Integrated crossing pooling of representation learning for Vision Transformer

Libo Xu*
NingboTech University, Ningbo, China
xlb@nbt.edu.cn

Xingsenli†
Guangdong University of Technology, Guangzhou, China
lixs@gdut.edu.cn

Zhenrui Huang
NingboTech University, Ningbo, China
huangzhenrui@nbt.edu.cn

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

Jiagong Wang
NingboTech University, Ningbo, China
3190439064@nbt.edu.cn

ABSTRACT

In recent years, transformer technology such as ViT, has been widely developed in the field of computer vision. In the ViT model, a learnable class token parameter is added to the head of the token sequence. The output of the class token through the whole transformer encoder is looked as the final representation vector, which is then passed through a multi-layer perception (MLP) network to get the classification prediction. The class token can be seen as an information aggregation of all other tokens. But we consider that the global pooling of tokens can aggregate information more effective and intuitive. In the paper, we propose a new pooling method, called cross pooling, to replace class token to obtain representation vector of the input image, which can extract better features and effectively improve model performance without increasing the computational cost. Through extensive experiments, we demonstrate that cross pooling methods achieve significant improvement over the original class token and existing global pooling methods such as average pooling or maximum pooling.

CCS CONCEPTS

• : **Artificial Intelligence, deep learning**;

KEYWORDS

vision transformer, ViT, Pooling method, class token

ACM Reference Format:
Libo Xu, Xingsenli, Zhenrui Huang, Yucheng Sun, and Jiagong Wang. 2021. Integrated crossing pooling of representation learning for Vision Transformer. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI-IAT '21 Companion), December 14–17, 2021, ESSENDON, VIC, Australia.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3498851.3499004

1 INTRODUCTION

In the last two years, Transformer [1] technology has been introduced to the field of computer vision and has achieved great success. The application of Transformer technology in image classification, target detection, semantic segmentation, etc. has become a research hotspot. ViT [2] is a Transformer-based image classification model proposed by Google in 2020, and has achieved SOTA results on the ImageNet dataset [3]. ViT divides the image into small patches and converts them into a sequence of tokens similar to that in NLP tasks, and adds a learnable class token parameter to the head of this sequence.The output of this class token through the whole transformer encoder is the representation vector of the input image, which is then passed through a MLP network to get the final classification result. The class token can be seen as an information aggregation of all other tokens. Since it is not based on the image content itself, it avoids the bias towards a particular token in the sequence. Many ViT-type models such as PiT and PVT, also use the class token to get the representation vector. However, making information aggregation for all tokens can also be done directly by global pooling methods, such as Swin Transformer using global average pooling to get the representation vector. We believe that it may be more efficient and faster to do information aggregation directly for the output of all tokens. It no longer needs class token parameters, and the process of collecting information is more straightforward.

In the paper, our contributions include the following three points: (1) We propose a new pooling method, called crosspooling, for vision transformer applications. This pooling method can extract token features better than the traditional average or maximum pooling, and can effectively improve the classification accuracy without increasing the computational cost. (2) Based on crosspooling, we propose integrated crosspooling methods, which can go further than the performance of simple crosspooling. (3) We synthetically compare different pooling methods through experiments and demonstrate the superiority of crosspooling method.

2 RELATED WORK

ViT was the first model to use Transformer technology to achieve the better performance in image classification with sufficient training data (e.g., ImageNet22k, JFT-300M). DeiT [4] further explores a efficient training strategy and a distillation approach for ViT. However, ViT does not change the image size during processing,

[*] Corresponding author
[†] Co-author

WI-IAT '21 Companion, December 14–17, 2021, ESSENDON, VIC, Australia
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-9187-0/21/12…$15.00
https://doi.org/10.1145/3498851.3499004

