Appendix for "Tokens-to-Token ViT: Training Vision Transformers from Scratch on ImageNet"

Li Yuan[1], Yunpeng Chen[2], Tao Wang[1,3], Weihao Yu[1], Yujun Shi[1],
Zihang Jiang[1], Francis E.H. Tay[1], Jiashi Feng[1], Shuicheng Yan[1]

[1] National University of Singapore [2] YITU Technology [3] Institute of Data Science, National University of Singapore

yuanli@u.nus.edu, yunpeng.chen@yitu-inc.com, shuicheng.yan@gmail.com

0.1. Details of transfer from CNN structure to ViT

We attempt to transfer the dense connection as DenseNets, wide or narrow channel dimensions as Wide-ResNet, channel attention as SE module, more heads as ResNeXt structure, and Ghost operation to ViT to validate the effects of CNN-based structure on ViT. ON the other hand, we also attempt to transfer these structure designs to our T2T-ViT. To simplify the designs, we only take ViT-S/16 and T2T-ViT-14 as examples and transfer the following designs strategies:

From ResNet-Wide to ViT&T2T-ViT Wide-ResNets are designed by decreasing layer depth and increasing width of ResNets, and such a design can improve model performance [11]. We thus design a ViT with deep-narrow backbone (ViT-DN) and Shallow-Wide backbone (ViT-SW), where ViT-DN has hidden dimensions 384 and 16 transformer layers and ViT-SW has hidden dimension 1024 and 4 layers.

From DenseNet to ViT&T2T-ViT Densely Connected Convolution Networks (DenseNets) [4] connect each convolutional layer with every other layer rather than only create short paths from early to later layer like ResNets, which can improve the information flow between layers in the network. As ViT adopts skip-connection as ResNets, a natural transfer is to apply the dense connection to ViT&T2T-ViT as ViT-Dense&T2T-ViT-Dense. Similar to DenseNet, if each block in ViT-Dense&T2T-ViT-Dense has L Transformer layers, there are $L(L+1)/2$ connections in this block and l-th layer has l input from the early layers. Specifically, we set the hidden dimension of the first layer in ViT-Dense&T2T-ViT-Dense as 128 and it increases 64 channels (''growth rate" as DenseNets) in each layer after concatenating with the early layers channels. The ViT-Dense&T2T-ViT-Dense has 4 blocks as [4,6,6,4] and transition layers can compress the channels after each block to improve model compactness. Such a design can make the ViT-Dense&T2T-ViT-Dense are deeper than ViT&T2T-ViT with a similar number of parameters and MACs.

From SENet to ViT&T2T-ViT Squeeze-an-Excitation (SE) Networks [3] apply the SE module in channel dimension, which can learn the inter-dependency between channels and bring improvement in performance on ResNets. The SE module is extremely simple and useful in CNN, so we transfer such modules to ViT&T2T-ViT. In ResNets, the SE module is applied after each bottleneck structure, thus we add the SE module in the channels after multi-head attention computation, and create ViT-SE&T2T-ViT-SE. The SE module in ViT&T2T-ViT can not only simply learn the inter-dependency between channels but also learn the local attention in the spatial dimension, as in the patch embedding, the spatial information in each patch will be squeezed to channel dimension.

From ResNeXt to ViT&T2T-ViT ResNeXt is constructed by splitting the channels with multiple paths and then concatenate a set of transformations on each split path, which is similar to the split-transform-merge strategy in Inception models [8]. In each split path, only 4 channels are transformed and then concatenated with other paths. Such a strategy is the same as the multi-heads attention design by splitting the channel dimensions into multiple heads. The size of the set of transformations in ResNeXt is exactly the number of heads, which is always 32 in ResNeXt. So for ViT&T2T-ViT, we can simply add the number of heads from 8 to 32 as ViT-ResNeXt&T2T-ViT-ResNeXt to validate the effects of such aggregated transformations in ViT and T2T-ViT.

From Ghost-CNN to ViT&T2T-ViT GhostNets [2] propose Ghost operation to generate more feature with cheap operations, which is a simple but effective method as the

Table 1. The hyper-parameters for all T2T-ViT models on ImageNet.

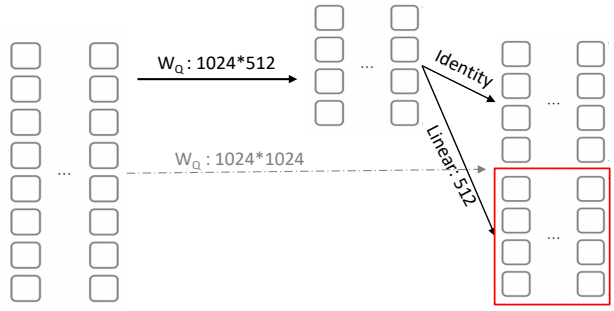
Models	T2T-ViT-7/12	T2T-ViT-14	T2T-ViT-19/24
Epochs	310	310	310
Warmup Epochs	5	5	5
Batch size	1024	512	512
Learning rate	1e-3	5e-4	5e-4
Weight decay	3e-2	5e-2	6.5e-2
Label smoothing	0.1	0.1	0.1
Dropout	0	0	0
Stoch.Depth	0.1	0.1	0.1
Mixup prob.	0.8	0.8	0.8
Cutmix prob.	1.0	1.0	1.0
Erasing prob.	0.25	0.25	0.25

feature maps in ResNets always has redundant channels. The ViT models have more redundant channels and invalid channels than ResNets. So we can transfer the ghost operations from CNN to ViT by applying such operations on both attention blocks and feed-forward blocks. As shown in Fig. 1, the ghost operation can be simply applied to ViT structure. Different with T2T-ViT-Dense and T2T-ViT-SE with comparable model size with T2T-ViT-14, the ghost operation can reduce the number of parameters and MACs of models, so the T2T-ViT-Ghost only has 80% parameters and of T2T-ViT-14.

