

Lite Vision Transformer with Enhanced Self-Attention

Chenglin Yang^{1*}; Yilin Wang²(✉), Jianming Zhang², He Zhang², Zijun Wei², Zhe Lin², Alan Yuille¹
¹Johns Hopkins University ²Adobe Inc.

{chenglin.yangw, alan.l.yuille}@gmail.com {yilwang, jianmzha, hezhan, zwei, zlin}@adobe.com

Abstract

Despite the impressive representation capacity of vision transformer models, current light-weight vision transformer models still suffer from inconsistent and incorrect dense predictions at local regions. We suspect that the power of their self-attention mechanism is limited in shallower and thinner networks. We propose Lite Vision Transformer (LVT), a novel light-weight transformer network with two enhanced self-attention mechanisms to improve the model performances for mobile deployment. For the low-level features, we introduce Convolutional Self-Attention (CSA). Unlike previous approaches of merging convolution and self-attention, CSA introduces local self-attention into the convolution within a kernel of size 3×3 to enrich low-level features in the first stage of LVT. For the high-level features, we propose Recursive Atrous Self-Attention (RASA), which utilizes the multi-scale context when calculating the similarity map and a recursive mechanism to increase the representation capability with marginal extra parameter cost. The superiority of LVT is demonstrated on ImageNet recognition, ADE20K semantic segmentation, and COCO panoptic segmentation. The code is made publicly available¹.

1. Introduction

Transformer-based architectures have achieved remarkable success most recently, they demonstrated superior performances on a variety of vision tasks, including visual recognition [65], object detection [38, 56], semantic segmentation [10, 60] and etc [32, 54, 55].

Inspired by the success of the self-attention module in the Natural Language Processing (NLP) community [53], Dosovitskiy [18] first propose a transformer-based network for computer vision, where the key idea is to split the image into patches so that it can be linearly embedded with positional embedding. To reduce the computational complexity introduced by [18], Swin-Transformer [38] upgrades

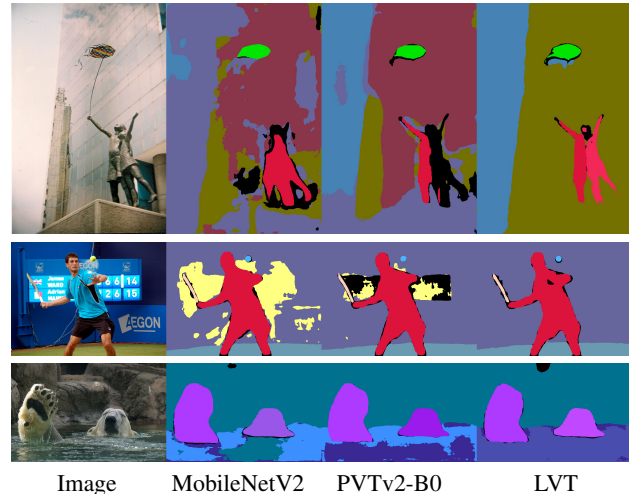


Figure 1. Mobile COCO panoptic segmentation. The model needs to recognize, localize, and segment both objects and stuffs at the same time. All the methods have less than 5.5M parameters with same training and testing process under panoptic FPN framework. The only difference is the choice of encoder architecture. Lite Vision Transformer (LVT) leads to the best results with significant improvement over the accuracy and coherency of the labels.

the architecture by limiting the computational cost of self-attention with local non-overlapping windows. Additionally, the hierarchical feature representations are introduced to leverage features from different scales for better representation capability. On the other hand, PVT [56, 57] proposes spatial-reduction attention (SRA) to reduce the computational cost. It also removes the positional embedding by inserting depth-wise convolution into the feed forward network (FFN) that follows the self-attention layer. Both Swin-Transformer and PVT have demonstrated their effectiveness for downstream vision tasks. However, when scaling down the model to a mobile friendly size, there is also a significant performance degradation.

In this work, we focus on designing a light yet effective vision transformer for mobile applications [47]. More specifically, we introduce a Lite Vision Transformer (LVT) backbone with two novel self-attention layers to pursue both

*Work done while an intern at Adobe.

¹<https://github.com/Chenglin-Yang/LVT>

the performance and compactness. LVT follows a standard four-stage structure [21, 38, 57] but has similar parameter size to existing mobile networks such as MobileNetV2 [47] and PVTv2-B0 [56].