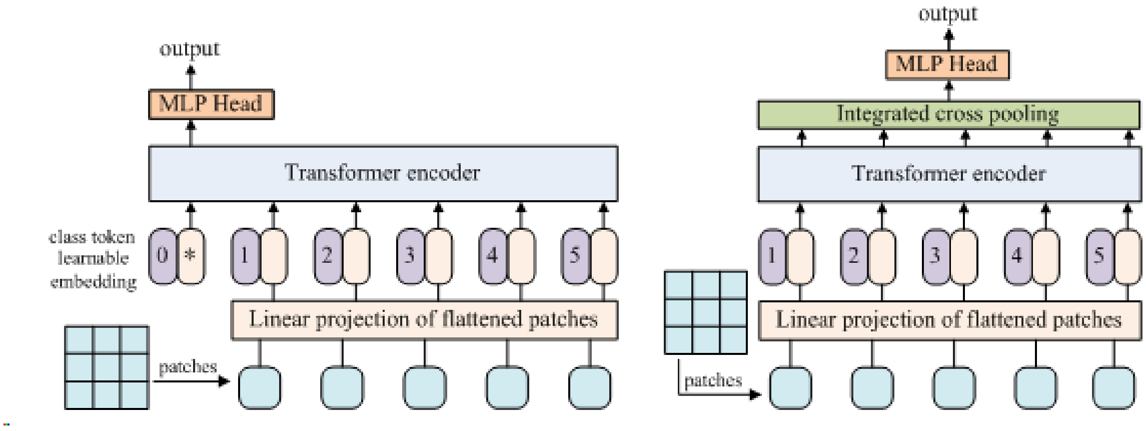


Figure 1: the final representation vectorof the ViT model

which leads to large amount of parameters and computational cost. Inspired by CNN backbone, PiT [5] proposes to add severalpooling layers to ViT. As the layers of the model go deeper, the number of channels of the feature map increases and the size of the feature map decreases. ConViT[6] introduces a new self-attention layer called gated positional self-attention into vision transformer. TNT[7] utilizes inner and outer Transformer blocks to generate pixel embeddings and patch embeddings respectively. CPVT[8] designed the convolution-based Positional Encoding Generator to perform positional encodings and proposed to replace class token with global average pooling. LV-ViT[9] provides a unique token label for each image patch by its dense score map, and defines a cross-entropy loss between token and its token label. Swin Transformer [10] replaces the fixed position embeddings by the relative position embeddings and restricts self-attention range within a certain window. Twins [11] combined local attention and global attention mechanisms to obtain better feature representation. PVT [12] outputs the feature maps from different layers by introducing a pyramid structure, and adds class token to the last layer to obtain the final representation information. PVTv2 [13] adopts zero-padding positional encoding into PVT as in CPVT, and performs global averaging pooling.

3 POOLING MODULE

3.1 Model Structure

Figure 1 shows the ViT-style structure and our revised part. In ViT-style model as Figure 1(a), the input image is converted to the vector sequencethrougha linear projection layer. Then, class token is added tothe sequence by concatenating operation. The class token is learnable. After transformer encoder layers, the output vector of the class token is used as a final representation vector. The vector is sent into a fully connected layer (FC) to obtain the classification prediction. Figure 1(b) shows our revised structure, which used cross pooling to instead of class token to obtain final representation vector. class token is no longer needed. We use cross pooling to gather all the output sequences through transformer layers. Therefore, the extraction of features is based on the output of all patches, and we believe that this will collect more effective feature information than class token did.

Figure 2shows our framework based on PiT. PiT is a ViT-style model, and a class token is chosen to be added at the head of the sequence as a final representation vector. For processing image features more efficiently, PiT use pooling method to progressively reduce the size of feature map, just as many convolutional networks did. Firstly, all the input images are resized to a size of $H\times W \times 3$. After three transformer layers,theinput image X should be converted to the feature map$X'$with size of $\frac{H}{32} \times \frac{W}{32} \times 576$. The feature vectors are extracted by cross pooling module and passed into the layer normalization (LN) layer and the fully connected layer (FC). The classification predicted value y can be obtained. The process canbe expressed as the following equation,

$$y = softmax(FC(LN(PoolingModule(X))))$$

3.2 Cross Pooling Modules

3.2.1 *Traditional pooling method.*

Global MaxPool and AvgPool are the common two traditional pooling methods. As shown in Figure 3(a), the AvgPool method converts a$H\times W$ feature map into a single value by calculating the average of the pixels of the whole map. The MaxPoolmethod do it by getting the max value of the pixels of the whole map, as shown in Figure 3(b). In general, AvgPool can reduce the variance of the pixels of the whole map and retain more background information, while MaxPool can retain more texture information. In the vision transformer model, the pooling operation converts the feature map with size of $H\times W\times 576$ from transformer into a vector with size of 1×576. Then the vector is sent into a FC layer and softmax operator to get classification probability. The processcan be expressed by the formula,

