IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 2 July 2024, accepted 19 July 2024, date of publication 25 July 2024, date of current version 5 August 2024.

*Digital Object Identifier 10.1109/ACCESS.2024.3433374*

RESEARCH ARTICLE

# Research on Multi-Scale CNN and Transformer-Based Multi-Level Multi-Classification Method for Images

**QUANDENG GOU[1] AND YUHENG REN[2,3]**
[1]Informatization Construction and Service Center, Neijiang Normal University, Neijiang 641000, China
[2]Xiamen Kunlu IoT Information Technology Company Ltd., Xiamen, Fujian 361021, China
[3]School of Business Economics, European Union University, 1820 Montreux, Switzerland

Corresponding author: Yuheng Ren (hoture@126.com)

This work was supported in part by the Research Project on Educational Information Technology in Sichuan Province under Grant DSJZXKT229, and in part by the Research Project of Neijiang Normal College under Grant HXL-19150.

**ABSTRACT** With the vigorous development of digital creativity, the image data generated by it has exploded. To effectively manage massive image data, multi-level and multi-classification management of images has become very necessary. However, the existing hierarchical classification models of deep learning images are all based on convolutional neural networks, which have limitations in capturing the underlying global features. Different from this, Transformer, as a new neural network, captures the global context information through the attention mechanism, so it performs excellently in various visual recognition tasks. However, the existing work based on Transformer does not use the hierarchical structure information in the model, making it challenging to apply the model to multi-level and multi-classification tasks of images. Therefore, this paper proposes a new image multi-level and multi-classification model, which uses multi-scale CNN to effectively capture feature information at different scales and combines it with the Transformer's ability to extract global features. At the same time, the model makes full use of the hierarchical structure information in Transformer to better understand the complex relationship of images. We have done a lot of experiments on three data sets, CIFAR-10, CIFAR-100, and CUB-200-2011, and compared the performance with the existing multi-level and multi-classification model of images. The results show that our model has higher classification accuracy and better robustness.

**INDEX TERMS** Transformer, hierarchical characteristics of the model, multi-scale convolution, multi-level and multi-classification of images.

## I. INTRODUCTION

In recent years, the flourishing development of the digital economy has propelled rapid growth in the digital creative industry [1]. Digital creativity has garnered increasing attention and demand as a crucial industry component. However, with the continuous evolution of digital creativity, the production of digital images has also seen an explosive growth trend [2]. These data may originate from different channels, covering various themes and types. Without proper classification and storage of these digital images, it will be difficult to effectively utilize these valuable data resources, potentially leading to confusion and misuse [3]. Therefore, devising intelligent and efficient methods for classifying and recognizing these structurally complex and massive image data has become challenging in digital creativity.

Today, image classification holds a crucial position in artificial intelligence learning methods. In traditional image

The associate editor coordinating the review of this manuscript and approving it for publication was Zhenbao Liu.

classification, an image is only associated with a single label from a label set [4], [5], [6], [7]. However, in real-life scenarios, the granularity of category labels is becoming increasingly nuanced and complex. An image can often be assigned multiple labels, leading to a continual expansion of the label system, where hierarchical relationships among labels may exist [8], [9], [10]. For example, a digital image of a shirt may be labeled as ''shirt,'' which can not only be classified as ''tops'' but also as ''clothing.'' In such cases, coarse-grained labels may contain multiple fine-grained labels, with each fine-grained label corresponding to only one coarse-grained label [8], [11], [12]. This classification method, which has a hierarchical structure, is called multi-level multi-class classification.

Compared with simple multi-classification, the results of multi-level multi-classification are more intuitive, which can better show the structure of prediction result labels and is easy to understand [13], [14], [15], [16]. When the corresponding image data needs to be used, it can be found and used quickly through the hierarchical structure. For example, in the digital collection system, hierarchical classification of images that need to be uploaded can reduce the manual image classification operation and, at the same time, make the image database structure more concise and clear, thus improving retrieval efficiency and accuracy. In addition, if only simple multi-classification is carried out, the visual separability between different object categories may be uneven. For example, it is relatively easy to distinguish a table from a coat, but it is slightly difficult to determine a T-shirt from a shirt. From the perspective of methodology and industry, hierarchical classification has advantages [17], [18], [19], [20]. Specifically, by distinguishing completely different types, we can reduce the misclassification of entirely different kinds of pictures as the same type, such as wrongly classifying ''shirts'' (belonging to ''tops'') as ''skirts'' (belonging to ''bottoms''). Moreover, in subsequent fine-grained classification layers, the model can focus more on learning to differentiate quite similar types of images, such as distinguishing between ''T-shirts'' and ''shirts,'' both belonging to the ''tops'' category. Employing only simple multi-classification methods would overlook these advantages.

Currently, the methods for image multi-level multi-classification based on deep learning mainly focus on Convolutional Neural Networks (CNNs), such as Hierarchical Deep Convolutional Neural Network (HD-CNN) [21], Branch Convolutional Neural Network (B-CNN) [22], Visual Tree Convolutional Neural Network (VT-CNN) [23], and Coarse-to-Fine Convolutional Neural Network (CF-CNN) [24]. However, these methods have two areas for improvement.

- All the models above utilize CNNs as their backbone networks, which extract local information from images using convolutional layers and gradually capture global features. However, they do not directly capture the overall features from the bottom layer, which can lead to issues such as high computational complexity and gradient vanishing [25].
- In handling hierarchical image features, these models typically only consider the features of each level separately and overlook how to combine and utilize features from different levels.

Because the Transformer model is excellent in learning long-distance dependence by using the attention mechanism and can capture global context information, this paper proposes a new image multi-level and multi-classification model to solve the above problems. Specifically, our contributions are as follows:

1) This paper adopts multiple convolutional kernels of different sizes to extract features from the original image, which are then fed into the Transformer. The aim is to assist the model in understanding distant dependencies and semantic information of intrinsic features at different scales, thereby enhancing its understanding and representation capability of the complexity within the image.
2) This paper utilizes hierarchical information of sample features in Transformer to delve into the intricate interplay among various elements within images, thereby enhancing the understanding of subtle complexities in image sample features.
3) The paper proposes a new multi-level, multi-classification model for images and compares its performance with several mainstream multi-level, multi-classification models in the field. The results show that the model has significantly improved the classification accuracy and robustness.

