Mixing Pooling Instead of Class Token Representation Learning for Vision Transformer

Zhenrui Huang
Ningbo Tech University, Ningbo, China
e-mail: huangzhenrui@nbt.edu.cn

Libo Xu*
Ningbo Tech University, Ningbo, China
* Corresponding author: xlb@nbt.edu.cn

Yucheng Sun
China E-Port Data Center Ningbo Branch, Ningbo, China
e-mail: sunyc@qq.com

Jiagong Wang
Ningbo Tech University, Ningbo, China
e-mail: 3190439064@nit.zju.edu.cn

Lingchao Zhu
Ningbo Tech University, Ningbo, China
e-mail: 386303983@nbt.edu.cn

***Abstract*—With the introduction of Transformer technology, Vision Transformer (ViT) has gained promising results in the field of CV. In the ViT model, the authors use adding class token at the beginning of the sequence to classify the images in order to avoid bias towards a particular token in the input sequence. Through our study, we found that using global pooling to downscale the last layer of the output image sequence not only reduces the computational effort but also allows extracting more effective features than a single class token. In the paper, we propose a new pooling method called mixing pool method of all the output sequence to replace class token of original Vision Transformer, which can extract better features and improve model performance. Through extensive experiments, we demonstrate that Vision Transformer model with mixing pool achieves significant improvement on the original class token. The pooling modules can be used as an alternative to class token in other Vision Transformer models.**

Keywords-component; ViT; Pooling method; class token;

I. Introduction

Recently, Transformer has been introduced to image classification, it demonstrates that Vision Transformer (ViT) [1] has the potential to replace the CNN backbone. Among them, ViT chooses to add a class token at the head of the sequence as an input for classification prediction in order to avoid bias in prediction for a particular token in the sequence. We believe that this structure has some limitations: a single class token only learns limited features in the process of shrinking image resolution and increasing number of channels throughout the model, and the remaining sequence stores more classification features. Therefore, we propose a new pooling method called mixing pooling. Using mixing pooling on the output sequence instead of ViT's class token, better features are extracted and model performance is improved.

Overall, our contributions are three-fold: (1) We propose a new pooling method called mixing pooling. By performing average pooling and maximum pooling on image rows and columns separately, we integrate image background information about texture information to extract more feature information. (2) We use integrated pooling instead of ViT's class token to effectively extract global and personalized features of images. (3) By synthetically comparing several pooling methods of extensive experiments on Cifar10 [2] dataset, we demonstrate that our proposed pooling methods extracts more classification features, achieving the better results than traditional class token in vision transformers.

II. Related Work

ViT was the first to demonstrate that Transformer also achieves the better performance in image classification with sufficient training data (e.g., ImageNet22k [3]). DeiT [4] further explores an efficient training strategy and a distillation approach for ViT. However, ViT is columnar in structure, and the model can only output the features of one layer, which has the disadvantage of relatively high number of parameters and computational cost. Various optimization schemes have been proposed to make ViT more efficient in processing high resolution images and in various downstream tasks.

PiT [5] takes inspiration for CNN backbone and proposes to add pooling layers to ViT, increase the number of channels of class token and feature maps with the depth of the model, and decrease the spatial sizes of the feature map. ConViT [6] brings the inductive bias of CNN into transformer. TNT [7] utilizes inner and outer Transformer blocks to generate pixel embeddings and patch embeddings respectively. CPVT [8] replaces the fixed size position embedding in ViT with conditional position encodings, making it easier to process images of arbitrary resolution. LV-ViT [9] provides a unique label for each image block of with its corresponding token called token labeling, which utilizes the dense score map of each training image and computes the cross-entropy loss between each token and its corresponding label as an auxiliary loss. Swin Transformer [10] replaces fixed-size position embeddings with relative position deviations and restricts self-attentiveness within a moving window. CvT [11], CoaT [12] and LeViT [13] introduce convolutional operations into visual transformers. Twins [14] combines local attention and global attention mechanisms to obtain stronger feature representations. CrossViT [15] processes image blocks of different sizes through a two branch transformer. PVT [16] implements several different levels of feature output by introducing a pyramid structure, and at the last level feature maps Adding class token to obtain information of higher level features. PVTv2[17] improves PVT by adding three improved designs: (1) locally continuous features of convolution, (2) positional encode with zero padding, and (3) averaging pooling.