$$y = softmax(FC(LN(AvgPool(X))))$$
$$y = softmax(FC(LN(MaxPool(X))))$$

3.2.2 *Cross Pooling method.*

We propose a new pooling method, called cross pooling, which can better extract global feature information. Unlike the traditional global pooling such as MaxPool and AvgPool, cross pooling done pooling operation by row and

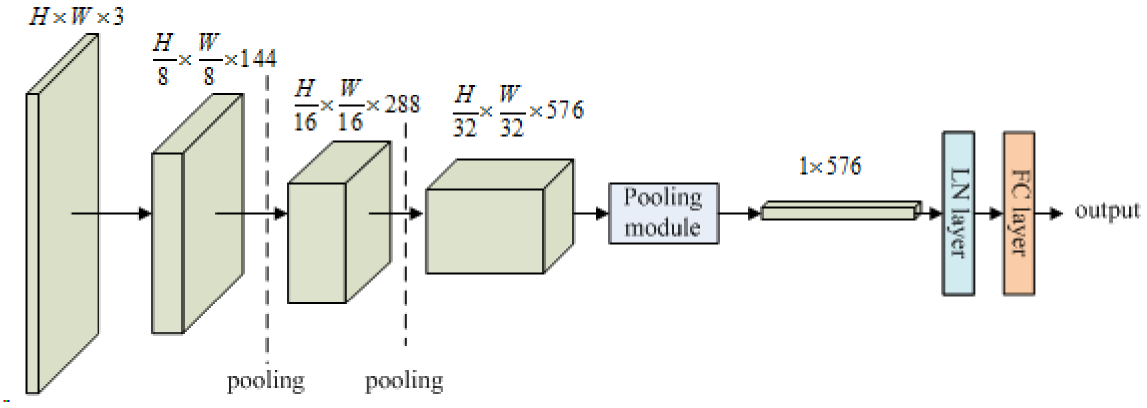

$H\times W\times 3$
$\frac{H}{8}\times\frac{W}{8}\times 144$
$\frac{H}{16}\times\frac{W}{16}\times 288$
$\frac{H}{32}\times\frac{W}{32}\times 576$
Pooling module
1×576
LN layer
FC layer
output
pooling
pooling

Figure 2: the framework based on the PiT model

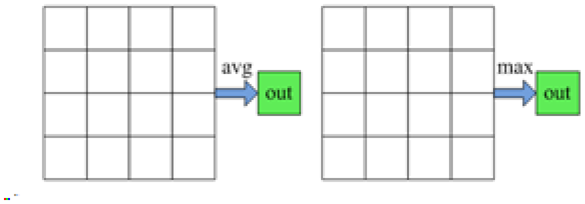


Figure 3: Traditional pooling method

column separately to get the final value.Figure 4shows four modes of cross pooling. As in Figure 4(a), for the input feature map$X$ $(C\times H\times W)$, we first take the average value on each row of X to obtain X_{ra} $(C\times H\times 1)$, then take the max value of X_{ra} to obtain the final value X'_{ra} $(C\times 1\times 1)$. Since the average operation is performed on the rows first, we call it RA_Cross Pooling. Figure 4(b) shows that the maximum operation is performed on the rows first, which called RM_Cross Pooling. In Figure 4(c), we first take the average value on each column of X to obtain X_{ca} $(C\times 1\times W)$, then take the max value of X_{ca} to obtain the final value X'_{ca} $(C\times 1\times 1)$. Since the average operation is performed on the columns first, we call it CA_Cross Pooling. Figure 4(d) shows that the maximum operation is performed on the columns first, which called CM_Cross Pooling. For RA_Cross Pooling, it is expressed by the formula,

