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

Abstract

Transformers, which are popular for language modeling, have been explored for solving vision tasks recently, e.g., *the Vision Transformer (ViT) for image classification. The ViT model splits each image into a sequence of tokens with fixed length and then applies multiple Transformer layers to model their global relation for classification. However, ViT achieves inferior performance to CNNs when trained from scratch on a midsize dataset like ImageNet. We find it is because: 1) the simple tokenization of input images fails to model the important local structure such as edges and lines among neighboring pixels, leading to low training sample efficiency; 2) the redundant attention backbone design of ViT leads to limited feature richness for fixed computation budgets and limited training samples. To overcome such limitations, we propose a new Tokens-To-Token Vision Transformer (T2T-ViT), which incorporates 1) a layer-wise Tokens-to-Token (T2T) transformation to progressively structurize the image to tokens by recursively aggregating neighboring Tokens into one Token (Tokens-to-Token), such that local structure represented by surrounding tokens can be modeled and tokens length can be reduced; 2) an efficient backbone with a deep-narrow structure for vision transformer motivated by CNN architecture design after empirical study. Notably, T2T-ViT reduces the parameter count and MACs of vanilla ViT by half, while achieving more than 3.0% improvement when trained from scratch on ImageNet. It also outperforms ResNets and achieves comparable performance with MobileNets by directly training on ImageNet. For example, T2T-ViT with comparable size to ResNet50 (21.5M parameters) can achieve 83.3% top1 accuracy in image resolution 384×384 on ImageNet.* [1]

1. Introduction

Self-attention models for language modeling like Transformers [37] have been recently applied to vision tasks, including image classification [5, 12, 43], object detection [3, 61] and image processing like denoising, super-resolution and deraining [4]. Among them, the Vision Transformer (ViT) [12] is the first full-transformer model that can be directly applied for image classification. In particular, ViT splits each image into 14×14 or 16×16 patches (*a.k.a.*, tokens) with fixed length; then following practice of the transformer for language modeling, ViT applies transformer layers to model the global relation among these tokens for classification.

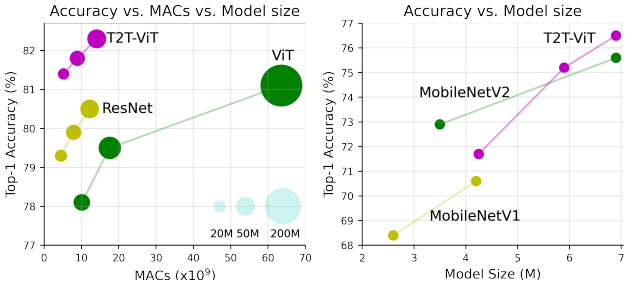

Figure 1. Comparison between T2T-ViT with ViT, ResNets and MobileNets when trained from scratch on ImageNet. Left: performance curve of MACs vs. top-1 accuracy. Right: performance curve of model size vs. top-1 accuracy.

Though ViT proves the full-transformer architecture is promising for vision tasks, its performance is still inferior to that of similar-sized CNN counterparts (*e.g.* ResNets) when trained from scratch on a midsize dataset (*e.g.*, ImageNet). We hypothesize that such performance gap roots in two main limitations of ViT: 1) the straightforward tokenization of input images by hard split makes ViT unable to model the image local structure like edges and lines, and thus it requires significantly more training samples (like JFT-300M for pretraining) than CNNs for achieving similar performance; 2) the attention backbone of ViT is not well-designed as CNNs for vision tasks, which contains redundancy and leads to limited feature richness and difficulties in model training.

To verify our hypotheses, we conduct a pilot study to investigate the difference in the learned features of ViT-L/16 [12] and ResNet50 [15] through visualization in Fig. 2. We observe the features of ResNet capture the desired local

*Work done during an internship at Yitu Tech.

[1] Code: https://github.com/yitu-opensource/T2T-ViT

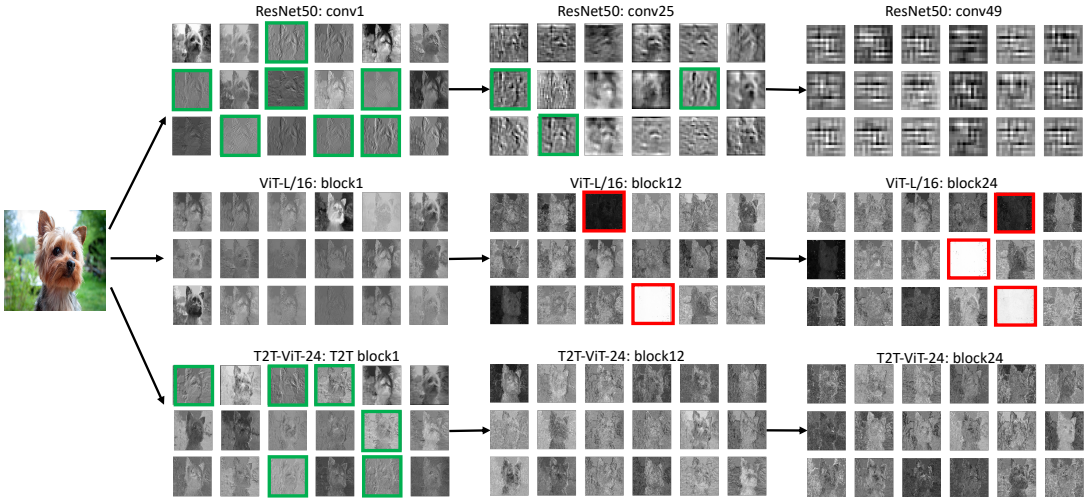

Figure 2. Feature visualization of ResNet50, ViT-L/16 [12] and our proposed T2T-ViT-24 trained on ImageNet. Green boxes highlight learned low-level structure features such as edges and lines; red boxes highlight invalid feature maps with zero or too large values. Note the feature maps visualized here for ViT and T2T-ViT are not attention maps, but image features reshaped from tokens. For better visualization, we scale the input image to size 1024×1024 or 2048×2048.