Our first improvement of self-attention is named Convolutional Self-Attention (CSA). The self-attention layers [1, 24, 44, 58, 66] are the essential components in vision transformer, as self-attention captures both short- and long-range visual dependencies. However, identifying the locality is another significant key to success in vision tasks. For example, the convolution layer is a better layer to process low-level features [16]. Prior arts have been proposed to combine convolution and self-attention with the global receptive field [16, 59]. Instead, we introduce the local self-attention into the convolution within the kernel of size 3×3 . CSA is proposed and used in the first stage of LVT. As a result of CSA, LVT has better generalization ability as it enriches the low-level features over existing transformer models. As shown in Fig 1, compared to PVTv2-B0 [56], LVT is able to generate more coherent labels in local regions.

On the other hand, the performances of lite models are still limited by the parameter number and model depth [60]. We further propose to increase the representation capacity of lite transformers by Recursive Atrous Self-Attention (RASA) layers. As shown in Fig 1, LVT results have better semantic correctness due to such effective representations. Specifically, RASA incorporates two components with weight sharing mechanisms. The first one is Atrous Self-Attention (ASA). It utilizes the multi-scale context with a single kernel when calculating the similarities between the query and key. The second one is the recursion pipeline. Following standard recursive network [19, 28], we formalize RASA as a recursive module with ASA as the activation function. It increases the network depth without introducing additional parameters.

Experiments are performed on ImageNet [46] classification, ADE20K [69] semantic segmentation and COCO [36] panoptic segmentation to evaluate the performance of LVT as a generalized vision model backbone. Our main contributions are summarized in the following:

- We propose Convolutional Self-Attention (CSA). Unlike previous methods of merging global self-attention with convolution, CSA integrates local self-attention into the convolution kernel of size 3×3 . It is proposed to process low-level features by including both dynamic kernels and learnable filters.
- We propose Recursive Atrous Self-Attention (RASA). It comprises two parts. The first part is Atrous Self-Attention (ASA) that captures the multi-scale context in the calculation of similarity map in self-attention. The other part is the recursive formulation with ASA as the activation function. RASA is proposed to increase the representation ability with marginal extra

parameter cost.

- We propose Lite Vision Transformer (LVT) as a light-weight transformer backbone for vision models. LVT contains four stages and adopts CSA and RASA in the first and last three stages, respectively. The superior performances of LVT are demonstrated in ImageNet recognition, ADE20K semantic segmentation, and COCO panoptic segmentation.

2. Related Work

Vision Transformer. ViT [18] is the first vision transformer that proves that the NLP transformer [53] architecture can be transferred to the image recognition task with excellent performances. After ViT, a series of improving methods are proposed. For training, DeiT [51] introduces the knowledge distillation strategy for transformer. For tokenization, T2T-ViT [64] recursively aggregates neighboring tokens into one token to enrich local structure modeling. TNT [20] further splits tokens into smaller ones, extracts their features and integrates them with normal token features. For position embedding, CPVT [13] proposes dynamic position encoding that generalizes to images with arbitrary resolutions. For multi-scale processing, Twins [12] investigates the combination of local and global self-attention. CoaT [62] studies the cross-attention among features at various scales from different stages. CrossViT [3] proposes dual-path transformers that process tokens of different scales. For hierarchical design, Swin-Transformer [38] and PVT [57] both adopt four-stage design and gradually downsample the feature maps which is beneficial to the downstream tasks.

Combining Convolution and Self-Attention. There are four categories of methods. The first one is incorporating the position embedding in self-attention with convolution, including CPVT [13] and CoaT [62]. The second one is applying convolution before self-attention, including CvT [59], CoAtNet [16] and BoTNet [49]. The third one is inserting convolution after self-attention, including LocalViT [31] and PVTv2 [56]. The fourth one is to parallel self-attention and convolution, including AA [2] and LESA [63]. Different from all the above methods that merge local convolutions with global self-attention, we propose Convolutional Self-Attention (CSA) that combines self-attention and convolution both with 3×3 kernels as a powerful layer in the first stage of the model.

Recursive Convolutional Neural Networks. Recursive methods have been exploited on the convolutional neural networks (CNNs) for various vision tasks. It includes image recognition [33], super resolution [29], object detection [33, 37, 42, 48], semantic segmentation [11, 34]. Unlike these methods, we investigate a recursive method in the light-weight vision transformer as a general model backbone. Specifically, we propose a recursive self-attention