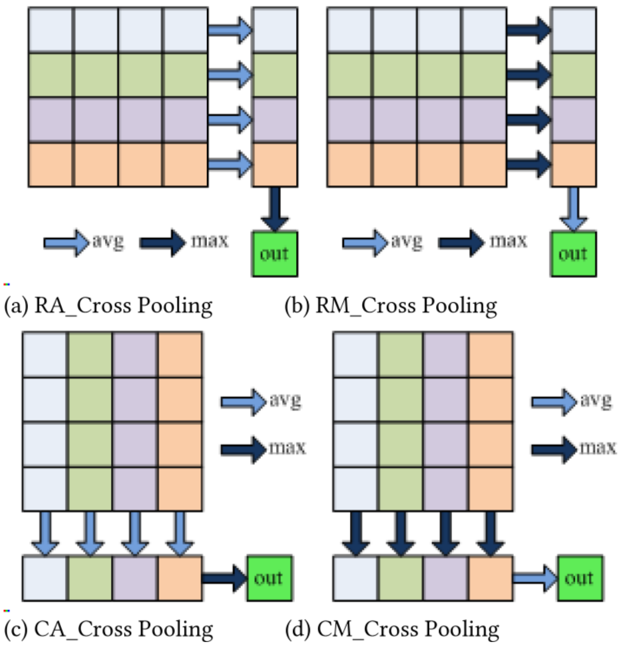
$$X'_{ra} = MaxPool(\underset{i\in \text{rowsof}X}{AvgPool}\ (X_i))$$

$$y = softmax(FC(LN(X'_{ra})))$$

3.2.3 Integrated Cross Pooling method. Figure 5showsthe process to obtain aIntegrated Cross Pooling methodcalled RARM_Integrated Cross Pooling. wefirst obtain the feature vector by applying RA_Cross Pooling and RM_Cross Pooling to the feature map X, and then concatenate them to get finalrepresentation vector. Then the vector is sent into a FC layer and softmax operator to achieve classification probability. It can be expressed by the equation,

$$X'_{ra\_rm} = concat(RA\_CrossPooling(X), RM\_CrossPooling(X))$$

$$y = softmax(FC(LN(X'_{ra\_rm})))$$



(a) RA_Cross Pooling (b) RM_Cross Pooling

(c) CA_Cross Pooling (d) CM_Cross Pooling

Figure 4: four modes of cross pooling

Following the same approach, we can construct other three Integrated Cross Pooling method: CACM_Integrated Cross Pooling, RACA_Integrated Cross Pooling, RMCM_Integrated Cross Pooling. Finally, we concatenate all four pooling outputs of Figure 4to get finalrepresentation vector, as shown in Figure 6. Due to every size of the output vector of simple cross pooling is 1× 576, the finalrepresentation vector is 1× 2304.

$$X'_{\text{total}} = concat(RA\_CrossPooling(X), RM\_CrossPooling(X), CA\_CrossPooling(X), CM\_CrossPooling(X))$$

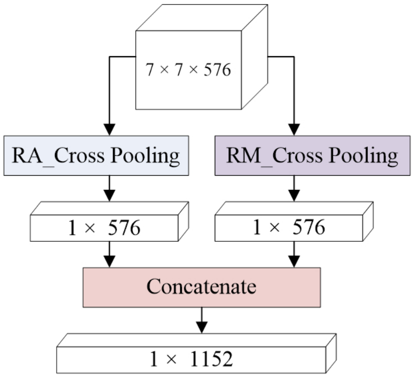


Figure 5: RARM_Integrated Cross Pooling

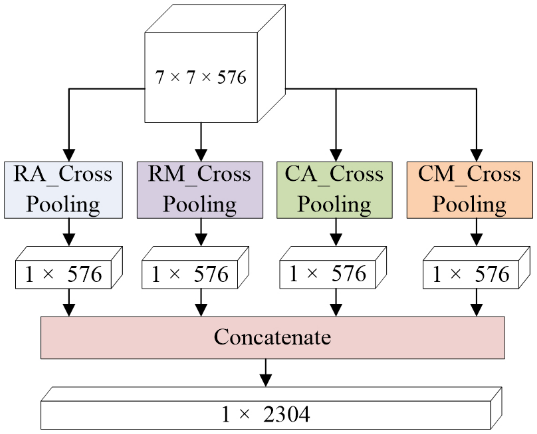


Figure 6: RACA_Integrated Cross Pooling

4 EXPERIMENTS

4.1 Datasets and Evaluation metrics

We insert cross pooling module into ViT-type models such as PiT_S and PVT_ti, and conduct experiments on the Cifar10 dataset. We measure the performance of different pooling methods of Top-1 and Top-5 accuracy.