Although some above optimization schemes try to use average pooling instead of class token for classification prediction in ViT, we consider that average pooling does not retain the sufficient texture and characteristics information of images. So we propose more pooling methods instead of class token.

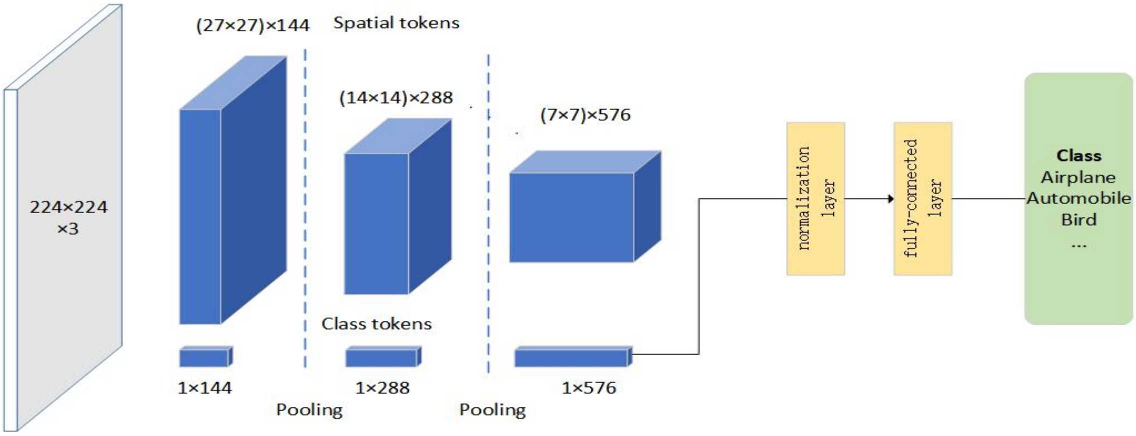


Figure 1 The PiT model using class token

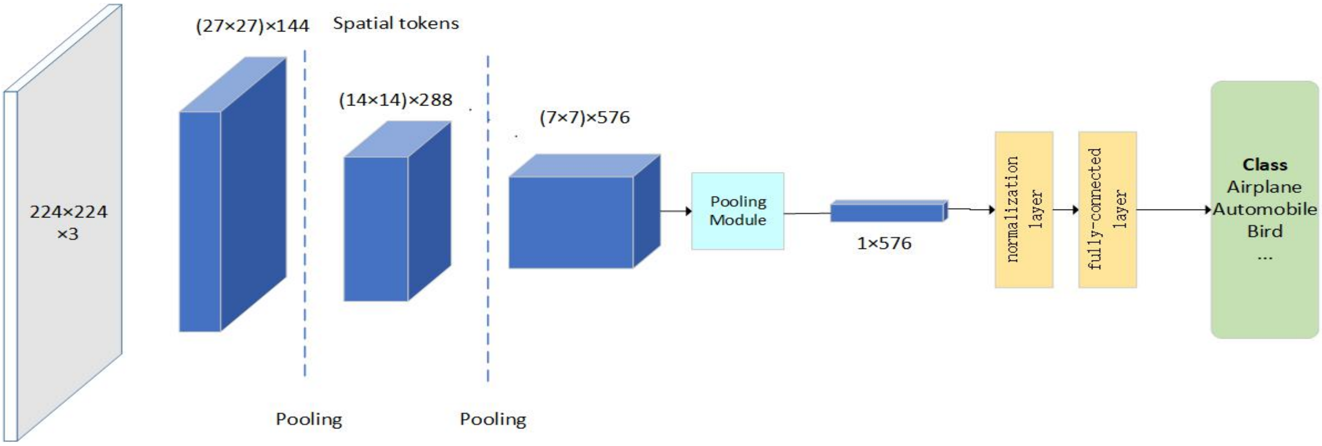


Figure 2 Use pooling modules instead of class token in the base structure of the PiT model