## II. RELATED WORK

### A. CURRENT SITUATION OF MULTI-LEVEL AND MULTI-CLASSIFICATION IMAGE CLASSIFICATION MODEL BASED ON DEEP LEARNING

Yan et al. first proposed HD-CNN [21] for multi-level and multi-classification of images, which was trained by multiple logical losses and new time sparseness punishment. Specifically, the model first uses an initial rough classifier to separate easily separated classes, expressed as rough classes, and then fine classes. However, its limitation lies in the need for two-step training. Therefore, Zhu and Bain put forward B-CNN [22], whose CNN layer learns effective abstract feature representation from input data in a hierarchical way, to evaluate its model's effectiveness in predicting the accuracy of image class hierarchy in descending order of abstraction. Liu et al. proposed VT-CNN [23], which combined the original CNN model with the visual tree. The VT-CNN model improves the final fine-grained category classification performance using the prior information obtained in the rough classification stage. One of the main benefits of the branching network proposed by Cho et al. [26] is that it gradually reduces the number of classes distinguished at the classification level, aiming to extract features of all granularity levels in the primary

network and transmit them to each branch. In addition, features extracted for predicting higher-level class labels are passed to branches that predict lower or finer-grained class labels, allowing networks to share features for classification. Park et al. proposed a hierarchical learning method called CF-CNN [24], which first created a class group with hierarchical association and then assigned a new label to each class with coarse and fine granularity to obtain multiple labels.

Zhou et al. [27] proposed the Deep Collaborative Multi-Task Network for hierarchical image classification. Specifically, the method first extracts the relationship matrix between every two subtasks defined by the hierarchical label structure. Then, the information for each subtask is spread to all related subtasks through these relationship matrices. Finally, a new fusion function based on task evaluation and decision uncertainty is designed to constrain the model. Li et al. proposed Multi-Task Multi-Structure Fusion (MMF) [28], which uses superclasses from different tag structures to guide and identify subclasses. Specifically, it is a deep convolutional neural network with two kinds of classification branches: one is a conventional classification branch for identifying subclasses, and the other contains multiple superclass classification branches, each responsible for determining the specific tag structure of a defined superclass. Mayouf and Dupin de Saint-Cyr put forward globally hierarchically coherent (GH-CNN) [29], which uses Bayesian rules and branch CNN to create a robust framework with a well-designed semantic loss function, which can punish hierarchy violations. Chang et al. [30] designed a multi-task learning framework to perform horizontal feature separation, aiming to separate the adverse effects of coarse-grained features from fine-grained ones.

### B. RESEARCH STATUS OF TRANSFORMER IN VISUAL FIELD

The Google team first proposed Transformer [31], a new neural network structure used in the task of seq2seq. It breaks through the limitation that circular neural networks can't realize parallel computing and has made remarkable achievements in machine translation and other tasks. Then, Dosovitskiy et al. [32] proposed the Vision Transformer (ViT) for image classification tasks. When the model is pre-trained on the large-scale data set ImageNet-21K [33] or JFT300M [34], its performance is better than the most advanced CNN. The appearance of the Transformer provides a new way for traditional visual feature learning. More and more researchers put forward some models based on Transformer. They tried to apply them to a wide range of visual tasks, such as object detection [35], image classification [36], semantic segmentation [37], image processing [38], [39], image segmentation [40], and video understanding [41].

In recent years, the ViT framework has achieved excellent performance in image classification, and more and more researchers have proposed a series of models to improve the performance of image classification tasks. Han et al. [42] proposed the Transformer in Transformer (TNT), which uses an internal transformer to transform the pixel representation

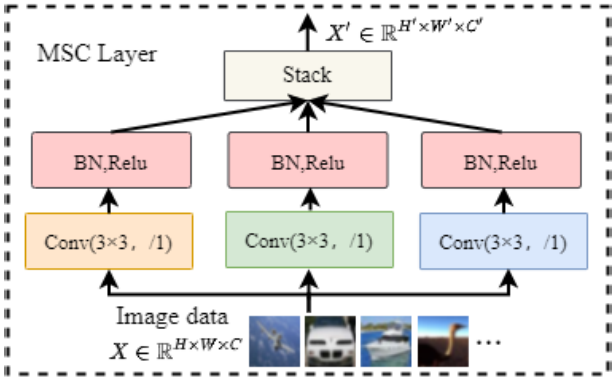


**FIGURE 1.** The overall structure of the multi-scale convolution network.

at each layer and an external transformer to take the fusion vector of patch representation and pixel representation as input. Yuan et al. [43] proposed the Tokens-to-Token Transformer (T2T-ViT), aiming at overcoming the limitation of simple labeling of input images in ViT and gradually structuring images into labels to capture rich local structural patterns. Jiang et al. [44] put forward LV-ViT, which uses new training targets to improve the performance of the ViT model. In this model, a picture is divided into multiple patches, and each patch is transformed into a token. Then, the soft label of each token is obtained by Re-labeling technology, and the image is re-labeled; thus, the image classification problem is transformed into multiple token-label recognition problems. Zhou et al. [45] proposed DeepViT, which solved the problem of attention collapse in deep ViT architecture. Chen et al. [46] believe that Transformer models without pre-training perform poorly on small datasets and, in some cases, even fall short of CNN. Therefore, they proposed a new Transformer model that does not require pre-training, which performs excellently on the CIFAR-100 dataset. Gong et al. proposed TripleFormer [47], a model that seamlessly integrates the triple self-attention mechanism with the Transformer structure. This model not only extracts local features but also captures long-range visual dependencies, thereby enhancing feature learning capabilities.

To sum up, the model based on Transformer has been applied in all directions of the CV field and has a comparable effect with CNN. Still, researchers have yet to use it for the multi-level and multi-classification task of images. Based on this, this paper designs a new image multi-level and multi-classification model, aiming to fully use the advantages of Transformer to improve the classification performance further.

## III. PROPOSED METHODS

This chapter describes the overall framework of this model in detail. To put it simply, we first use the "Multi-scale convolution layer" to extract the features of the image at different spatial scales and then process the extracted features $X' \in \mathbb{R}^{H' \times W' \times C'}$ through "liner projection of flattened