4.2 Implementation details

The paddlepaddle2.0and python3.7 are used as program framework and program language, and the experiments are performed under the environment of GPU Tesla V100 and 4 core CPU i7. For the experiment on PiT_S, we train our models for 50 epochs with batch size of 16. The initial learning rate is 0.000375 and CosineDecay strategy [14] is used. We choose AdamW optimizer[15]with the weight decay being 0.05. For PVT_ti, the initial learning rate is 0.0001, and other hyperparameters are the same as PiT_S.

4.3 The experiment results for PiT

Table 1 shows the accuracy results of various pooling methods for PiT. The baseline model is the original PiT with class token. The results illustrate that all the pooling methods are better than baseline model. Compared to traditional global pooling methods such as MaxPool and AvgPool, all the cross pooling methods show their superiority. And, any simple cross pooling method can achieve a better accuracy than MaxPool ,AvgPool and the integrated pooling method by the two. When concatenating two simple cross pooling methods such as RARM_Integrated Cross Pooling, the better accuracy is obtained, which exceeds 1.3% over the integrated pooling method and exceeds 5.3% over the baseline. Finally, the integrated crosspooling which concatenated all the simple cross pooling methods can achieve the best accuracy that exceeds 6.0% over the baseline model.

Figure 7shows the top1 accuracy curve of PiT with various pooling methods. We compare the integrated crosspooling and all the simple cross pooling methods, the integrated crosspooling and the half-integrated crosspooling methods in Figure 7(a), (b) and (c), respectively. There are some fluctuations in the results of each method at different training epochs. However, overall, the results of the integrated crosspooling were significantly better than those of the other methods. The difference between the simple cross pooling methods is not significant.

4.4 The experiment results for PVT

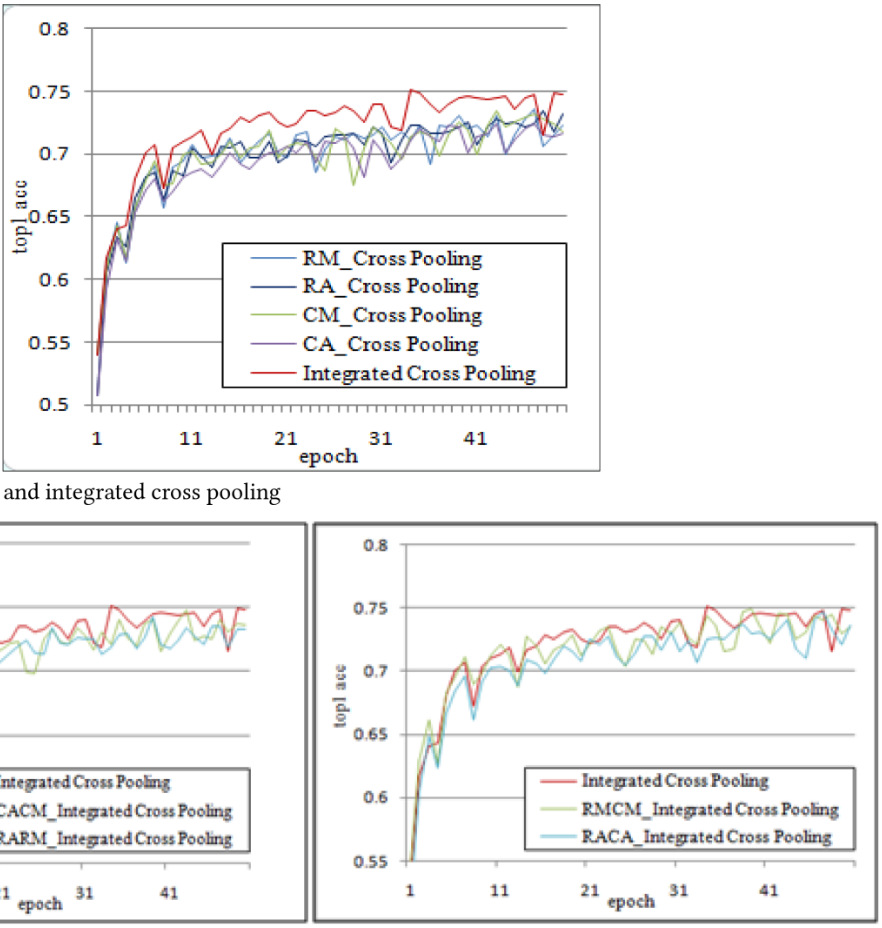
Table 2shows the accuracy results of various pooling methods for PVT. The baseline model is the original PVT with class token, which obtains 72.11% accuracy. Like as the experimentresults for PiT, all the pooling methods are significantly better than the baseline model. Compared to traditional global pooling methods such as MaxPool and AvgPool, all the cross pooling methods show their superiority. And, the accuracy of any simple cross pooling method is no less than MaxPool, AvgPool and the integrated pooling method by the two. When concatenating two simple cross pooling methods such as RARM_Integrated Cross Pooling, the better accuracy is obtained, which exceeds 1.5% over the integrated pooling method and exceeds 5.5% over the baseline. Finally, the integrated crosspooling which concatenated all the simple cross pooling methods can achieve the best accuracy that almost 6.0% over the baseline model.

