1, B_2, B_3, B_4) are generated. While only B_4 is utilized for final image classification prediction, all pyramid features remain available for downstream scene understanding tasks.

3 Results and Discussion

3.1 Experimental setup and evaluation metrics

The training parameter configurations are detailed in Table 2, with a batch size of 64, learning rate of 0.000 2, 100 training epochs, CrossEntropyLoss function, Adam optimizer, and a training-to-test set ratio of 9∶1. The accuracy of the best-performing model is computed and saved after each epoch. For model evaluation, precision (P), recall (R), specificity (S), and F_1-score are employed as metrics, all derived from the confusion matrix values shown in Table 3. The computational formulas for precision, recall, specificity, and F_1-score are as follows

$$P=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{16}$$

$$R=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{17}$$

$$S=\frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FN}} \tag{18}$$

$$F_1=\frac{2\times P\times R}{P+R} \tag{19}$$

Table 2 Training parameter configuration

Batch size	Learning rate	Epoch	Loss function	Optimizer	Split
64	0.000 2	100	CrossEntropyLoss	Adam	0.1

Table 3 Fundamentals of the classification confusion matrix

True label	Prediction result	
	Positive	Negative
Positive	TP (True positive)	FN (False negative)
Negative	FP (False positive)	TN (True negative)

3.2 Analysis of experimental results

The confusion matrix of the proposed P2ViT model, obtained through training, is shown in Fig.14. The diagonal values in the matrix represent the number of samples where the model's predictions exactly match the true labels, with darker shades indicating higher counts. The diagonal values in the confusion matrix are 6 339 (initial wear), 6 205 (normal wear), and 9 923 (severe wear), demonstrating that the model achieves the highest recognition accuracy for severe wear states (darkest color), followed by initial wear and normal wear, all exhibiting peak identification rates. As detailed in Table 4, all four evaluation metrics exceed 90%. The F_1-scores surpass 90%, indicating strong comprehensive performance of the enhanced P2ViT in tool condition monitoring. Similarly, precision, recall, and specificity metrics also exceed 90%, with the overall accuracy reaching 94.4%. These results confirm that the improved model delivers stable overall performance and notable effectiveness across all evaluation criteria. It is noteworthy that among the three states of tool wear, the normal wear stage exhibits relatively lower performance metrics (accuracy, precision, recall, and F_1-score) due to discriminative ambiguity in transitional state identifica-

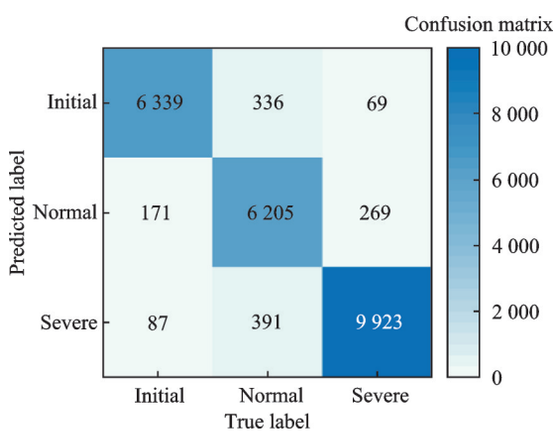


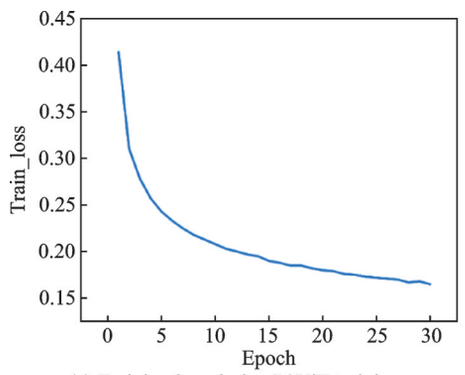
Fig.14 Confusion matrix of the enhanced model's training results

Table 4 Experimental values of performance evaluation metrics

Label	P	R	S	F_1	Accuracy	Sample
Initial	0.940	0.961	0.976	0.950	0.944	Train: 203 710 Val: 22 618
Normal	0.934	0.895	0.974	0.914		
Severe	0.954	0.967	0.965	0.960		
Average	0.943	0.941	0.972	0.941		

tion. This occurs because the sample characteristics during normal wear represent an intermediate phase between initial wear and severe wear, creating overlapping feature distributions that challenge precise classification.