structure (edges, lines, textures, *etc*.) progressively from the bottom layer (conv1) to the middle layer (conv25). However, the features of ViT are quite different: the structure information is poorly modeled while the global relations (*e.g*., the whole dog) are captured by all the attention blocks. These observations indicate that the vanilla ViT ignores the local structure when directly splitting images to tokens with fixed length. Besides, we find many channels in ViT have zero value (highlighted in red in Fig. 2), implying the backbone of ViT is not efficient as ResNets and offers limited feature richness when training samples are not enough.

We are then motivated to design a new full-transformer vision model to overcome above limitations. 1) Instead of the naive tokenization used in ViT [12], we propose a progressive tokenization module to aggregate neighboring *Tokens* to one *Token* (named Tokens-to-Token module), which can model the local structure information of surrounding tokens and reduce the length of tokens iteratively. Specifically, in each Token-to-Token (T2T) step, the tokens output by a transformer layer are reconstructed as an image (*re-structurization*) which is then split into tokens with overlapping (*soft split*) and finally the surrounding tokens are aggregated together by flattening the split patches. Thus the local structure from surrounding patches is embedded into the tokens to be input into the next transformer layer. By conducting T2T iteratively, the local structure is aggregated into tokens and the length of tokens can be reduced by the aggregation process. 2) To find an efficient backbone for vision transformers, we explore borrowing some architecture designs from CNNs to build transformer layers for improving the feature richness, and we find "deep-narrow" architecture design with fewer channels but more layers in ViT brings much better performance at comparable model size and MACs (Multi-Adds). Specifically, we investigate Wide-ResNets (shallow-wide vs deep-narrow structure) [52], DenseNet (dense connection) [21], ResneXt structure [44], Ghost operation [14, 59] and channel attention [20]. We find among them, deep-narrow structure [52] is the most efficient and effective for ViT, reducing the parameter count and MACs significantly with nearly no degradation in performance. This also indicates the architecture engineering of CNNs can benefit the backbone design of vision transformers.

Based on the T2T module and deep-narrow backbone architecture, we develop the Tokens-to-Token Vision Transformer (T2T-ViT), which significantly boosts the performance when trained from scratch on ImageNet (Fig. 1), and is more lightweight than the vanilla ViT. As shown in Fig. 1, our T2T-ViT with 21.5M parameters and 4.8G MACs can achieve 81.5% top-1 accuracy on ImageNet, much higher than that of ViT [12] with 48.6M parameters and 10.1G MACs (78.1%). This result is also higher than the popular CNNs of similar size, like ResNet50 with 25.5M parameters (76%-79%). Besides, we also design lite variants of T2T-ViT by simply adopting fewer layers, which achieve comparable results with MobileNets [17, 32] (Fig. 1).

To sum up, our contributions are three-fold:

- For the first time, we show by carefully designing transformers architecture (T2T module and efficient backbone), visual transformers can outperform CNNs at different complexities on ImageNet without pre-training on JFT-300M.
- We develop a novel progressive tokenization for ViT and demonstrate its advantage over the simple tokenization approach by ViT, and we propose a T2T module that can encode the important local structure for each token.

- We show the architecture engineering of CNNs can benefit the backbone design of ViT to improve the feature richness and reduce redundancy. Through extensive experiments, we find deep-narrow architecture design works best for ViT.

2. Related Work

Transformers in Vision Transformers [37] are the models that entirely rely on the self-attention mechanism to draw global dependencies between input and output, and currently they have dominated natural language modelling [10, 30, 2, 46, 29, 23]. A transformer layer usually consists of a multi-head self-attention layer (MSA) and an MLP block. Layernorm (LN) is applied before each layer and residual connections in both the self-attention layer and MLP block. Recent works have explored applying transformers to various vision tasks: image classification [5, 12], object detection [3, 61, 58, 8, 34], segmentation [4, 40], image enhancement [4, 45], image generation [27], video processing [60, 53], and 3D point cloud processing [56]. Among them, the Vision Transformer (ViT) proves that a pure Transformer architecture can also attain state-of-the-art performance on image classification. However, ViT heavily relies on large-scale datasets such as ImageNet-21k and JFT-300M (which is not publically available) for model pretraining, requiring huge computation resources. In contrast, our proposed T2T-ViT is more efficient and can be trained on ImageNet without using those large-scale datasets. A recent concurrent work DeiT [36] applies Knowledge Distillation [16, 49] to improve the original ViT by adding a KD token along with the class token, which is orthogonal to our work, as our T2T-ViT focuses on the architecture design, and our T2T-ViT can achieve higher performance than DeiT without CNN as teacher model.

Self-attention in CNNs Self-attention mechanism has been widely applied to CNNs in vision task [38, 57, 19, 47, 20, 39, 1, 6, 18, 31, 42, 13, 50, 48]. Among these works, the SE block [20] applies attention to channel dimensions and non-local networks [39] are designed for capturing long-range dependencies via global attention. Compared with most of the works exploring global attention on images [1, 42, 13, 39], some works [18, 31] also explore self-attention in a local patch to reduce the memory and computation cost. More recently, SAN [55] investigates both pairwise and patchwise self-attention for image recognition, where the patchwise self-attention is a generalization of convolution. In this work, we also replace the T2T module with multiple convolution layers in experiments and find the convolution layers do not perform better than our designed T2T module.