5 CONCLUSION

In this paper, we propose a new pooling method, called cross pooling method, to replace the class token in the traditional vision transformer. In fact, the method contains a series of methods such as simple cross pooling methods and integrated cross pooling methods. Unlike the traditional global pooling, cross pooling methods done pooling operation by row and column separately to get the final value, which can better describe the key information of feature maps.We experimentally demonstrate that cross pooling not only reduce computational cost, but also extracts more effective image classification features, which leads to a effective improvement in classification performance. In fact, as a standalone module, cross pooling can not only be used to replace the class token in vision transformer models, but also be applied to in many convolutional neural networks.

Table 1: Accuracy of various pooling methods for PiT on Cifar10

Model	Top1 acc	Top5 acc
PiT	69.00%	–
PiT + AvgPool	69.98%	–
PiT + MaxPool	71.64%	–
PiT + Integration Pool	73.00%	–
PiT + RA_Cross Pooling	73.49%	97.77%
PiT + RM_Cross Pooling	73.57%	97.81%
PiT + CA_Cross Pooling	72.38%	97.80%
PiT + CM_Cross Pooling	73.47%	97.72%
PiT + RARM_Integrated Cross Pooling	74.30%	97.83%
PiT + CACM_Integrated Cross Pooling	74.81%	98.04%
PiT + RACA_Integrated Cross Pooling	74.70%	97.72%
PiT + RMCM_Integrated Cross Pooling	74.96%	97.99%
PiT + Integrated Cross Pooling	75.12%	98.14%



(a) Top-1 accuracy of simple and integrated cross pooling

(b) accuracy of half-integrated and integrated cross pooling

Figure 7: Top-1 accuracy of PiT using the pooling methods on Cifar10

Table 2: Accuracy of various pooling methods for PVT on Cifar10

Model	Top1 acc	Top5 acc
PVT	72.11%	–
PVT + AvgPool	74.09%	–
PVT + MaxPool	75.07%	–
PVT + Integration Pool	76.10%	–
PVT + RA_Cross Pooling	77.32%	98.37%
PVT + RM_Cross Pooling	75.85%	98.37%
PVT + CA_Cross Pooling	76.45%	98.06%
PVT + CM_Cross Pooling	77.09%	98.28%
PVT + RARM_Integrated Cross Pooling	77.62%	98.61%
PVT + CACM_Integrated Cross Pooling	77.65%	98.50%
PVT + RACA_Integrated Cross Pooling	77.48%	98.55%
PVT + RMCM_Integrated Cross Pooling	77.58%	98.35%
PVT + Integrated Cross Pooling	78.06%	98.58%

ACKNOWLEDGMENTS

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

REFERENCES

[1] Vaswani A , Shazeer N , Parmar N , *et al.* Attention Is All You Need. NIPS 2017, 2017.

[2] 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.

[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] 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

[12] 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.

[13] 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.

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

[15] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In ICLR, 2019.