Fig.15 visualizes the convergence process of the training loss (Train_loss) and validation accuracy (Val_acc) during P2ViT model training. Since the training process reaches convergence after 30 epochs, Fig.15 displays data from the first 30 epochs to clearly illustrate parameter variations. The train_loss curve demonstrates a gradual decline toward a minimal steady-state value, quantifying the discrepancy between predictions and ground-truth labels while driving parameter optimization. This metric reflects the model's average loss on the training set and indicates its fitting performance at each epoch. Notably, the enhanced model exhibits stable convergence behavior in tool condition recognition tasks. Val_acc represents classification accuracy on the validation set, which progressively increases with training epochs and ultimately converges to a high value (94.4%). This trend confirms the model's successful learning of discriminative features and continuous improvement in recognition accuracy.



(a) Training loss during P2ViT training process

(b) Validation accuracy during P2ViT training process

Fig.15 Training loss and validation accuracy during P2ViT training process (Epoch=30)

This section conducts performance comparisons between various CNNs and the proposed enhanced model to objectively demonstrate the improvement in recognition accuracy. The benchmarked models include AlexNet[23], VGG[24], GoogLeNet[25], ResNet[26], MobileNet[27], ShuffleNet[28], ViT, and the proposed P2ViT model. Fig.16 presents the training confusion matrices for these different models. In each matrix, the darkest coloration appears along the main diagonal corresponding to respective labels, indicating successful identification of tool wear states by each CNN. In each confusion matrix, the deepest color intensity

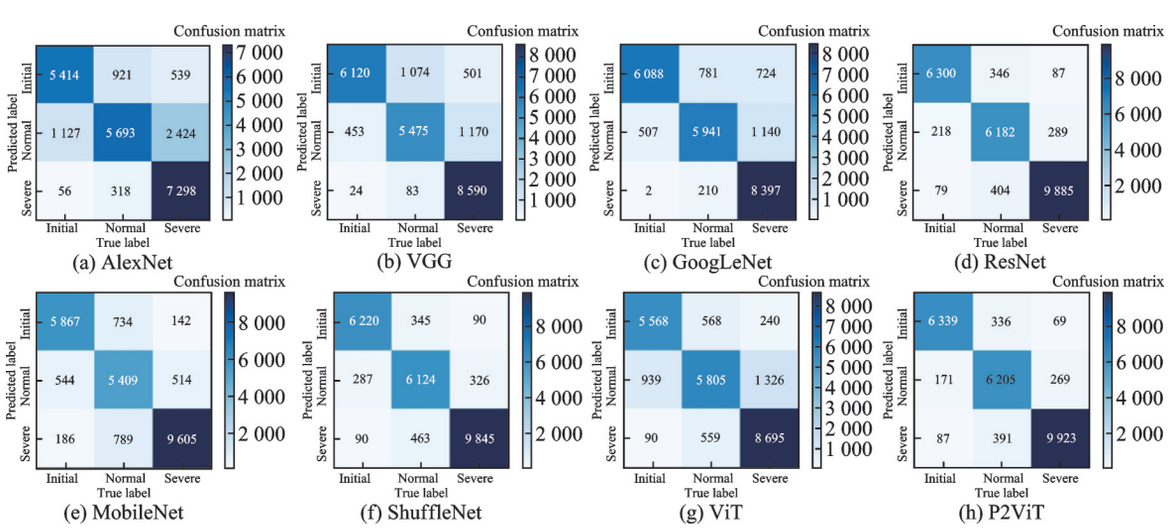
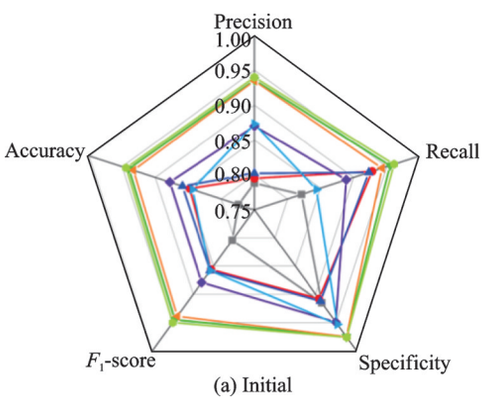