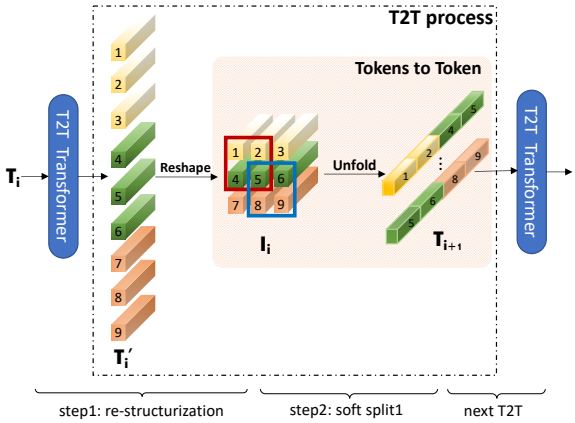

Figure 3. Illustration of T2T process. The tokens T_i are re-structurized as an image I_i after transformation and reshaping; then I_i is split with overlapping to tokens T_{i+1} again. Specifically, as shown in the pink panel, the four tokens (1,2,4,5) of the input I_i are concatenated to form one token in T_{i+1}. The T2T transformer can be a normal Transformer layer [37] or other efficient transformers like Performer layer [34] at limited GPU memory.

3. Tokens-to-Token ViT

To overcome the limitations of simple tokenization and inefficient backbone of ViT, we propose Tokens-to-Token Vision Transformer (T2T-ViT) which can progressively tokenize the image to tokens and has an efficient backbone. Hence, T2T-ViT consists of two main components (Fig. 4): 1) a layer-wise "Tokens-to-Token module" (T2T module) to model the local structure information of the image and reduce the length of tokens progressively; 2) an efficient "T2T-ViT backbone" to draw the global attention relation on tokens from the T2T module. We adopt a deep-narrow structure for the backbone to reduce redundancy and improve the feature richness after exploring several CNN-based architecture designs. We now explain these components one by one.

3.1. Tokens-to-Token: Progressive Tokenization

The Token-to-Token (T2T) module aims to overcome the limitation of simple tokenization in ViT. It progressively structurizes an image to tokens and models the local structure information, and in this way the length of tokens can be reduced iteratively. Each T2T process has two steps: *Re-structurization* and *Soft Split (SS)* (Fig. 3).

Re-structurization As shown in Fig. 3, given a sequence of tokens T from the preceding transformer layer, it will be transformed by the self-attention block (the T2T transformer in Fig. 3):

$$T' = \text{MLP}(\text{MSA}(T)), \tag{1}$$

where MSA denotes the multihead self-attention operation with layer normalization and "MLP" is the multilayer per-

ceptron with layer normalization in the standard Transformer [12]. Then the tokens T' will be reshaped as an image in the spatial dimension,

$$I = \text{Reshape}(T'). \tag{2}$$

Here "Reshape" re-organizes tokens $T' \in \mathbb{R}^{l \times c}$ to $I \in \mathbb{R}^{h \times w \times c}$, where l is the length of T', h, w, c are height, width and channel respectively, and $l = h \times w$.

Soft Split As shown in Fig. 3, after obtaining the re-structurized image I, we apply the soft split on it to model local structure information and reduce length of tokens. Specifically, to avoid information loss in generating tokens from the re-structurizated image, we split it into patches with overlapping. As such, each patch is correlated with surrounding patches to establish a prior that there should be stronger correlations between surrounding tokens. The tokens in each split patch are concatenated as one token (Tokens-to-Token, Fig. 3), and thus the local information can be aggregated from surrounding pixels and patches.

When conducting the soft split, the size of each patch is $k \times k$ with s overlapping and p padding on the image, where $k - s$ is similar to the stride in convolution operation. So for the reconstructed image $I \in \mathbb{R}^{h \times w \times c}$, the length of output tokens T_o after soft split is

$$l_o = \left\lfloor \frac{h + 2p - k}{k - s} + 1 \right\rfloor \times \left\lfloor \frac{w + 2p - k}{k - s} + 1 \right\rfloor. \tag{3}$$

Each split patch has size $k \times k \times c$. We flatten all patches in spatial dimensions to tokens $T_o \in \mathbb{R}^{l_o \times ck^2}$. After the soft split, the output tokens are fed for the next T2T process.

T2T module By conducting the above Re-structurization and Soft Split iteratively, the T2T module can progressively reduce the length of tokens and transform the spatial structure of the image. The iterative process in T2T module can be formulated as

$$\begin{aligned} T_i' &= \text{MLP}(\text{MSA}(T_i), \\ I_i &= \text{Reshape}(T_i'), \\ T_{i+1} &= \text{SS}(I_i), \qquad\qquad i = 1...(n-1). \end{aligned} \tag{4}$$

For the input image I_0, we apply a soft split at first to split it to tokens: $T_1 = \text{SS}(I_0)$. After the final iteration, the output tokens T_f of the T2T module has fixed length, so the backbone of T2T-ViT can model the global relation on T_f.

Additionally, as the length of tokens in the T2T module is larger than the normal case (16×16) in ViT, the MACs and memory usage are huge. To address the limitations, in our T2T module, we set the channel dimension of the T2T layer small (32 or 64) to reduce MACs, and optionally adopt an efficient Transformer such as Performer [7] layer to reduce memory usage at limited GPU memory. We provide an ablation study on the difference between adopting standard Transformer layer and Performer layer in our experiments.