III. Pooling module

A. Model Structure

As in Figure 1, PiT is the same as ViT, and a class token is chosen to be added at the head of the sequence as an input for classification prediction, and the class token varies from the feature map size. As in Figure 2, to study the efficiency and effectiveness of the pooling module, we choose to use pooling instead of class token in PiT. Firstly, all the input images are resized to a size of 224 × 224. After three transformer layers a feature map X_L with size of 7 × 7 × 576 is obtained. The feature vectors are extracted by pooling for the feature map and passed into the layer normalization (LN) and the fully connected layer (FC) for linear transformation. The predicted classification value y can be calculated by the following equation.

$$y = softmax(FC(LN(PoolingModule(X_L)))) \qquad (1)$$

B. Pooling Modules

We use four different pooling methods instead of class token, which are. (1) AvgPool, (2) MaxPool, (3) Mixing Pool: AvgPool for the rows and MaxPool for the columns of the image, (4) Integration Pool: Getting the feature vectors by performing AvgPool and MaxPool on the image separately, and then using to concatenate the two feature vectors.

1) Traditional pooling method

To demonstrate the performance of the two traditional pooling modules, we use MaxPool and AvgPool to extract the feature vectors respectively, which can be expressed by the formula,

$$y = softmax(FC(LN(AvgPool(X_L)))) \qquad (2)$$

$$y = softmax(FC(LN(MaxPool(X_L)))) \quad (3)$$

2) Pooling method

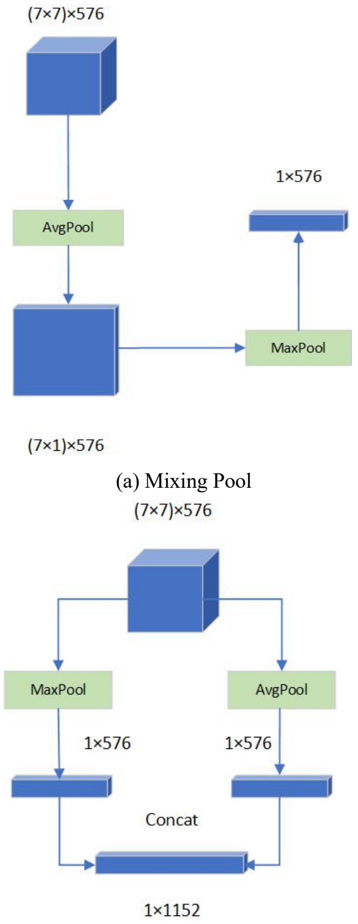
In general, AvgPool can reduce the error of increasing variance in the estimate due to restricted neighborhood size and retain more background information on the image, while MaxPool can reduce the error of shifting the estimate mean due to convolution parameter error and retain more texture information. We propose a new pooling method, called mixing pooling, which combines AvgPool and MaxPool methods. As in Figure 3(a), we use the method of mixing Pool, i.e., by performing AvgPool on the rows X_L of the output feature map X_W with shape (B, C, H, W), and then perform MaxPool on the columns X_H with shape (B, C, H, 1) to output the feature vector. It is expressed by the formula,

$$X_H = AvgPool(X_W) \quad (4)$$

$$y = softmax(FC(LN(MaxPool(X_H)))) \quad (5)$$

As in Figure 3(b), we obtain the feature vector by applying MaxPool and AvgPool to the output feature map X_L with shape (B, C, H, W), and then concatenate the two feature vectors. It can be expressed by the equation,

$$y = softmax(FC(LN(Concat(AvgPool(X_L), MaxPool(X_L))))) \quad (6)$$



(a) Mixing Pool

(b) Integration Pool

Figure 3 Mixing pool and Integration pool

IV. Experiments

Compared methods. To study the effectiveness of the pooling module, we take pooling module instead of the class token on two ViT-type models PiT_S [5] and PVT_ti [16].

Datasets and Evaluation metrics. We conduct experiments on the Cifar10 [2] dataset and measure the performance of different pooling methods based on Top-1 accuracy.