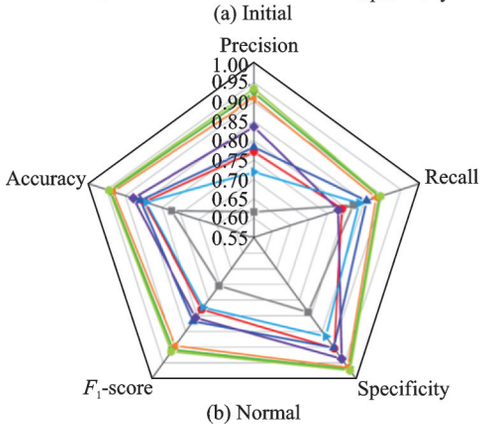
Fig.16 Confusion matrices of training results for different deep CNN models

along the main diagonal positions corresponds to their respective labels, demonstrating that all CNNs successfully identify tool wear states. The color gradient reflects recognition confidence levels, with darker hues indicating higher classification certainty. Fig.17 presents a radar chart of performance metrics derived from these confusion matrices, which visually compares the operational parameters across three wear states (initial, normal, severe) under different model architectures, along with their ensemble averages.

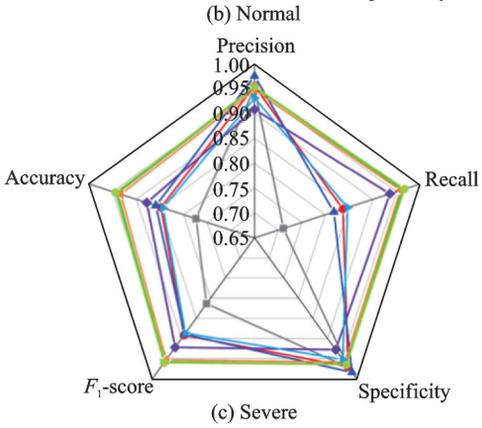
The specific values of evaluation metrics are obtained through the confusion matrix, with the accuracy rates of eight distinct models illustrated in Fig.18. A dashed line serves as the demarcation,



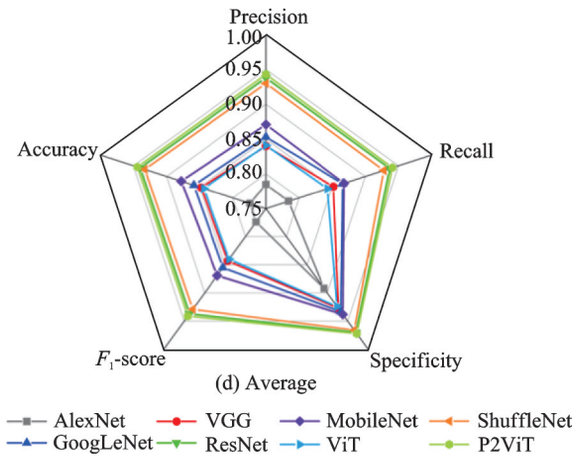
(a) Initial



(b) Normal



(c) Severe



(d) Average

Fig.17 Comparative radar chart of performance metrics across different deep CNN models