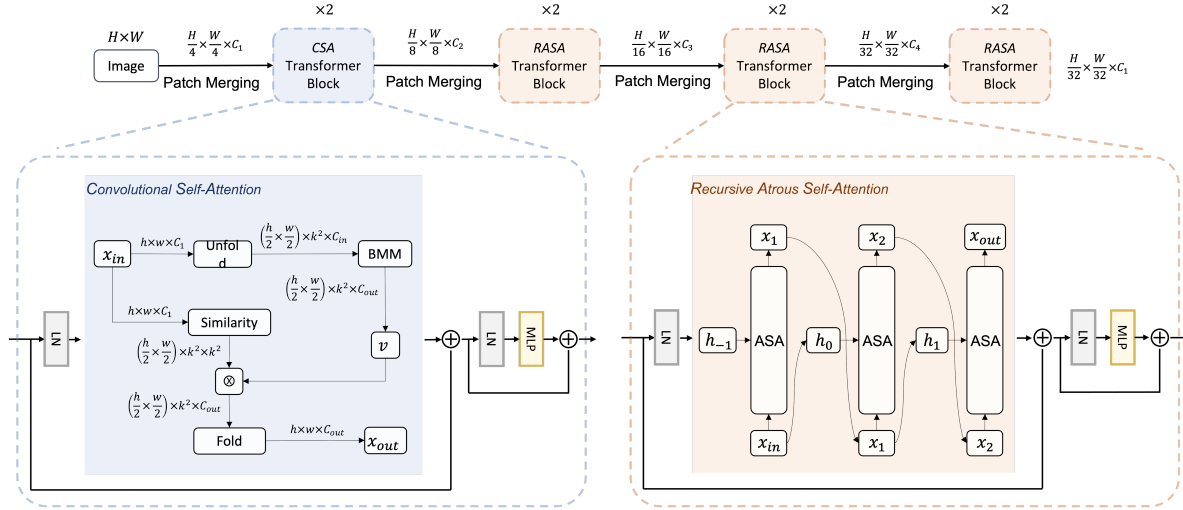


Figure 2. Lite Vision Transformer (LVT). The top row represents the overall structure of LVT. The bottom left and right parts visualize the proposed Convolutional Self-Attention (CSA) and Recursive Atrous Self-Attention (RASA). H, W represents the height and width of the image. C is the feature channel. The output resolution of each module is shown. Both the `Unfold` and `Fold` operations has a stride of 2. BMM stands for batched matrix multiplication, which corresponds to $W_{i \rightarrow j} x_j$ in Eqn. (1) with the batch dimension being the number of spatial locations in a local window. ASA stands for the proposed Atrous Self-Attention.

layer with the multi-scale query information, which improves the performance of the mobile model effectively.

Unlike the concurrent light-weight model, MobileFormer [8] which mixes the mobile CNN and transformer architecture, our work, LVT, is only based on transformer architecture with enhanced self-attention layers. Transformer architecture is demonstrated to be more parameter efficient and leads to SOTA performances on downstream tasks [9, 32, 38].

3. Lite Vision Transformer

We propose Lite Vision Transformer (LVT), which is shown in Fig 2. As a backbone network for multiple vision tasks, we follow the standard four-stage design [21, 38, 57]. Each stage performs one downsampling operation and consists of a series of building blocks. Their output resolutions are from stride-4 to stride-32 gradually. Unlike previous vision transformers [18, 38, 57, 65], LVT is proposed with limited amount of parameters and two novel self-attention layers. The first one is the Convolutional Self-Attention layer which has a 3×3 sliding kernel and is adopted in the first stage. The second one is the Recursive Atrous Self-Attention layer which has a global kernel and is adopted in the last three stages.

3.1. Convolutional Self-Attention (CSA)

The global receptive field benefits the self-attention layer in feature extraction. However, convolution is preferred in the early stages of vision models [16] as the locality is more important in processing low-level features. Unlike previous

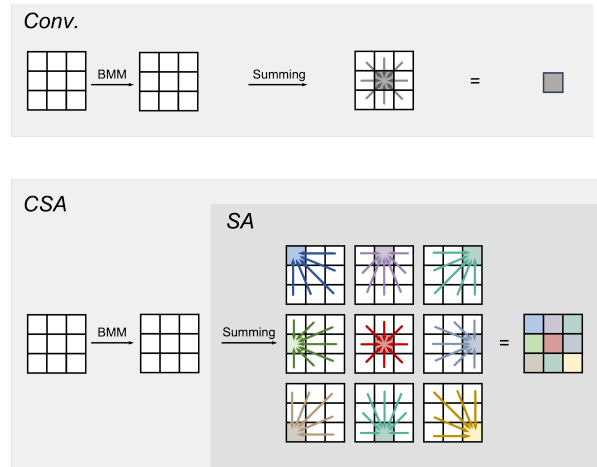


Figure 3. Illustration of Convolutional Self-Attention (CSA) in a 3×3 local window. The outputs of convolution and CSA are 1×1 and 3×3 , respectively. Mathematically, convolution comprises two procedures: the batched matrix multiplication (BMM) and summation. BMM corresponds to $W_{i \rightarrow j} x_j$ in Eqn. (1) with the batch dimension being the number of spatial locations. CSA has the BMM operation, but has the same summation process as SA. It performs 9 different input-dependent summations with the weights α in Eqn. (2), which process is shown by the colored arrows and patches. Through this design, CSA contains both the learnable filter and dynamic kernel.

methods of combining convolution and large kernel (global self-attention [16, 59], we focus on designing a window-based self-attention layer that has a 3×3 kernel and incorporates the representation of convolution.