3.2. T2T-ViT Backbone

As many channels in the backbone of vanilla ViT are invalid (Fig. 2), we plan to find an efficient backbone for our T2T-ViT to reduce the redundancy and improve the feature richness. Thus we explore different architecture designs for ViT and borrow some designs from CNNs to improve the backbone efficiency and enhance the richness of the learned features. As each transformer layer has skip connection as ResNets, a straightforward idea is to apply dense connection as DenseNet [21] to increase the connectivity and feature richness, or apply Wide-ResNets or ResNeXt structure to change the channel dimension and head number in the backbone of ViT. We explore five architecture designs from CNNs to ViT:

1. Dense connection as DenseNet [21];
2. Deep-narrow vs. shallow-wide structure as in Wide-ResNets [52];
3. Channel attention as Squeeze-an-Excitation (SE) Networks [20];
4. More split heads in multi-head attention layer as ResNeXt [44];
5. Ghost operations as GhostNet [14].

The details of these structure designs in ViT are given in the appendix. We conduct extensive experiments on the structures transferring in Sec. 4.2. We empirically find that 1) by adopting a deep-narrow structure that simply decreases channel dimensions to reduce the redundancy in channels and increase layer depth to improve feature richness in ViT, both the model size and MACs are decreased but performance is improved; 2) the channel attention as SE block also improves ViT but is less effective than using the deep-narrow structure.

Based on these findings, we design a deep-narrow architecture for our T2T-ViT backbone. Specifically, it has a small channel number and a hidden dimension d but more layers b. For tokens with fixed length T_f from the last layer of T2T module, we concatenate a class token to it and then add Sinusoidal Position Embedding (PE) to it, the same as ViT to do classification:

$$\begin{aligned} T_{f_0} &= [t_{cls}; T_f] + E, \qquad E \in \mathbb{R}^{(l+1) \times d} \\ T_{f_i} &= \text{MLP}(\text{MSA}(T_{f_i - 1})), \qquad i = 1...b \\ y &= \text{fc}(\text{LN}(T_{f_b})) \end{aligned} \tag{5}$$

where E is Sinusoidal Position Embedding, LN is layer normalization, fc is one fully-connected layer for classification and y is the output prediction.

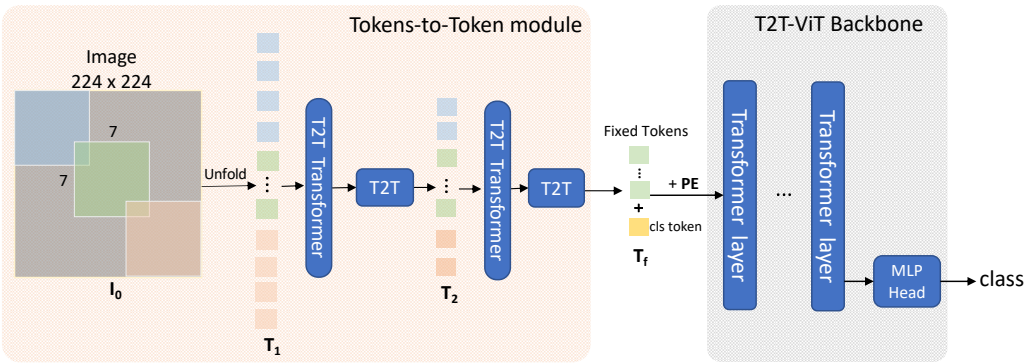

Figure 4. The overall network architecture of T2T-ViT. In the T2T module, the input image is first soft split as patches, and then unfolded as a sequence of tokens T_0. The length of tokens is reduced progressively in the T2T module (we use two iterations here and output T_f). Then the T2T-ViT backbone takes the fixed tokens as input and outputs the predictions. The two T2T blocks are the same as Fig. 3 and PE is Position Embedding.

Table 1. Structure details of T2T-ViT. T2T-ViT-14/19/24 have comparable model size with ResNet50/101/152. T2T-ViT-7/12 have comparable model size with MobileNetV1/V2. For T2T transformer layer, we adopt Transformer layer for T2T-ViT$_t$-14 and Performer layer for T2T-ViT-14 at limited GPU memory. For ViT, 'S' means Small, 'B' is Base and 'L' is Large. 'ViT-S/16' is a variant from original ViT-B/16 [12] with smaller MLP size and layer depth.

Models	Tokens-to-Token module				T2T-ViT backbone			Model size	
	T2T transformer	Depth	Hidden dim	MLP size	Depth	Hidden dim	MLP size	Params (M)	MACs (G)
ViT-S/16 [12]	-	-	-	-	8	786	2358	48.6	10.1
ViT-B/16 [12]	-	-	-	-	12	786	3072	86.8	17.6
ViT-L/16 [12]	-	-	-	-	24	1024	4096	304.3	63.6
T2T-ViT-14	Performer	2	64	64	14	384	1152	21.5	4.8
T2T-ViT-19	Performer	2	64	64	19	448	1344	39.2	8.5
T2T-ViT-24	Performer	2	64	64	24	512	1536	64.1	13.8
T2T-ViT$_t$-14	Transformer	2	64	64	14	384	1152	21.5	6.1
T2T-ViT-7	Performer	2	64	64	8	256	512	4.2	1.1
T2T-ViT-12	Performer	2	64	64	12	256	512	6.8	1.8

3.3. T2T-ViT Architecture

The T2T-ViT has two parts: the Tokens-to-Token (T2T) module and the T2T-ViT backbone (Fig. 4). There are various possible design choices for the T2T module. Here, we set $n = 2$ as shown in Fig. 4, which means there is $n+1 = 3$ soft split and $n = 2$ re-structurization in T2T module. The patch size for the three soft splits is $P = [7, 3, 3]$, and the overlapping is $S = [3, 1, 1]$, which reduces size of the input image from 224×224 to 14×14 according to Eqn. (3).