Implementation details. The paddlepaddle is used as our program framework, the experiments are carried out under the environment of GPU Tesla V100 and 4 core CPU i7. When using PiT_S, we train our models for 50 epochs with batch size of 16. The initial learning rate is 0.000375 and CosineDecay strategy [18] is used. We use AdamW optimizer and set the weight decay to 0.05. For the experiment on PVT_ti, the initial learning rate is 0.0001, and other hyperparameters are the same as PiT_S.

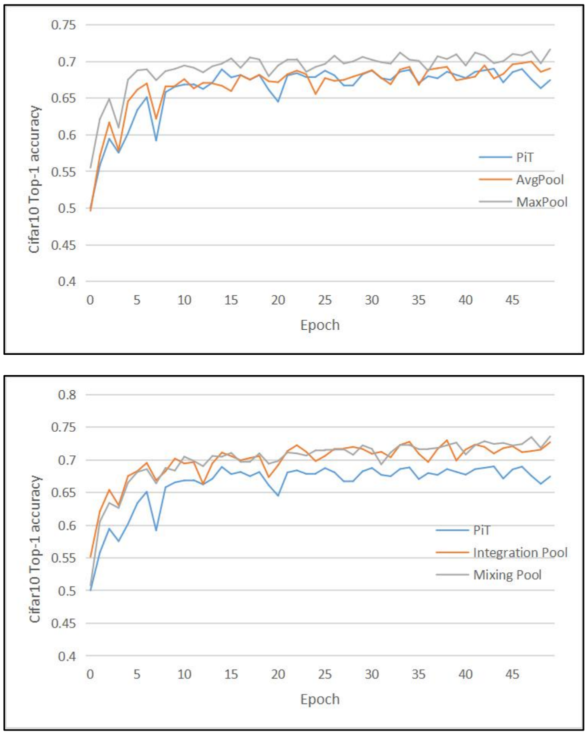
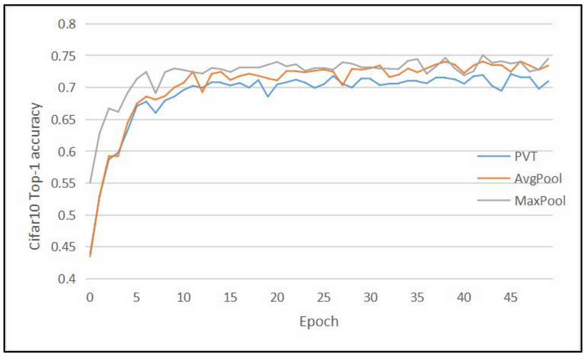


Figure 4 Top-1 accuracy curve of PiT on Cifar10 dataset using the pooling



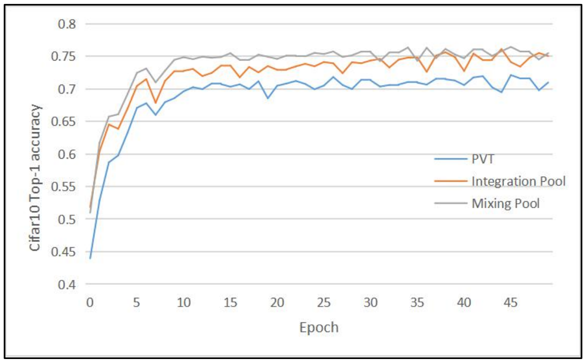


Figure 5 Top-1 accuracy curve of PVT on Cifar10 dataset using the pooling

Main Results. In Figure 4 and Figure 5, we plot the acc curves of different pooling modules with class token, and we find that after using pooling modules, both PiT and PVT models converge faster and more accurate than class token method. Everyone is in the smooth stage after 20 rounds. We report the numerical results from Tables 1 and 2. From Table 1, the performance of PiT increases by 0.98% and 2.64%, and PVT increases by 1.98% and 2.96%, respectively. Through the traditional pooling method, it proves that AvgPool and MaxPool can extract more classification features than class token. After using the integration pooling and mixing pooling method, the performance of PiT increases by 4% and 4.49%, and the performance of PVT increases by 3.99% and 4.32%, respectively. It is proved that the new pooling method can retain more image information and obtain better performance. Among them, mixing pooling achieved the highest performance.