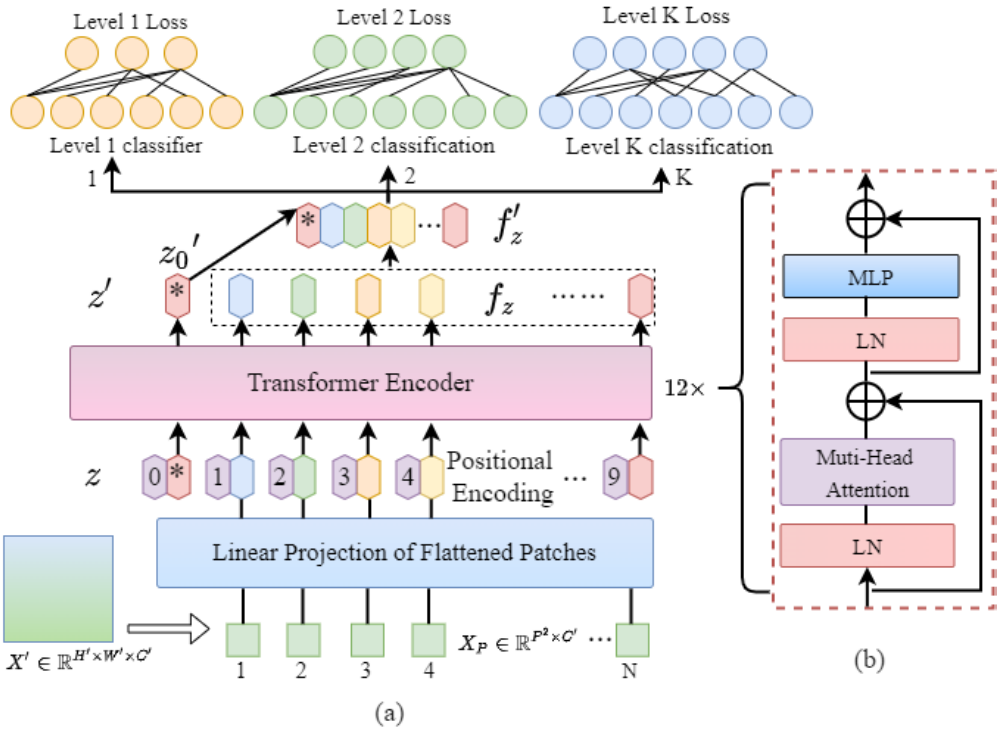


**FIGURE 2.** The general framework of multi-level and multi-classification model of images.

patches'' and send them to ''Transformer encoder'' to get the abstract feature representation $z'$ of the image. Finally, the ''hierarchical feature fusion module'' completes images' multi-level and multi-classification tasks.

### A. MULTI-SCALE CONVOLUTION LAYER

As shown in Fig 1, the multi-scale convolution layer contains three convolution kernels with different sizes, aiming to capture the feature information of images at various spatial scales. Smaller convolution kernels can capture more minor details and textures in pictures, while larger ones can capture broader context information. The multi-scale convolution layer can provide a richer and more comprehensive image representation by fusing these features of different scales, thus improving the model's ability to understand and classify images.

Specifically, suppose there is an input image $X \in \mathbb{R}^{H \times W \times C}$, where $W$ and $H$ represent the length and width of the image, and $C$ denotes the number of channels in the image. The multi-scale convolutional layer first performs multiple convolution operations on $X$ using different sizes of convolutional kernels ($3 \times 3$, $5 \times 5$, and $7 \times 7$ in this paper), obtaining feature maps at various scales. Then, these feature maps are fused in terms of channels, resulting in the multi-scale features of the image, denoted as $X'$.

$$X' = \left[\sigma\left(f^{3\times3;1}(X)\right); \sigma\left(f^{5\times5;1}(X)\right); \sigma\left(f^{7\times7;1}(X)\right)\right] \tag{1}$$

Here, $f^{3\times3;1}$ represents the convolution operation with the convolution kernel size of $3 \times 3$ and the number of channels of 1, and $\sigma$ represents the ReLU activation function. For example, suppose the input image size is $32 \times 32$, and after three convolution operations with different sizes, the sizes of the three feature maps obtained are also different. Therefore, convolution operations with different sizes need different filling sizes to promote the fusion of feature maps with different scales in channels, denoted as $X' \in \mathbb{R}^{30\times30\times3}$.

### B. LINER PROJECTION OF FLATTENED PATCHES

As shown in Fig 2. (a), we process the feature map $X' \in \mathbb{R}^{H' \times W' \times C'}$ to resemble sequence embeddings commonly used in Natural Language Processing. Precisely, we first reshape the feature map into flattened blocks $X_P \in \mathbb{R}^{P^2 \times C'}$ with dimensions $P \times P \times C'$, where $H'$ and $W'$ correspond to the length and width of the feature map, respectively, $C'$ is the number of channels in the feature map, and $P$ is the length and width of each block. This results in $N = \frac{H \times W}{P^2}$ flattened blocks. Then, each block is mapped to a one-dimensional vector $x_p \in \mathbb{R}^D$ via a fully connected layer. Finally, the processed flattened blocks are concatenated with the class token and used as input by the Transformer encoder. The formula is as follows:

$$x_p = X_P E \tag{2}$$

$$z = \left[x_{\text{class}}; x_p^1; x_p^2; \ldots; x_p^N\right] + E_{\text{pos}}, E_{\text{pos}} \in \mathbb{R}^{(N+1)\times D} \tag{3}$$

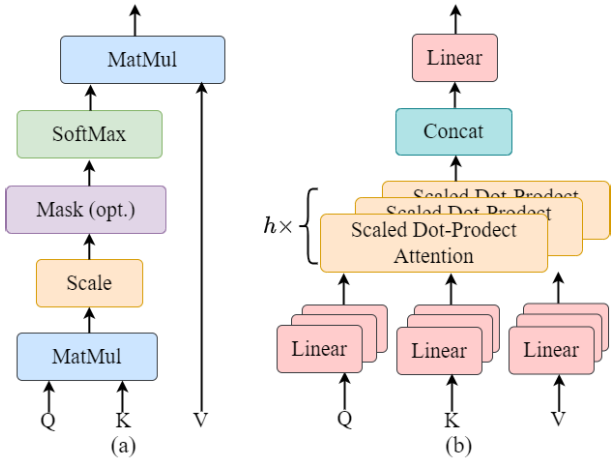


**FIGURE 3. (a) Overall process of self-attention. (b) Multi-head attention block.**

where $z_0$ is a learnable category word artificially added to realize the classification task, the vector is numbered "0" in Fig 2. The vector does not contain any image information or features but realizes information aggregation or interaction of image features through a self-attention mechanism after being connected in series with image block symbols. $E$ is a linear mapping matrix with an input size of P × P × C and an output size of D. $E_{\text{pos}}$ stands for position coding, which adds information to each embedded vector. Position coding is usually introduced by sine or cosine functions, as follows:

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{\frac{2i}{d}}}\right)$$
$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{\frac{2i}{d}}}\right) \quad (4)$$

where $pos$ is the position of the flat block in the feature map, $i$ is the embedded position and the position-coding changes with the change in the position of the flat block.