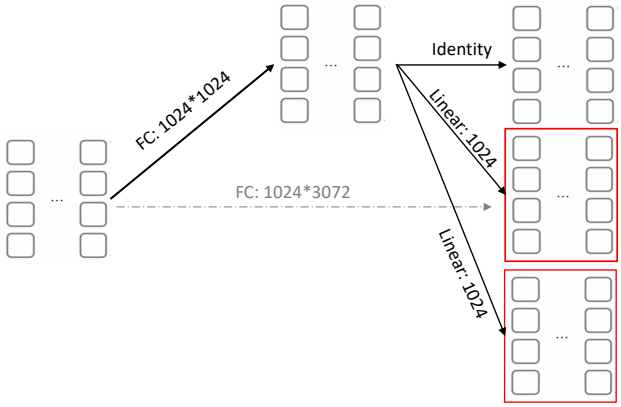
For fair comparisons, the above variants of T2T-ViT are designed with comparable size with T2T-ViT-14 and ResNet50 except for T2T-ViT-Ghost.

We give a simple reason why some CNN-based operation not work well in Transformer-based model. For Ghost, it would not improve the feature richness when using linear projected features in self-attention. For Dense, the gradient backpropagated through dense connections would perturb the self-attention dynamics per layer and thus hurt the model performance. For SE, one of the reasons is that self-attention is followed by MLP. Self-attention and MLP together dynamically reweigh and integrate multiple feature channels, and thus override the benefits from channel attention of SE.

It is noted that our design of each transferring is not the only choice, and we wish the transfers can motivate the model designs of Transformers in vision tasks.



(a) Ghost operation on attention block of ViT&T2T-ViT



(b) Ghost operation on feed-forward block of ViT&T2T-ViT

Figure 1. Ghost operation to reduce the hidden dimensions: (a) on the attention block (take the Query matrix W_Q as example). (b) on the feed-forward module. The dash line is the original operation and the solid lines are our ghost operation.

Table 2. Ablation study on training methods. We take T2T-ViT-14 as baseline model on ImageNet.

Ablation on ↓	Apply?	T2T-ViT-14
All Applied (Baseline)	Yes	81.5%
Mixup	No	81.2%
Cutmix	No	80.6%
Rand-augment	No	80.9%
Random erasing	No	81.0%
Label smoothing	No	81.3%
Stoch.Depth	No	81.2%
EMA	No	81.4%

0.2. Details of experimental setting

Our models and experiments are built and conducted upon PyTorch [5] and timm library [7], where we adopt some regularization and data-augmentation methods to train vision transformers to obtain reasonable results for ViT. Thoughout the experiments, we adopt AdamW as optimizer on ImageNet and SGD for CIFAR10 and CIFAR100 with cosine learning rate decay. In most of experiments, we set image size as 224×224 except for some special cases with 384×384 on ImageNet. In this section, we discuss the experimental setting adopted in this work.

Data augmentation and Regularization Without some inductive bias inherent to CNN, vision transformers require a large amount of data. In our experiments, we use randaugment [1] and random erasing [13] to enhance vision transformers. We find that the data augmentation is crucial to improve the transformers, as shown in Tab 2. The regularization methods we used in this work including Label Smoothing [6, 9], Mixup [12] and Cutmix [10]. We conduct ablation study on the augmentation and regularization methods, and the results are gieven in Tab. 2. We can find that without one of the augmentation or regularization methods, T2T-ViT-14 decrease around 0.1%-1.0% in accuracy for different methods.

Exponential Moving Average (EMA) EMA can improve the stability of training and we empirically find that it can improve the T2T-ViT model with 0-0.3% improvements (Tab. 2). In practical training, EMA test results are smaller than the normal testing at the beginning of training but can increase very fast after 10-20 epochs.

Hyper-parameters The hyper-parameters used in our experiments such as learning rate (lr), weight decay, batch size, Mixup and Cutmix are summarized in Tab. 1.

Transfer learning When fine-tuning our pretrained T2T-ViT from ImageNet to downstream datasets like CIFAR10 and CIFAR100, we adopt learning rate 5e-2 and weight decay 5e-4 by using SGD optimizer. We train T2T-ViT with 60 epochs with cosine learning rate decay, and the images of CIFAR10 and CIFAR100 are resized as 224×224 for finetuning.

Ablation study on the effects of patch size or overlapping size. In our work, we use patch size (7,3,3) and overlapping size (4,2,2) in the three T2T layers. We also evaluated the models with different patch sizes like (10,6,6) and (7,5,5), and different overlapping sizes (4,4,1) or (4,1,4). We found performance difference is quite small (around 0.2%), demonstrating our T2T module is robust to patch size and overlapping size choice. Our experiments provide two principles to design patch size and overlapping size in T2T: 1) it is better to use a smaller patch size than a larger one to save GPU memory, considering the performance gain is similar; 2). Progressive downsampling is better than aggressive downsampling as (4,2,2) is slightly better than (4,4,1) (0.2% in accuracy). So we adopt the current middle-size patch size and overlapping configurations.

References

[1] E. D. Cubuk, B. Zoph, J. Shlens, and Q. V. Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020.

[2] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu. Ghostnet: More features from cheap operations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1580–1589, 2020.

[3] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[4] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[5] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037, 2019.

[6] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[7] R. Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019.

[8] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[9] L. Yuan, F. E. Tay, G. Li, T. Wang, and J. Feng. Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3903–3911, 2020.

[10] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019.

[11] S. Zagoruyko and N. Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[12] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[13] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang. Random erasing data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13001–13008, 2020.