TABLE I. BEST ACC FOR PIT ON CIFAR10 DATASET AFTER USING POOLING MODULE

	Model	Top-1 accuracy(%)
1	PiT	69.00%
2	PiT + AvgPool	69.98% (+0.98%)
3	PiT + MaxPool	71.64% (+2.64%)
4	PiT + Integration Pool	73.00% (+4.00%)
5	PiT + Mixing Pool	73.49% (+4.49%)

TABLE II. BEST ACC FOR PVT ON CIFAR10 DATASET AFTER USING POOLING MODULE

	Model	Top-1 accuracy(%)
1	PVT	72.11%
2	PVT + AvgPool	74.09% (+1.98%)
3	PVT + MaxPool	75.07% (+2.96%)
4	PVT + Integration Pool	76.10% (+3.99%)
5	PVT + Mixing Pool	76.43% (+4.32%)

V. CONCLUSION

In this paper, we propose a new pooling method called mixing pooling to replace the class token in the traditional transformer, and compare other different pooling methods. We experimentally demonstrate that mixing pooling not only saves computational cost of a simple and effective form, but also extracts more effective image classification features, which leads to a larger improvement in model performance. As a standalone module, mixing pooling can also be used to replace the class token in other vision transformer models.

ACKNOWLEDGMENT

This research was funded by the Science and Technology innovation 2025 major special project of Ningbo of China, grant number 2019B10036.

REFERENCES

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov,Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner,Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. Proc. Int. Conf.Learn. Representations, 2021.

[2] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. 2009.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In Proc. IEEE Conf. Comp. Vis. Patt. Recogn.,2009.

[4] Hugo Touvron, Matthieu Cord, Matthijs Douze, FranciscoMassa, Alexandre Sablayrolles, and Herv´e J´egou. Training data-efficient image transformers & distillation through attention. arXiv preprint arXiv:2012.12877, 2020.

[5] Byeongho Heo, Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Junsuk Choe, and Seong Joon Oh. Rethinking spatial dimensions of vision transformers. arXiv preprint arXiv:2103.16302, 2021.

[6] Stéphane d'Ascoli, Hugo Touvron, Matthew Leavitt, Ari Morcos, Giulio Biroli, and Levent Sagun. ConViT: Improving vision transformers with soft convolutional inductive biases. arXiv:2103.10697, 2021.

[7] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. arXiv preprint:2103.00112, 2021.

[8] Xiangxiang Chu, Zhi Tian, Bo Zhang, Xinlong Wang, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Conditional positional encodings for vision transformers. arXiv preprint arXiv:2102.10882, 2021.

[9] Zihang Jiang, Qibin Hou, Li Yuan, Daquan Zhou, Xiaojie Jin, Anran Wang, and Jiashi Feng. All tokens matter: Token labeling for training better vision transformers. arXiv preprint arXiv:2104.10858, 2021.

[10] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. arXiv preprint:2103.14030, 2021.

[11] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. arXiv preprint arXiv:2103.15808, 2021.

[12] Weijian Xu, Yifan Xu, Tyler Chang, and Zhuowen Tu. Coscale conv-attentional image transformers. arXiv preprint: 2104.06399, 2021.

[13] Ben Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Herv´e J´egou, and Matthijs Douze. Levit: a vision transformer in convnet's clothing for faster inference. arXiv preprint:2104.01136, 2021.

[14] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting the design of spatial attention in vision transformers. arXiv preprint arXiv:2104.13840, 2021

[15] Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. arXiv preprint arXiv:2103.14899, 2021.

[16] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. arXiv preprint arXiv:2102.12122, 2021.

[17] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvtv2: Improved baselines with pyramid vision transformer. arXiv preprint arXiv:2106.13797, 2021.

[18] I. Loshchilov, F. Hutter, Sgdr: Stochastic gradient descent with warm restarts, in: Proceedings of the International Conference on Learning Representations (ICLR), 2017.