### C. TRANSFORMER ENCODER

As depicted in Fig 2. (b), The transformer encoder primarily comprises a multi-head attention mechanism and a Multi-layer perceptron. A critical component of the Transformer, as shown in Fig 3. (a), the attention mechanism operates as follows: initially, the input sequences $M \in \mathbb{R}^{n_m \times d_m}$ and $H \in \mathbb{R}^{n_h \times d_h}$ are linearly transformed into three distinct sequence vectors: Q (query), K (key), and V (value), where $n$ and $d$ denote the length and dimensionality of the input sequences, respectively.

$$Q = MW^q, \quad K = HW^k, \quad V = HW^v \quad (5)$$

where $W^q \in \mathbb{R}^{d_m \times d^k}$, $W^k \in \mathbb{R}^{d_h \times d^k}$, and $W^v \in \mathbb{R}^{d_h \times d^v}$ are all linear matrices, $d^k$ is the dimension of query and key, and $d^v$ is the value dimension. The query is projected from $M$, and the key and value are projected from $H$. These two sequential input schemes are called cross-attention mechanisms.

Then, the attention layer integrates the query with the corresponding key, aggregates the obtained result with value again, and outputs the updated vector. The formula is as follows:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (6)$$

Attention weight is generated by dot product operation between query and key, then adjusted by scaling factor $\sqrt{d_k}$, and finally normalized by Softmax function. The Softmax function is defined as follows:

$$y_i = \frac{e^{u_i}}{\sum_j e^{u_j}} \quad (7)$$

where $u_i$ represents the scaled value of position $i$, and $y_i$ represents the normalized value of the corresponding position. The normalized weights are assigned to the corresponding elements in value, thus generating the final output vector.

Due to the limitation of feature subspaces, the modeling capability of single-head attention modules is insufficient to achieve the desired level of refinement. To enhance the performance of self-attention layers, Vaswani et al. proposed multi-head attention, which significantly improves the performance of conventional self-attention layers, as shown in Fig 3. (b). This mechanism is achieved by assigning different query, key, and value matrices to different attention heads. The formula is as follows:

$$Q = MW^{q_i}, \quad K = HW^{k_i}, \quad V = HW^{v_i} \quad (8)$$
$$Z_i = \text{Attention}(Q_i, K_i, V_i), \ i = 1 \ldots h \quad (9)$$
$$\text{MultiHead}(Q, K, V) = \text{Concat}(Z_1, Z_2, \ldots, Z_h)W^o \quad (10)$$

where: $i$ represents the head number, ranging from 1 to $h$, $W^o \in \mathbb{R}^{hd_v \times D}$ represents the output projection matrix, $Z_i$ represents the output matrix of each head, and $W^{q_i} \in \mathbb{R}^{D \times d^k}$, $W^{k_i} \in \mathbb{R}^{D \times d^k}$, and $W^{v_i} \in \mathbb{R}^{D \times d^v}$ are three different linear matrices. Therefore, the vector $z$ obtained from the "Liner Projection of Flattened Patches" layer can be further processed by the Transformer encoder to obtain:

$$z' = z + \text{Multihead}(\text{LN}(z)) + \text{MLP}\,(z + \text{Multihead}(\text{LN}(z))) \quad (11)$$

### D. HIERARCHICAL FEATURE FUSION MODULE

Existing image classification models based on Transformers typically only utilize the final classification token to perform the classification task without fully exploiting the information from other image patches. Xie et al. [48] conducted extensive experiments and found that image patch tokens contain rich information and are highly beneficial for image classification tasks. Inspired by this, our model first performs average pooling on the image patch tokens $f_z = \{z_i' \in \mathbb{R}, i = 1, \ldots, N\}$ outputted by the Transformer encoder module and then concatenates them with the class token $z_0'$. The formula is as follows:

$$f_z' = \text{AvgPool}(f_z) + z_0' \quad (12)$$

**TABLE 1.** Two kinds of two-level labels and three-level label division table of the CIFAR-10 data set.

| Number of levels | - | size of image | Level | Number of categoriesl | Specific category |
|---|---|---|---|---|---|
| three-level | - | 32×32 | 1 | N=2 | transport,animal |
| | - | 32×32 | 2 | N=7 | sky,water,road,bird,reptile,pet,medium |
| | - | 32×32 | 3 | N=10 | airplane,ship,automobile,truck,bird,frog,cat,dog,deer,horse |
| two-level | 2-10 case | 32×32 | 1 | N=2 | transport,animal |
| | | 32×32 | 2 | N=10 | airplane,ship,automobile,truck,bird,frog,cat,dog,deer,horse |
| two-level | 7-10 case | 32×32 | 1 | N=7 | sky,water,road,bird,reptile,pet,medium |
| | | 32×32 | 2 | N=10 | airplane,ship,automobile,truck,bird,frog,cat,dog,deer,horse |

**TABLE 2.** Two kinds of two-level classification accuracy of different models on CIFAR-10 test set.

| Method | Backbone | Size of image | "2-10" case | | "7-10" case | |
|---|---|---|---|---|---|---|
| | | | Level 1 Acc | Level 2 Acc | Level 1 Acc | Level 2 Acc |
| CF-CNN | CNN | 32×32 | - | 82.89 | 76.13 | 84.56 |
| B-CNN | CNN | 32×32 | 95.42 | 85.47 | 84.95 | 84.34 |
| ViT | Transformer | 32×32 | 97.33 | 96.45 | 97.42 | 96.77 |
| Add-Vit | Transformer | 32×32 | 97.16 | 96.68 | 97.34 | 97.05 |
| TripleFormer-T | Transformer | 32×32 | 98.15 | 96.71 | 97.67 | 96.87 |
| TripleFormer-S | Transformer | 32×32 | **98.62** | 97.15 | 98.04 | 97.21 |
| **Ours** | **CNN+Transformer** | **32×32** | 98.37 | **97.76** | **98.31** | **97.45** |

Then, feature $f_z'$ is fed into $K$ different fully connected layers to obtain corresponding multi-level multi-classification results for images. Here, $K$ represents the number of layers for multi-level image classification, so if it is a two-level multi-classification task, $K = 2$.