The T2T-ViT backbone takes tokens with fixed length from the T2T module as input, the same as ViT; but has a deep-narrow architecture design with smaller hidden dimensions (256-512) and MLP size (512-1536) than ViT. For example, T2T-ViT-14 has 14 transformer layers in T2T-ViT backbone with 384 hidden dimensions, while ViT-B/16 has 12 transformer layers and 768 hidden dimensions, which is 3x larger than T2T-ViT-14 in parameters and MACs.

To fairly compare with common hand-designed CNNs, we make T2T-ViT models have comparable size with ResNets and MobileNets. Specifically, we design three models: T2T-ViT-14, T2T-ViT-19 and T2T-ViT-24 of comparable parameters with ResNet50, ResNet101 and ResNet152 respectively. To compare with small models like MobileNets, we design two lite models: T2T-ViT-7, T2T-ViT-12 with comparable model size with MibileNetV1 and MibileNetV2. The two lite TiT-ViT have no special designs or tricks like efficient convolution [26] and simply reduce the layer depth, hidden dimension, and MLP ratio. The network details are summarized in Tab. 1.

4. Experiments

We conduct the following experiments with T2T-ViT for image classification on ImageNet. a) We validate the T2T-ViT by training from scratch on ImageNet and compare it with some common convolutional neural networks such as ResNets and MobileNets of comparable size; we also transfer the pretrained T2T-ViT to downstream datasets such as CIFAR10 and CIFAR100 (Sec. 4.1). (b) We compare

five T2T-ViT backbone architecture designs inspired from CNNs (Sec. 4.2). (c) We conduct ablation study to demonstrate effects of the T2T module and the deep-narrow architecture design of T2T-ViT (Sec. 4.3).

4.1. T2T-ViT on ImageNet

All experiments are conducted on ImageNet dataset [9], with around 1.3 million images in training set and 50k images in validation set. We use batch size 512 or 1024 with 8 NVIDIA GPUs for training. We adopt Pytorch [28] library and Pytorch image models library (timm) [41] to implement our models and conduct all experiments. For fair comparisons, we implement the same training scheme for the CNN models, ViT, and our T2T-ViT. Throughout the experiments on ImageNet, we set default image size as 224×224 except for some specific cases on 384×384, and adopt some common data augmentation methods such as mixup [54] and cutmix [11, 51] for both CNN and ViT&T2T-ViT model training, because ViT models need more training data to reach reasonable performance. We train these models for 310 epochs, using AdamW [25] as the optimizer and cosine learning rate decay [24]. The details of experiment setting are given in appendix. We also use both Transformer layer and Performer layer in T2T module for our models, resulting in T2T-ViT$_t$-14/19/24 (Transformer) and T2T-ViT-14/19/24 (Performer).

T2T-ViT vs. ViT We first compare performance of T2T-ViT and ViT on ImageNet. The results are given in Tab. 2. Our T2T-ViT is much smaller than ViT in number of parameters and MACs, yet giving higher performance. For example, the small ViT model ViT-S/16 with 48.6M and 10.1G MACs has 78.1% top-1 accuracy when trained from scratch on ImageNet, while our T2T-ViT$_t$-14 with only 44.2% parameters and 51.5% MACs achieves more than 3.0% improvement (81.5%). If we compare T2T-ViT$_t$-24 with ViT-L/16, the former reduces parameters and MACs around 500% but achieves more than 1.0% improvement on ImageNet. Comparing T2T-ViT-14 with DeiT-small and DeiT-small-Distilled, our T2T-ViT can achieve higher accuracy without large CNN models as teacher to enhance ViT. We also adopt higher image resolution as 384×384 and get 83.3% accuracy by our T2T-ViT-14↑384.

T2T-ViT vs. ResNet For fair comparisons, we set up three T2T-ViT models that have similar model size and MACs with ResNet50, ResNet101 and ResNet152. The experimental results are given in Tab. 3. The proposed T2T-ViT achieves 1.4%-2.7% performance gain over ResNets with similar model size and MACs. For example, compared with ResNet50 of 25.5M parameters and 4.3G MACs, our T2T-ViT-14 have 21.5M parameters and 4.8G MACs obtain 81.5% accuracy on ImageNet.

T2T-ViT vs. MobileNets The T2T-ViT-7 and T2T-ViT-12 have similar model size with MobileNetV1 [17] and MobileNetV2 [32], but achieve comparable or higher performance than MobileNets (Tab. 4). For example, Our T2T-ViT-12 with 6.9M parameters achieves 76.5% top1 accuracy, which is higher than MobileNetsV2$_{1.4x}$ by 0.9%. But we also note the MACs of our T2T-ViT are still larger than MobileNets because of the dense operations in Transformers. However, there are no special operations or tricks like efficient convolution [26, 32] in current T2T-ViT-7 and T2T-ViT-12, and we only reduce model size by reducing the hidden dimension, MLP ratio and depth of layers, indicating T2T-ViT is also very promising as a lite model. We also apply knowledge distillation on our T2T-ViT as the concurrent work DeiT [36] and find that our T2T-ViT-7 and T2T-ViT-12 can be further improved by distillation. Overall, the experimental results show, our T2T-ViT can achieve superior performance when it has mid-size as ResNets and reasonable results when it has a small model size as MobileNets.

Table 2. Comparison between T2T-ViT and ViT by training from scratch on ImageNet.