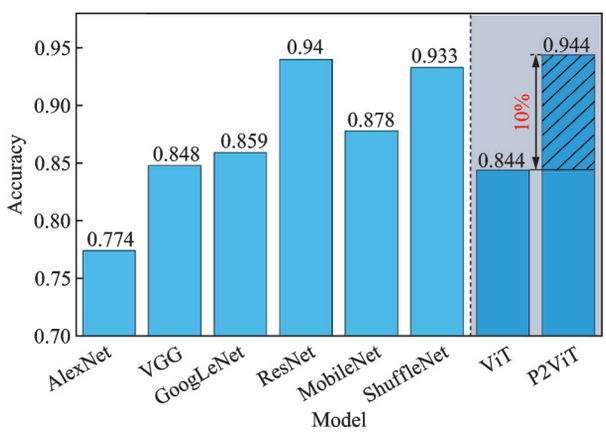
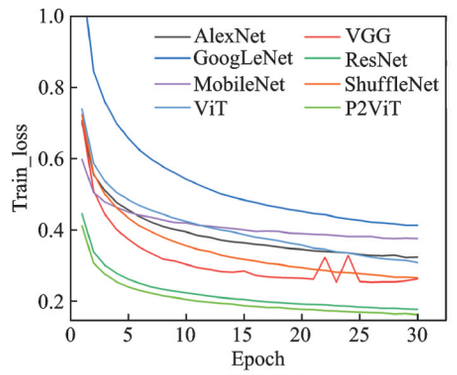


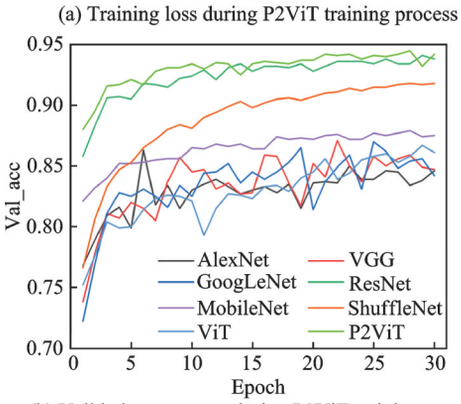
Fig.18 Comparative classification accuracies of tool wear state recognition across different deep CNNs

where the left side represents the recognition performance of deep CNNs, while the right side displays the improved image classification models based on Transformer architecture. Regarding the Transformer-enhanced models, the proposed model demonstrates a significant accuracy improvement of 10%. Although the enhancement appears relatively modest (0.4%) when compared with the highest-accuracy model among deep convolutional neural networks, it ultimately achieves the superior overall accuracy among all benchmarked models. Consequently, the experimental results substantiate that the proposed P2ViT model exhibits robust recognition capability in tool wear image identification.

Fig.19 presents the variation processes of train_loss and val_acc during the training phase of different models for tool condition recognition. While all methodologies exhibit similar trend patterns, the



(a) Training loss during P2ViT training process



(b) Validation accuracy during P2ViT training process

Fig.19 Training loss and validation accuracy across different deep CNNs (Epoch=30)

proposed P2ViT model outperforms others in both metrics, manifesting the lowest training loss value and the highest validation accuracy among various deep CNN architectures.

4 Conclusions

This paper proposes an improved P2ViT-Net recognition model to address the limitations of traditional feature extraction methods in tool condition monitoring. The enhanced P2ViT-Net achieves 94.4% recognition accuracy on the PHM2010 dataset, representing a 10% improvement over conventional ViT. First, one-dimensional time-series signals are converted into two-dimensional time-frequency images through continuous wavelet transform to enhance feature representation capability. Second, a pyramid pooling module is introduced into the ViT architecture to improve the model's ability to capture tool wear details through multi-scale feature fusion. Meanwhile, DW optimization is embedded in the FFN layers to enhance local feature interaction and improve 2D local feature learning. Experimental results demonstrate significant improvements compared with traditional CNN models (AlexNet, VGG, GoogLeNet, ResNet, MobileNet, ShuffleNet) and ViT, validating the effectiveness of the global attention mechanism and multi-scale feature fusion. This confirms the relative advantages of the improved model in tool condition recognition. However, the current methodology exhibits insufficient sensitivity in detecting geometric anomalies of cutting tools (e.g., edge chipping). Future work will focus on employing advanced defect detection methodologies to conduct in-depth investigations of tool wear patterns under such abrupt geometric variations.