The loss function of the multi-level multi-classification model in this article is the sum of the classification losses at each level. Specifically, the loss function formula for the $i-th$ level of classification is as follows:

$$L_C^i = -\sum_{0}^{m} y_{ij} \log(\hat{y}_{ij}) \tag{13}$$

$y_{ij}$ represents the predicted output of class $j$ in the $i-th$ layer, and $\hat{y}_{ij}$ represents the true label of class $j$ in the $i-th$ layer, with $m$ denoting the number of classes to be classified in the $i-th$ layer. Therefore, the total loss of the model is represented as follows:

$$L_C = \sum_{i=1}^{K} L_C^i \tag{14}$$

## IV. EXPERIMENTS AND ANALYSIS

### A. EXPERIMENTAL PARAMETER SETTINGS AND DETAILS

All the experiments in this chapter are carried out on the Rongtian server based on the Ubuntu 16.8 operating system, equipped with 64GB of RAM and 4 GPUs of the GeForce RTX 2080Ti model. The deep learning framework adopts PyTorch, and the Python version is 3.9.0. The model in this chapter uses ViT-L/16 pre-training weights trained by official Vit on the ImageNet21k data set, and the optimizer adopts SGD, with momentum set to 0.9, and the number of iterations of all data sets is limited to 20 times. This chapter's multi-level and multi-classification experiments are carried out on three public data sets: CIFAR-10, CIFAR-100, and CUBR-200-2011. In the experiments of CIFAR-10 and CIFAR-100, the batch size is set to 8, and in the experiments of CUBR-200-2011, it is set to 3.

### B. CIFAR-10 DATASET

The CIFAR-10 data set was collected by Krizhevsky et al. [49], containing 60,000 RGB images with a pixel size 32 × 32.

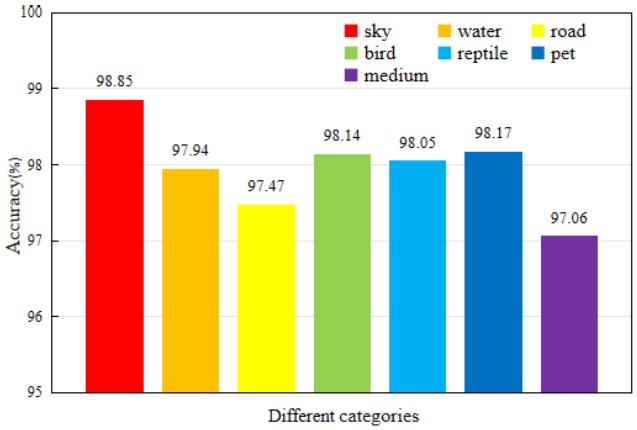


**FIGURE 4.** The classification accuracy of our model in the "level 2" specific category of the CIFAR-10 test set under the three-level classification scenario.

**TABLE 3.** Three-level classification accuracy of different models on CIFAR-10 test set.

| Method | Level 1 Acc | Level 2 Acc | Level 3 Acc |
|---|---|---|---|
| CF-CNN | - | - | 83.12 |
| B-CNN | 96.32 | 86.74 | 82.19 |
| VT-CNN | - | - | 89.51 |
| ViT | 97.23 | 97.38 | 96.59 |
| Add-Vit | 97.08 | 97.26 | 97.11 |
| TripleFormer-T | - | - | 96.87 |
| TripleFormer-S | 98.16 | 97.83 | 96.82 |
| **Ours** | **98.65** | **97.96** | **97.44** |

This data set includes ten categories of images, including Dog, Airplane, Ship, Automobile, Deer, Bird, Cat, Frog, Horse, and Truck. There are 6000 images in each Category, and the designs between categories are mutually exclusive. For example, "Dog" does not contain "cat." The 60,000 images in this data set are divided into 50,000 training images and 10,000 test images.

Due to the lack of hierarchical labels in CIFAR-10, this study divides CIFAR-10 into two types of two-level multi-classification scenarios based on specific conditions. The first scenario involves splitting the first level into two classes and the second level into ten classes, referred to as the "2-10 case;" the second scenario involves splitting the first level into seven classes and the second level into ten classes, referred to as the "7-10 case." Additionally, this study adopts the manually partitioned three-level classification label tree from B-CNN [21]. The hierarchical multi-classification label division of CIFAR-10 is shown in Table 1.

**TABLE 4.** The comparison of different models in terms of parameter count and Flops.

| Method | Backbone | Params | Flops |
|---|---|---|---|
| CF-CNN | CNN | 21.9M | 3.4G |
| B-CNN | CNN | 25.6M | 4.0G |
| VT-CNN | CNN | 44.7M | 7.2G |
| ViT | Transformer | 32M | 5.1G |
| Add-Vit | Transformer | 26.8M | 4.2G |
| TripleFormer-T | Transformer | 17.7M | 2.7G |
| TripleFormer-S | Transformer | 37.4M | 6.1G |
| **Ours** | **CNN+Transformer** | 29.8M | 4.8G |

First, we do experiments on the CIFAR-10 data set and compare the performance of our model with CF-CNN, B-CNN, Vit, Add-vit, TripleFormer-T, and TripleFormer-S under two kinds of secondary classification. The results are shown in Table 2. Our model has achieved ideal results in both cases. Specifically, compared with B-CNN, the best-performing network with CNN as the backbone, in the case of the "2-10 case," our accuracy in the first and second layers is improved by 2.95% and 12.29%, respectively. In the "7-10 case" case, our accuracy in the first and second layers increased by 13.36% and 13.11%, respectively. Compared to the state-of-the-art Transformer-based backbone network TripleFormer-S, in the "2-10 case," we improved accuracy by 0.61% in the second layer. In the "7-10 case," we achieved accuracy improvements of 0.55% and 0.24% in the first and second layers, respectively.

Then, we conducted experiments on the CIFAR-10 dataset for three-level classification, comparing the performance of our model with CF-CNN, B-CNN, VT-CNN, Vit, Add-vit, TripleFormer-T and TripleFormer-S in terms of accuracy on the test set, as shown in Table 3. Our proposed model outperforms existing models in terms of accuracy at each layer for three-level classification. Specifically, compared to B-CNN, which performs the best among CNN-based backbone networks, our model improves the accuracy by 2.33%, 11.22%, and 15.25% for the first, second, and third layers, respectively. Compared to TripleFormer-S, which utilizes Transformer as the backbone network and performs best, our model achieved improvements in accuracy of 0.49%, 0.13%, and 0.62% in the first, second, and third layers, respectively. By observing Tables 2 and 3, we also notice a gradual decrease in accuracy as the number of layers increases (i.e., the model needs to classify more categories). This suggests that the model's performance may face challenges in more complex multi-level classification tasks.