Models	Top1-Acc (%)	Params (M)	MACs (G)
ViT-S/16 [12]	78.1	48.6	10.1
DeiT-small [36]	79.9	22.1	4.6
DeiT-small-Distilled [36]	81.2	22.1	4.7
T2T-ViT-14	**81.5**	21.5	4.8
T2T-ViT-14↑384	**83.3**	21.5	17.1
ViT-B/16 [12]	79.8	86.4	17.6
ViT-L/16 [12]	81.1	304.3	63.6
T2T-ViT-24	**82.3**	**64.1**	**13.8**

Table 3. Comparison between our T2T-ViT with ResNets on ImageNet. T2T-ViT$_t$-14: using Transformer in T2T module. T2T-ViT-14: using Performer in T2T module. * means we train the model with our training scheme for fair comparisons.

Models	Top1-Acc (%)	Params (M)	MACs (G)
ResNet50 [15]	76.2	25.5	4.3
ResNet50*	79.1	25.5	4.3
T2T-ViT-14	**81.5**	21.5	4.8
T2T-ViT$_t$-14	**81.7**	21.5	6.1
ResNet101 [15]	77.4	44.6	7.9
ResNet101*	79.9	44.6	7.9
T2T-ViT-19	**81.9**	39.2	8.5
T2T-ViT$_t$-19	**82.2**	39.2	9.8
ResNet152 [15]	78.3	60.2	11.6
ResNet152*	80.8	60.2	11.6
T2T-ViT-24	**82.3**	64.1	13.8
T2T-ViT$_t$-24	**82.6**	64.1	15.0

Transfer learning We transfer our pretrained T2T-ViT to downstream datasets such as CIFAR10 and CIFAR100. We

Table 4. Comparison between our lite T2T-ViT with MobileNets. Models with '-Distilled' are taught by teacher model with the method as DeiT [36].

Models	Top1-Acc (%)	Params (M)	MACs (G)
MobileNetV1 1.0x*	70.8	4.2	0.6
T2T-ViT-7	**71.7**	4.3	1.1
T2T-ViT-7-Distilled	**73.1**	4.3	1.1
MobileNetV2 1.0x*	72.8	3.5	0.3
MobileNetV2 1.4x*	75.6	6.9	0.6
MobileNetV3 (Searched)	75.2	5.4	0.2
T2T-ViT-12	**76.5**	6.9	1.8
T2T-ViT-12-Distilled	**77.4**	6.9	1.9

Table 5. The results of fine-tuning the pretrained T2T-ViT to downstream datasets: CIFAR10 and CIFAR100.

Models	Params (M)	ImageNet	CIFAR10	CIFAR100
ViT/S-16	48.6	78.1	97.1	87.1
T2T-ViT-14	21.5	81.5	97.5	88.4
T2T-ViT-19	39.1	81.9	98.3	89.0

finetune the pretrained T2T-ViT-14/19 with 60 epochs by using SGD optimizer and cosine learning rate decay.The results are given in Tab. 5. We find that our T2T-ViT can achieve higher performance than the original ViT with smaller model sizes on the downstream datasets.

4.2. From CNN to ViT

To find an efficient backbone for vision transformers, we experimentally apply DenseNet structure, Wide-ResNet structure (wide or narrow channel dimensions), SE block (channel attention), ResNeXt structure (more heads in multihead attention), and Ghost operation from CNN to ViT. The details of these architecture designs are given in the appendix. From experimental results on "CNN to ViT" in Tab. 6, we can find both SE (ViT-SE) and Deep-Narrow structure (ViT-DN) benefit the ViT but the most effective structure is deep-narrow structure, which decreases model size and MACs nearly 2x and brings 0.9% improvement on the baseline model ViT-S/16.

We further apply these structures from CNN to our T2T-ViT, and conduct experiments on ImageNet under the same training scheme. We take ResNet50 as the baseline for CNN, ViT-S/16 for ViT, and T2T-ViT-14 for T2T-ViT. All experimental results are given in Tab. 6, and those on CNN and ViT&T2T-ViT are marked with the same colors. We summarize the effects of each CNN-based structure below.

Deep-narrow structure benefits ViT: The models ViT-DN (Deep-Narrow) and ViT-SW (Shallow-Wide) in Tab. 6 are two opposite designs in channel dimension and layer depth, where ViT-DN has 384 hidden dimensions and 16 layers and ViT-SW has 1,024 hidden dimensions and 4 layers. Compared with the baseline model ViT-S/16 with 768 hidden dimensions and 8 layers, shallow-wide model ViT-SW has 8.2% decrease in performance while ViT-DN with only half of model size and MACs achieve 0.9% increase. These results validate our hypothesis that vanilla ViT with shallow-wide structure is redundant in channel dimensions and limited feature richness with shallow layers.

Dense connection hurts performance of both ViT and T2T-ViT: Compared with the ResNet50, DenseNet201 has smaller parameters and comparable MACs, while it has higher performance. However, the dense connection can hurt performance of ViT-Dense and T2T-ViT-Dense (dark blue rows in Tab. 6).

SE block improves both ViT and T2T-ViT: From red rows in Tab. 6, we can find SENets, ViT-SE and T2T-ViT-SE are higher than the corresponding baseline. The SE module can improve performance on both CNN and ViT, which means applying attention to channels benefits both CNN and ViT models.

ResNeXt structure has few effects on ViT and T2T-ViT: ResNeXts adopt multi-head on ResNets, while Transformers are also multi-head attention structure. When we adopt more heads like 32, we can find it has few effects on performance (red rows in Tab 6). However, adopting a large number of heads makes the GPU memory large, which is thus unnecessary in ViT and T2T-ViT.