References

[1] SU Yue, XU Kaifu, JIN Lu, et al. An attention mechanism-based support stiffness prediction for rocket turbo pumps[J]. Journal of Nanjing University of Aeronautics & Astronautics, 2024, 56(4): 639-649. (in Chinese)

[2] GUO L, ZHENG L R, FENG L. Intelligent prediction of surface roughness of PSZ ceramic grinding based on correlation analysis and CNN-BiLSTM neural network[J]. Scientific Insights and Discoveries Review, 2024, 4: 313-322.

[3] WEI Yan, WANG Fangli, CHEN Jichang, et al. Prediction of fatigue crack propagation life of 2195 aluminum-lithium alloy FSW joints based on SVR[J]. Journal of Nanjing University of Aeronautics & Astronautics, 2024, 56(6): 1134-1142. (in Chinese)

[4] ULLAH N, AHMAD B, KHAN A, et al. Attention-guided wheat disease recognition network through multi-scale feature optimization[J]. IECE Transactions on Sensing, Communication, and Control, 2025, 2(1): 11-24.

[5] REN H C, WANG Y Q, MA H J. Deep prediction network based on covariance intersection fusion for sensor data[J]. IECE Transactions on Intelligent Systematics, 2024, 1(1): 10-18.

[6] SHAH Z, JANG G, FAROOQ A. Feature fusion for performance enhancement of text independent speaker identification[J]. IECE Transactions on Intelligent Systematics, 2024, 2(1): 27-37.

[7] HAIDER Z A, ZEB A, RAHMAN T, et al. Optimizing cloud security with a hybrid BiLSTM-BiGRU model for efficient intrusion detection[J]. IECE Transactions on Sensing, Communication, and Control,

2025, 2(2): 106-121.

[8] YANG B, FAN Z G, WANG J G, et al. Tool wear state recognition combining wavelet packet and 1D CNN[J]. Machinery Design & Manufacture, 2024: 228-232, 237.

[9] WEI Yonghe, WANG Geng, WU Jingyuan. Small-sample tool wear state recognition based on Inception-BiLSTM[J]. Modular Machine Tool & Automatic Manufacturing Technique, 2024(5): 147-151.(in Chinese)

[10] HAN Ning, LI Guofu, REN Lu. Application of improved HPO-optimized VMD-GRU method in tool wear state recognition[J/OL]. Mechanical Science and Technology for Aerospace Engineering, 1-9 [2023-11-23]. https://doi.org/10.13433/j.cnki.1003-8728.20230357.

[11] GUO Runlan, XUE Kai, DENG Wenqiang, et al. Optimization of process parameters for multi-feature structural size fusing deposition based on random walk sparrow search algorithm[J]. Journal of Lanzhou University of Technology, 2024, 50(1): 41-47. (in Chinese)

[12] LAZEBNIK S, SCHMID C, PONCE J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories[C]//Proceedings of 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). New York, NY, USA: IEEE, 2006: 2169-2178.

[13] HE K M, ZHANG X Y, REN S Q, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[M]//Computer Vision — ECCV 2014. Cham: Springer International Publishing, 2014: 346-361.

[14] ZHAO H S, SHI J P, QI X J, et al. Pyramid scene parsing network[C]//Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, USA: IEEE, 2017: 6230-6239.

[15] PHMSociety. 2010 PHM society conference data challenge[EB/OL]. [2010-02-13]. https://www.phmsociety.org/competition/phm/.

[16] WEI P N, LI R Y, LIU X L, et al. Research on tool wear state identification method driven by multi-source information fusion and multi-dimension attention mechanism[J]. Robotics and Computer-Integrated Manufacturing, 2024, 88: 102741.

[17] DEMPSTER A P, LAIRD N M, RUBIN D B. Maximum likelihood from incomplete data via the EM algorithm[J]. Journal of the Royal Statistical Society Series B: Statistical Methodology, 1977, 39(1): 1-22.

[18] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An image is worth 16×16 words: Transformers for image recognition at scale[J]. ArXiv e-Prints, 2020.

[19] WU Y H, LIU Y, ZHAN X, et al. P2T: Pyramid pooling Transformer for scene understanding[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(11): 12760-12771.