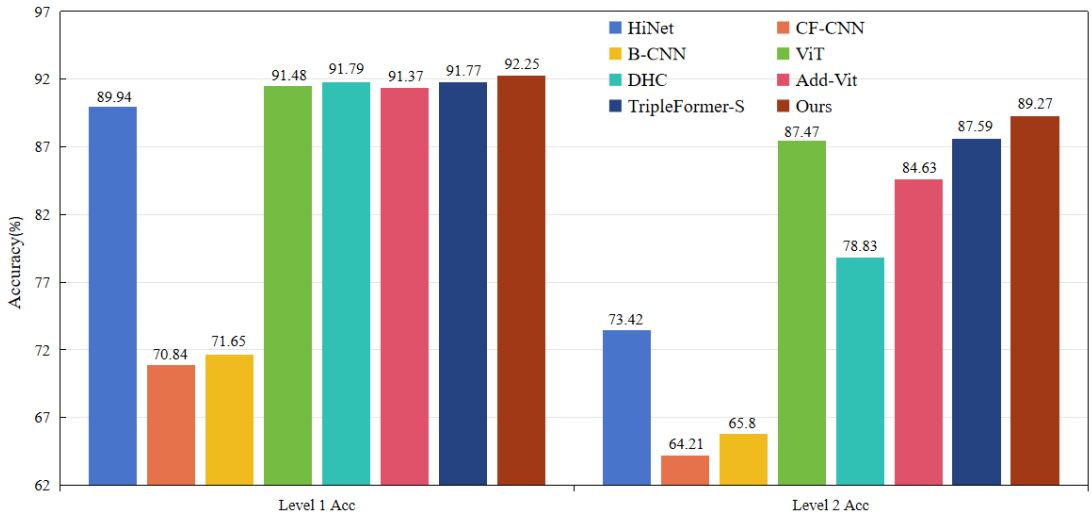


**FIGURE 5.** Two-level classification accuracy of different models on CIFAR-100 test set.

**TABLE 5.** Three-level classification accuracy of different models on CIFAR-100 test set.

| Method | Backbone | Size of image | Level 1 Acc | Level 2 Acc | Level 3 Acc |
|---|---|---|---|---|---|
| CF-CNN | CNN | 32×32 | - | - | 64.37 |
| B-CNN | CNN | 32×32 | 72.71 | 70.4 | 64.62 |
| VT-CNN | CNN | 32×32 | - | - | 74.13 |
| ViT | Transformer | 32×32 | 92.70 | 90.88 | 84.97 |
| Add-Vit | Transformer | 32×32 | 92.53 | 91.12 | 84.77 |
| TripleFormer-T | Transformer | 32×32 | 92.94 | 91.07 | 86.44 |
| TripleFormer-S | Transformer | 32×32 | 93.25 | 91.38 | 87.51 |
| **Ours** | **CNN+Transformer** | **32×32** | **94.65** | **92.14** | **89.63** |

In addition, we also compared the accuracy of each category in the intermediate layers of our model under the three-level classification scenario, as shown in Fig 4. Finally, we compared the parameter counts and Flops of different models to assess their computational complexity and performance, detailed in Table 4. Larger parameter counts and FLOPS indicate greater model complexity, requiring more computational resources for training and inference.

### C. CIFAR-100 DATASET

Similar to the CIFAR-10 data set, because the CIFAR-100 data set does not provide three-level classification labels, this paper adopts the hierarchical label tree manually divided in B-CNN [21]. Similar to the classification of the CIFAR-10 data set in Table 1, the three-level classification label of the CIFAR-100 data set contains 8, 20, and 100 categories, respectively, and the correlation between categories is strong. For example, fish include flounder, rays, sharks, etc., so different categories under the same parent class have strong commonness, similar to organisms' hierarchical structure. In addition, the CIFAR-100 data set also provides an officially classified second-level classification hierarchy label, which divides 100 categories into 20 parent categories; that is, the first and second layers contain 20 and 100 categories, respectively.

In the experimental part of this small chapter, we tested it on the CIFAR-100 data set. Firstly, we compare the two-level classification accuracy of our model with other popular multi-level image classification models on the CIFAR-100 test set, and the results are shown in Fig 5. Specifically, in the first-level classification task, compared with the best DHC, which uses CNN as the backbone

**TABLE 6.** Three-level classification accuracy of different models on the CUBR-200-2011 test set.

| Method | Backbone | Size of image | Level 1 Acc | Level 2 Acc | Level 3 Acc |
|---|---|---|---|---|---|
| F-GN | CNN | 84×84 | 96.37 | 90.39 | 77.95 |
| DC-MCL | CNN | 84×84 | **96.58** | 90.36 | 77.85 |
| **Ours** | **CNN+Transformer** | **84×84** | 96.43 | **90.84** | **79.63** |

**TABLE 7.** Three-level classification accuracy of different models on the CUBR-200-2011 test set.

| MSCL | HFFM | Transformer | Level 1 Acc | Level 2Acc | Level 3 Acc |
|---|---|---|---|---|---|
| ✗ | ✗ | ✓ | 92.70 | 90.88 | 84.97 |
| ✗ | ✓ | ✓ | 93.58 | 91.33 | 85.65 |
| ✓ | ✗ | ✓ | 93.37 | 91.86 | 87.72 |
| ✓ | ✓ | ✓ | 94.65 | 92.14 | 89.63 |

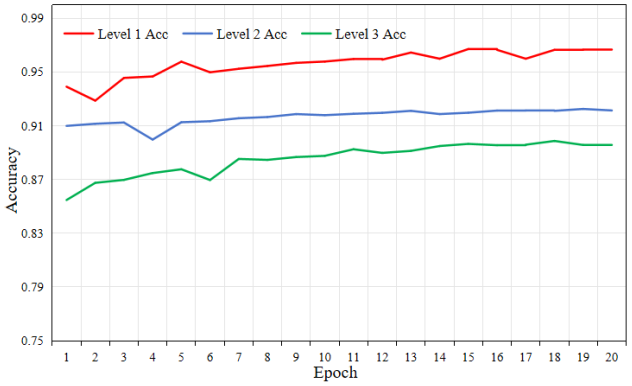


**FIGURE 6.** The curve of three-level classification accuracy on the CIFAR-100 test set during our model training.

network, our model improves the accuracy by 0.46%. In the case of the second-level classification, compared with the top-performing TripleFormer-S model with Transformer backbone, our model improved accuracy by 1.68%. On the whole, when our model is compared with TripleFormer-S, which performs well in the two-level classification, the accuracy of the first-level classification and the second-level classification is improved by 0.48% and 1.68% respectively, which shows the superior performance of our model in the second-level classification task on CIFAR-100 data set.

Then, we conducted a three-level classification experiment on the CIFAR-100 data set and compared our model with B-CNN, CF-CNN, VT-CNN, Vit, Add-Vit, TripleFormer-T and TripleFormer-S on the test set. The results are shown in Table 5. The accuracy of this model is 94.65%, 92.14%, and 89.63% at the first, second, and third levels of three-level classification, which is far superior to other existing models.