Ghost can further compress model and reduce MACs of T2T-ViT: Comparing experimental results of Ghost operation (magenta row in Tab. 6), the accuracy decreases 2.9% on ResNet50, 2.0% on T2T-ViT, and 4.4% on ViT. So the Ghost operation can further reduce the parameters and MACs of T2T-ViT with smaller performance degradation than ResNet. But for the original ViT, it would cause more decrease than ResNet.

Besides, for all five structures, the T2T-ViT performs better than ViT, which further validates the superiority of our proposed T2T-ViT. And we also wish this study of transferring CNN structure to ViT can motivate the network design of Transformers in vision tasks.

4.3. Ablation study

To further identify effects of T2T module and deep-narrow structure, we do ablation study on our T2T-ViT.

T2T module To verify the effects of the proposed T2T module, we experimentally compare three different models: T2T-ViT-14, T2T-ViT-14$_{wo\ T2T}$, and T2T-ViT$_t$-14, where T2T-ViT-14$_{wo\ T2T}$ has the same T2T-ViT backbone but without T2T module. We can find with similar model size and MACs, the T2T module can improve model performance by 2.0%-2.2% on ImageNet.

As the soft split in T2T module is similar to convolution operation without convolution filters, we also replace the T2T module by 3 convolution layers with kernel size (7,3,3), stride size (4,2,2) respectively. Such a model with

Table 6. Transfer of some common designs in CNN to ViT&T2T-ViT, including DenseNet, Wide-ResNet, SE module, ResNeXt, Ghost operation. The same color means the correspond transfer. All models are trained from scratch on ImageNet. * means we reproduce the model with our training scheme for fair comparisons.

Model Type	Models	Top1-Acc (%)	Params (M)	MACs (G)	Depth	Hidden_dim
Traditional CNN	AlexNet [22]	56.6	61.1	0.77	-	-
	VGG11 [33]	69.1	132.8	7.7	11	-
	Inception v3 [35]	77.4	27.2	5.7	-	-
Skip-connection CNN	ResNet50 [15]	76.2	25.6	4.3	50	-
	ResNet50* (Baseline)	79.1	25.6	4.3	50	-
	Wide-ResNet18x1.5*	78.0 (-1.1)	26.0	4.1	18	-
	DenseNet201*	77.5 (-1.6)	20.1	4.4	201	-
	SENet50*	80.3 (+1.2)	28.1	4.9	50	-
	ResNeXt50*	79.9 (+0.8)	25.0	4.3	50	-
	ResNet50-Ghost*	76.2 (-2.9)	19.9	3.2	50	-
CNN to ViT	ViT-S/16 (Baseline)	78.1	48.6	10.1	8	768
	ViT-DN	79.0 (+0.9)	24.5	5.5	16	384
	ViT-SW	69.9 (-8.2)	47.9	9.9	4	1024
	ViT-Dense	76.8 (-1.3)	46.7	9.7	19	128-736
	ViT-SE	**78.4** (+0.3)	49.2	10.2	8	768
	ViT-ResNeXt	78.0 (-0.1)	48.6	10.1	8	768
	ViT-Ghost	73.7 (-4.4)	32.1	6.9	8	768
CNN to T2T-ViT	T2T-ViT-14 (Baseline)	81.5	21.5	4.8	14	384
	T2T-ViT-Wide	77.9 (-3.4)	25.1	5.0	14	768
	T2T-ViT-Dense	80.6 (-1.1)	23.7	5.5	19	128-584
	T2T-ViT-SE	**81.6** (+0.1)	21.9	4.9	14	384
	T2T-ViT-ResNeXt	81.5 (+0.0)	21.5	4.8	14	384
	T2T-ViT-Ghost	79.5 (-2.0)	16.3	3.7	14	384

Table 7. Ablation study results on T2T module, Deep-Narrow(DN) structure.

Ablation type	Models	Top1-Acc (%)	Params (M)	MACs (G)
T2T module	T2T-ViT-14$_{wo\ T2T}$	79.5	21.1	4.2
	T2T-ViT-14	81.5 (+2.0)	21.5	4.8
	T2T-ViT$_t$-14	81.7 (+2.2)	21.5	6.1
	T2T-ViT$_c$-14	80.8 (+1.3)	21.3	4.6
DN Structure	T2T-ViT-14	81.5	21.5	4.8
	T2T-ViT-d768-4	78.8 (-2.7)	25.0	5.4

convolution layers to build T2T module is denoted as T2T-ViT$_c$-14. From Tab. 7, we can find the T2T-ViT$_c$-14 is worse than T2T-ViT-14 and T2T-ViT$_t$-14 by 0.5%-1.0% on ImageNet. We also note that the T2T-ViT$_c$-14 is still higher than T2T-ViT-14$_{wo\ T2T}$, as the convolution layers in the early stage can also model the structure information. But our designed T2T module is better than the convolution layers as it can model both the global relation and the structure information of the images.

Deep-narrow structure We use the deep-narrow structure with fewer hidden dimensions but more layers, rather than the shallow-wide one in the original ViT. We compare the T2T-ViT-14 and T2T-ViT-d768-4 to verify its effects. T2T-ViT-d768-4 is a shallow-wide structure with hidden dimension of 768 and 4 layers, with similar model size and MACs as T2T-ViT-14. From Tab. 7, we can find after changing our deep-narrow to shallow-wide structure, the T2T-ViT-d768-4 has 2.7% decrease in top-1 accuracy, validating deep-narrow structure is crucial for T2T-ViT.