[20] HE J J, DENG Z Y, ZHOU L, et al. Adaptive pyramid context network for semantic segmentation[C]//Proceedings of 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA: IEEE, 2019: 7511-7520.

[21] CHEN L C, PAPANDREOU G, KOKKINOS I, et al. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(4): 834-848.

[22] SANDLER M, HOWARD A, ZHU M L, et al. MobileNetV2: Inverted residuals and linear bottlenecks[C]//Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA: IEEE, 2018: 4510-4520.

[23] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60(6): 84-90.

[24] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[EB/OL]. [2014-09-04]. http://arxiv.org/licenses/nonexclusive-distrib/1.0/.

[25] SZEGEDY C, LIU W, JIA Y Q, et al. Going deeper with convolutions[C]//Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, MA, USA: IEEE, 2015: 1-9.

[26] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition[C]//Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA: IEEE, 2016: 770-778.

[27] HOWARD A G, ZHU M L, CHEN B, et al. MobileNets: Efficient convolutional neural networks for mobile vision applications[EB/OL]. [2017-04-04]. https://arxiv.org/abs/1704.04861v1.

[28] ZHANG X, ZHOU X, LIN M, et al. ShuffleNet: An extremely efficient convolutional neural network for mobile devices[C]//Proceedings of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. [S.l.]: IEEE, 2018.

Acknowledgements This work was supported by China Postdoctoral Science Foundation (No. 2024M754122), the Postdoctoral Fellowship Program of CPSF (No. GZB20240972), the Jiangsu Funding Program for Excellent Postdoctoral Talent (No.2024ZB194), Natural Science Foundation of Jiangsu Province (No.BK20241389), Basic Science Research Fund of China (No.JCKY2023203C026), and 2024 Jiangsu Province Talent Programme Qinglan Project.

Author

The first/corresponding author Dr. **ZHENG Kun** received the B.S. degree in aircraft power engineering and the Ph.D. degree in mechanical electronic engineering from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2006 and 2016, respectively. He joined in School of Traffic Engineering, Nanjing Institute of Technology in September 2016. His research is focused on smart manufacturing and industrial digital transformation. He has long been dedicated to advancing intelligent improvement and digital transformation in enterprises, specializing in goal-oriented strategic planning for intelligent and digital initiatives, optimization of smart manufacturing system architectures, production system design, process enhancement, and research and development of digital enabling key technologies.

Author contributions Dr. **ZHENG Kun** and Mr. **LI Yonglin** designed the study, complied the models, conducted the analysis, interpreted the results, and wrote the manuscript. Dr. **GU Xinyan** contributed to data and model components for the deep learning model. Mr. **DING Zhiying** contributed to the discussion and background of the study. Dr. **ZHU Haihua** provided project funding and management, and reviewed the full manuscript. All authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing interests.

(Production Editor: XU Chengting)

基于金字塔池化的视觉Transformer在刀具状态识别上的应用

郑 堃[1], 李永林[1], 顾新艳[1], 丁志颖[1], 朱海华[2]

(1.南京工程学院交通工程学院,南京 211167,中国; 2.南京航空航天大学机电学院,南京 210016,中国)

摘要:基于数据驱动的方法,围绕刀具状态识别展开深入研究,旨在提升数控加工过程的智能化水平,提高刀具使用效率。传统的刀具监测方法依赖于经验或有限的数学模型,难以适应复杂、多变的加工环境。为此,引入深度学习技术,实现刀具实时状态识别。针对刀具状态识别准确率不足的问题,提出了基于金字塔池化的视觉Transformer网络(Pyramid pooling-based vision Transformer network,P2ViT-Net)的刀具状态识别方法。使用图像作为输入,缓解信号特征维度低的问题。改进了基于视觉Transformer(Vision Transformer, ViT)模型用于图像分类的P2ViT模型,并将其应用在刀具状态识别上。实验结果表明,改进的P2ViT模型刀具状态识别准确率达94.4%,较传统ViT准确率提高10%,且均高于对比的卷积神经网络模型。

关键词:刀具状态识别;Transformer;金字塔池化;深度卷积神经网络