In addition, Fig 6 shows the change curve of classification accuracy of each level in the training process of this model. It can be observed from the figure that the training speed of the ''Level 1'' level is the fastest, and the classification effect is the best among the three levels. The experimental results are consistent with common sense understanding because the classification of the first level is the simplest compared with the classification of other levels, and only eight categories must be distinguished.

On the contrary, the classification difficulty of the third level is the most challenging of the three levels, and it is necessary to classify 100 categories in the data set. The training situation of the second level is between the first and third levels, which is consistent with reality. As the number of classifications becomes more and more fine-grained, the corresponding classification difficulty gradually increases.

### D. CUBR-200-2011 DATASET

California Institute of Technology collected the CUB-200-2011 data set [50], and its full name is Caltech-UCSD Birds-200 2011. This data set is an expanded version of CUB-200, which contains 11,788 pictures covering 200 species of birds—the official designated 5994 pictures for training and 5794 for verification. For the CUB-200-2011 data set, a three-level label reorganized by Chang et al. [30] is adopted in this study. The first, second, and third levels include 13, 38, and 200 categories.

This subsection conducted a three-level classification experiment on the CUB-200-2011 dataset. In this experiment, our model was compared with F-GN [30] and DC-MCL [51] regarding classification accuracy on each level of the CUB-200-2011 test set, as shown in Table 6. Our model performed well in classification accuracy on the second and third levels,

achieving 90.84% and 79.63%, respectively. However, the performance on the first level was not satisfactory. This indicates that our model demonstrates strength in handling fine-grained classification tasks but struggles with coarser-grained classification.

Finally, we conducted ablation experiments on the CIFAR-100 test set, as shown in Table 7. MSC represents the Multi-scale Convolution Layer, and HFFM represents the Hierarchical Feature Fusion Module, both crucial components introduced in Chapter 2. Transformer serves as the backbone framework of the model and is indispensable. From the experimental results, it is evident that the incorporation of MSC and HFFM not only improves the model's accuracy in classification tasks but also enhances its capability to handle complex features and multi-scale information, validating their effectiveness.

## V. CONCLUSION

With the development of computers and the continuous advancement of technology, the quantity of image data is experiencing explosive growth, making effective management of massive image data crucial. The multi-level and multi-classification model of deep learning images mainly uses CNN as the backbone network. However, due to the limited receptive field of CNN, it is difficult to capture the overall characteristics of the sample, and there may even be problems such as extensive computation and gradient disappearance. Therefore, this paper proposes a novel multi-level multi-classification model for images to efficiently manage vast image data, improve classification accuracy, and enhance robustness. Specifically, the proposed model effectively combines multi-scale CNN and Transformer. By utilizing multi-scale CNN to capture feature information at different scales and integrating it with the capability of Transformer to extract global features while fully exploiting the hierarchical structure information in Transformer, the model aims to understand the complex relationships within images better. Extensive experimental results on the CIFAR-10, CIFAR-100, and CUB-200-2011 datasets demonstrate that the proposed model achieves higher classification accuracy and better robustness.

## REFERENCES

[1] V. Scuotto, T. Tzanidis, A. Usai, and R. Quaglia, "The digital humanism era triggered by individual creativity," *J. Bus. Res.*, vol. 158, Mar. 2023, Art. no. 113709.
[2] C. Jones, N. Askin, F. Godart, S. Harvey, and D. Phillips, "Creative industries: Challenges and opportunities of digital technologies," *Acad. Manage. Discoveries*, Sep. 2023.
[3] W. Wang, Y. Sun, W. Li, and Y. Yang, "TransHP: Image classification with hierarchical prompting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2024, pp. 1–14.
[4] Y. Xu, S. Wu, B. Wang, M. Yang, Z. Wu, Y. Yao, and Z. Wei, "Two-stage fine-grained image classification model based on multi-granularity feature fusion," *Pattern Recognit.*, vol. 146, Feb. 2024, Art. no. 110042.
[5] J. J. Bird and A. Lotfi, "CIFAKE: Image classification and explainable identification of AI-generated synthetic images," *IEEE Access*, vol. 12, pp. 15642–15650, 2024.
[6] X. Huo, G. Sun, S. Tian, Y. Wang, L. Yu, J. Long, W. Zhang, and A. Li, "HiFuse: Hierarchical multi-scale feature fusion network for medical image classification," *Biomed. Signal Process. Control*, vol. 87, Jan. 2024, Art. no. 105534.
[7] M. V. Todescato, L. F. Garcia, D. G. Balreira, and J. L. Carbonera, "Multiscale patch-based feature graphs for image classification," *Expert Syst. Appl.*, vol. 235, Jan. 2024, Art. no. 121116.
[8] C. Anderson, M. Gwilliam, E. Gaskin, and R. Farrell, "Elusive images: Beyond coarse analysis for fine-grained recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2024, pp. 829–839.
[9] Q. Gao, T. Long, and Z. Zhou, "Mineral identification based on natural feature-oriented image processing and multi-label image classification," *Expert Syst. Appl.*, vol. 238, Mar. 2024, Art. no. 122111.
[10] W. Zhang, Y. Zhao, Y. Gao, and C. Sun, "Re-abstraction and perturbing support pair network for few-shot fine-grained image classification," *Pattern Recognit.*, vol. 148, Apr. 2024, Art. no. 110158.
[11] S. Zhang, S. Zheng, Z. Shui, and L. Yang, "HLS-FGVC: Hierarchical label semantics enhanced fine-grained visual classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2024, pp. 7370–7374.
[12] Z. Xu and S. Zhao, "Fine-grained urban blue-green-gray landscape dataset for 36 Chinese cities based on deep learning network," *Sci. Data*, vol. 11, no. 1, pp. 1–17, Mar. 2024.
[13] J. Ren, C. Li, Y. An, W. Zhang, and C. Sun, "Few-shot fine-grained image classification: A comprehensive review," *AI*, vol. 5, no. 1, pp. 405–425, Mar. 2024.
[14] Y. Pu, Y. Han, Y. Wang, J. Feng, C. Deng, and G. Huang, "Fine-grained recognition with learnable semantic data augmentation," *IEEE Trans. Image Process.*, vol. 33, pp. 3130–3144, 2024.
[15] Q. Zhou, K. Zhang, F. Yue, Z. Zhang, and H. Yu, "Naming conventions-based multi-label and multi-task learning for fine-grained classification," in *Proc. Int. Conf. Algorithm, Imag. Process., Mach. Vis. (AIPMV)*, Jan. 2024, pp. 316–322.
[16] W. Xiong, Z. Xiong, L. Yao, and Y. Cui, "Cog-Net: A cognitive network for fine-grained ship classification and retrieval in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024.
[17] Z. Wang, Z. Chen, and B. Du, "Active learning with co-auxiliary learning and multi-level diversity for image classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 3899–3911, Aug. 2023.
[18] H. Yu, X. Zhang, Y. Wang, Q. Huang, and B. Yin, "Fine-grained accident detection: Database and algorithm," *IEEE Trans. Image Process.*, vol. 33, pp. 1059–1069, 2024.
[19] R. Wu, J. Bai, W. Li, and J. Jiang, "DCNet: Exploring fine-grained vision classification for 3D point clouds," *Vis. Comput.*, vol. 40, no. 2, pp. 781–797, Feb. 2024.
[20] M. Rajesh Khanna, "Multi-level classification of Alzheimer disease using DCNN and ensemble deep learning techniques," *Signal, Image Video Process.*, vol. 17, no. 7, pp. 3603–3611, Oct. 2023.
[21] Z. Yan, H. Zhang, R. Piramuthu, V. Jagadeesh, D. DeCoste, W. Di, and Y. Yu, "HD-CNN: Hierarchical deep convolutional neural networks for large scale visual recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2740–2748.
[22] X. Zhu and M. Bain, "B-CNN: Branch convolutional neural network for hierarchical classification," 2017, *arXiv:1709.09890*.
[23] Y. Liu, Y. Dou, R. Jin, and P. Qiao, "Visual tree convolutional neural network in image classification," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 758–763.
[24] J. Park, H. Kim, and J. Paik, "CF-CNN: Coarse-to-fine convolutional neural network," *Appl. Sci.*, vol. 11, no. 8, p. 3722, Apr. 2021.
[25] W. Liu and X. Lu, "Research progress of transformer based on computer vision," *Comput. Eng. Appl.*, vol. 58, no. 6, pp. 1–16, 2022.
[26] H. Cho, C. Ahn, K. M. Yoo, J. Seol, and S.-G. Lee, "Leveraging class hierarchy in fashion classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 3197–3200.
[27] Y. Zhou, X. Li, Y. Zhou, Y. Wang, Q. Hu, and W. Wang, "Deep collaborative multi-task network: A human decision process inspired model for hierarchical image classification," *Pattern Recognit.*, vol. 124, Apr. 2022, Art. no. 108449.
[28] X. Li, Y. Zhou, Y. Zhou, and W. Wang, "MMF: Multi-task multi-structure fusion for hierarchical image classification," in *Proc. Int. Conf. Artif. Neural Netw.* Cham, Switzerland: Springer, 2021, pp. 61–73.
[29] M.-S. Mayouf and F. D. de Saint-Cyr, "GH-CNN: A new CNN for coherent hierarchical classification," in *Proc. Int. Conf. Artif. Neural Netw.* Cham, Switzerland: Springer, 2022, pp. 669–681.