5. Conclusion

In this work, we propose a new T2T-ViT model that can be trained from scratch on ImageNet and achieve comparable or even better performance than CNNs. T2T-ViT effectively models the structure information of images and enhances feature richness, overcoming limitations of ViT. It introduces the novel tokens-to-token (T2T) process to progressively tokenize images to tokens and structurally aggregate tokens. We also explore various architecture design choices from CNNs for improving T2T-ViT performance, and empirically find the deep-narrow architecture performs better than the shallow-wide structure. Our T2T-ViT achieves superior performance to ResNets and comparable performance to MobileNets with similar model size when trained from scratch on ImageNet. It paves the way for further developing transformer-based models for vision tasks.

References

[1] I. Bello, B. Zoph, A. Vaswani, J. Shlens, and Q. V. Le. Attention augmented convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3286–3295, 2019.

[2] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[3] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. *arXiv preprint arXiv:2005.12872*, 2020.

[4] 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. *arXiv preprint arXiv:2012.00364*, 2020.

[5] M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, and I. Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, pages 1691–1703. PMLR, 2020.

[6] Y. Chen, Y. Kalantidis, J. Li, S. Yan, and J. Feng. Aˆ 2-nets: Double attention networks. In *Advances in neural information processing systems*, pages 352–361, 2018.

[7] K. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, P. Hawkins, J. Davis, A. Mohiuddin, L. Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

[8] Z. Dai, B. Cai, Y. Lin, and J. Chen. Up-detr: Unsupervised pre-training for object detection with transformers. *arXiv preprint arXiv:2011.09094*, 2020.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[11] T. DeVries and G. W. Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[13] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019.

[14] 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.

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[16] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[17] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[18] H. Hu, Z. Zhang, Z. Xie, and S. Lin. Local relation networks for image recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3464–3473, 2019.

[19] J. Hu, L. Shen, S. Albanie, G. Sun, and A. Vedaldi. Gather-excite: Exploiting feature context in convolutional neural networks. *Advances in neural information processing systems*, 31:9401–9411, 2018.

[20] 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.

[21] 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.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[23] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[24] I. Loshchilov and F. Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[25] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[26] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[27] N. Parmar, A. Vaswani, J. Uszkoreit, Ł. Kaiser, N. Shazeer, A. Ku, and D. Tran. Image transformer. *arXiv preprint arXiv:1802.05751*, 2018.

[28] 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.

[29] M. E. Peters, M. Neumann, R. L. Logan IV, R. Schwartz, V. Joshi, S. Singh, and N. A. Smith. Knowledge enhanced contextual word representations. *arXiv preprint arXiv:1909.04164*, 2019.

[30] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training, 2018.

[31] P. Ramachandran, N. Parmar, A. Vaswani, I. Bello, A. Levskaya, and J. Shlens. Stand-alone self-attention in vision models. *arXiv preprint arXiv:1906.05909*, 2019.

[32] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[33] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[34] Z. Sun, S. Cao, Y. Yang, and K. Kitani. Rethinking transformer-based set prediction for object detection. *arXiv preprint arXiv:2011.10881*, 2020.

[35] 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.

[36] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou. Training data-efficient image transformers & distillation through attention. *arXiv preprint arXiv:2012.12877*, 2020.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30:5998–6008, 2017.

[38] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang. Residual attention network for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2017.

[39] X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.

[40] Y. Wang, Z. Xu, X. Wang, C. Shen, B. Cheng, H. Shen, and H. Xia. End-to-end video instance segmentation with transformers. *arXiv preprint arXiv:2011.14503*, 2020.

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

[42] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[43] B. Wu, C. Xu, X. Dai, A. Wan, P. Zhang, M. Tomizuka, K. Keutzer, and P. Vajda. Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*, 2020.

[44] 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.

[45] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo. Learning texture transformer network for image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5791–5800, 2020.

[46] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5753–5763, 2019.

[47] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola. Stacked attention networks for image question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 21–29, 2016.

[48] L. Yuan, S. Chang, Z. Huang, Y. Zhou, Y. Chen, X. Nie, F. E. Tay, J. Feng, and S. Yan. A simple baseline for pose tracking in videos of crowed scenes. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4684–4688, 2020.

[49] 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.

[50] L. Yuan, Y. Zhou, S. Chang, Z. Huang, Y. Chen, X. Nie, T. Wang, J. Feng, and S. Yan. Toward accurate person-level action recognition in videos of crowed scenes. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4694–4698, 2020.

[51] 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.

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

[53] Y. Zeng, J. Fu, and H. Chao. Learning joint spatial-temporal transformations for video inpainting. In *European Conference on Computer Vision*, pages 528–543. Springer, 2020.

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

[55] H. Zhao, J. Jia, and V. Koltun. Exploring self-attention for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10076–10085, 2020.

[56] H. Zhao, L. Jiang, J. Jia, P. Torr, and V. Koltun. Point transformer. *arXiv preprint arXiv:2012.09164*, 2020.

[57] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. Change Loy, D. Lin, and J. Jia. Psanet: Point-wise spatial attention network for scene parsing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 267–283, 2018.

[58] M. Zheng, P. Gao, X. Wang, H. Li, and H. Dong. End-to-end object detection with adaptive clustering transformer. *arXiv preprint arXiv:2011.09315*, 2020.

[59] D. Zhou, X. Jin, Q. Hou, K. Wang, J. Yang, and J. Feng. Neural epitome search for architecture-agnostic network compression. In *International Conference on Learning Representations*, 2019.

[60] L. Zhou, Y. Zhou, J. J. Corso, R. Socher, and C. Xiong. End-to-end dense video captioning with masked transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8739–8748, 2018.

[61] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.