[30] D. Chang, K. Pang, Y. Zheng, Z. Ma, Y.-Z. Song, and J. Guo, "Your 'flamingo' is my 'bird': Fine-grained, or not," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11471–11480.
[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.
[32] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16 × 16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
[33] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[34] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 843–852.
[35] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 213–229.
[36] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.
[37] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. S. Torr, and L. Zhang, "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 6877–6886.
[38] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, Z. Liu, S. Ma, C. Xu, C. Xu, and W. Gao, "Pre-trained image processing transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 12294–12305.
[39] J. Shi, Y. Wang, Z. Yu, G. Li, X. Hong, F. Wang, and Y. Gong, "Exploiting multi-scale parallel self-attention and local variation via dual-branch transformer-CNN structure for face super-resolution," *IEEE Trans. Multimedia*, vol. 26, pp. 2608–2620, 2024.
[40] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 1280–1289.
[41] Y. Zeng, J. Fu, and H. Chao, "Learning joint spatial–temporal transformations for video inpainting," in *Proc. 16th Eur. Conf. Comput. Vis. (ECCV)*, Glasgow, U.K. Cham, Switzerland: Springer, 2020, pp. 528–543.
[42] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, "Transformer in transformer," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 15908–15919.
[43] L. Yuan, Y. Chen, T. Wang, W. Yu, Y. Shi, Z. Jiang, F. E. H. Tay, J. Feng, and S. Yan, "Tokens-to-token ViT: Training vision transformers from scratch on ImageNet," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 538–547.
[44] Z.-H. Jiang, Q. Hou, L. Yuan, D. Zhou, Y. Shi, X. Jin, A. Wang, and J. Feng, "All tokens matter: Token labeling for training better vision transformers," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 18590–18602.
[45] D. Zhou, B. Kang, X. Jin, L. Yang, X. Lian, Z. Jiang, Q. Hou, and J. Feng, "DeepViT: Towards deeper vision transformer," 2021, *arXiv:2103.11886*.
[46] J. Chen, P. Wu, X. Zhang, R. Xu, and J. Liang, "Add-ViT: CNN-transformer hybrid architecture for small data paradigm processing," *Neural Process. Lett.*, vol. 56, no. 3, p. 198, Jun. 2024.
[47] Y. Gong, P. Wu, R. Xu, X. Zhang, T. Wang, and X. Li, "TripleFormer: Improving transformer-based image classification method using multiple self-attention inputs," *Vis. Comput.*, pp. 1–12, Mar. 2024.
[48] J. Xie, R. Zeng, Q. Wang, Z. Zhou, and P. Li, "SoT: Delving deeper into classification head for transformer," 2021, *arXiv:2104.10935*.
[49] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Tech. Rep., 2009.
[50] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD birds-200-2011 dataset," Tech. Rep., 2011.
[51] D. Chang, Y. Ding, J. Xie, A. K. Bhunia, X. Li, Z. Ma, M. Wu, J. Guo, and Y.-Z. Song, "The devil is in the channels: mutual-channel loss for fine-grained image classification," *IEEE Trans. Image Process.*, vol. 29, pp. 4683–4695, 2020.

**QUANDENG GOU** received the Ph.D. degree from Kharkiv National University of Economics, Ukraine, in March 2024. He is currently an Associate Professor. He has published more than 20 journal articles, written four textbooks, hosted or participated in more than ten research projects, and applied for more than ten invention patents or software copyrights. His main research interests include personalized learning, machine learning, and image processing.

**YUHENG REN** received the joint Ph.D. degree in economics from the University of Colombo and The University of Arizona. He is currently a Research Chair Professor with Belarusian State University. He has published more than 20 journal articles, written 15 monographs, chaired or participated in 32 research projects, and applied for more than 26 invention patents or software copyrights. His main research interests include industrial intelligent manufacturing technologies and digital economy.