Operations	Input dependent	Learnable filter
Conv.		✓
Self-Attention	✓	
Conv. Self-Attention	✓	✓

Table 1. Convolutional Self-Attention: Generalization of convolution and self-attention.

Analyzing Convolution. Let $x, y \in \mathbb{R}^d$ be the input and output feature vectors where d represents the channel number. Let $i, j \in \mathbb{R}$ index the spatial locations. Convolution is computed by sliding windows. In each window, we can write the formula of convolution as:

$$y_i = \sum_{j \in N(i)} W_{i \rightarrow j} x_j \quad (1)$$

where $N(i)$ represents the spatial locations in this local neighborhood that is defined by the kernel centered at location i . $|N(i)| = k \times k$ where k is the kernel size. $i \rightarrow j$ represents the relative spatial relationship from i to j . $W_{i \rightarrow j} \in \mathbb{R}^{d \times d}$ is the projection matrix. In total, there are $|N(i)|$ W s in a kernel. A 3×3 kernel consists of 9 such matrices W s.

Analyzing Self-Attention. Self-Attention needs three projection matrices $W_q, W_k, W_v \in \mathbb{R}^{d \times d}$ to compute query, key and value. In this paper, we consider sliding window based self-attention. In each window, we can write the formula of self-attention as

$$y_i = \sum_{j \in N(i)} \alpha_{i \rightarrow j} W_v x_j \quad (2)$$

$$\alpha_{i \rightarrow j} = \frac{e^{(W_q x_i)^T W_k x_j}}{\sum_{z \in N(i)} e^{(W_q x_i)^T W_k x_z}}$$

where $\alpha_{i \rightarrow j} \in (0, 1)$ is a scalar that controls the contribution of the value in each spatial location in the summation. α is normalized by softmax operation such that $\sum_j \alpha_{i \rightarrow j} = 1$. Compared with convolution with the same kernel size k , the number of learnable matrices is three rather than k^2 . Recently, Outlook Attention [65] is proposed to predict α instead of calculating it by the dot product of query and key, and shows superior performances when the kernel size is small. We employ this calculation, and it can be written as:

$$\alpha_{i \rightarrow j} = \frac{W_{qk} x_i [j]}{\sum_{z \in N(i)} W_{qk} x_i [z]} \quad (3)$$

where $W_{qk} \in \mathbb{R}^{d \times k^2}$ and $[j]$ means j th element of the vector.

Convolutional Self-Attention (CSA). We generalize self-attention and convolution into a unified convolutional self-attention as shown in Fig 3. Its formulation is shown in the

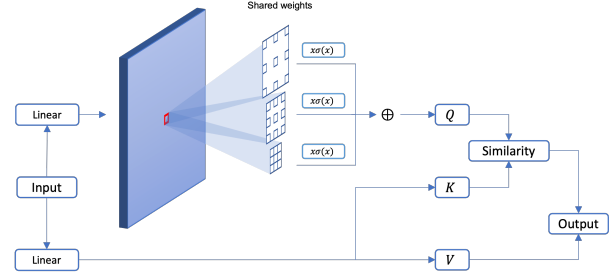


Figure 4. Illustration of Atrous Self-Attention (ASA). Q, K, V stands for the query, key, and value in self-attention. ASA calculates the multi-scale query by three depth-wise convolutions after the linear projection. These convolutions share the kernel weights but have different dilation rates: 1, 3, 5. Their outputs are added with the weights calculated by sigmoid function for the purpose of self-calibration. This can be implemented by the SiLU function. The multi-scale information is utilized in calculating the similarity map which weights the summation of the values.

following:

$$y_i = \sum_{j \in N(i)} \alpha_{i \rightarrow j} W_{i \rightarrow j} x_j \quad (4)$$

Both SA and CSA have the output of size $k \times k$ for a local window. When $\alpha_{i \rightarrow j} = 1$ where all the weights are the same, CSA is the convolution for the output center. When $W_{i \rightarrow j} = W_v$ where all the projection matrices are the same, CSA is self-attention. As we employ the dynamic α predicted by the input, as shown in Eqn. (3), Outlook Attention [65] is a special case of CSA. CSA has a bigger capacity than Outlook Attention. We summarize its property in Tab. 1. By this generalization, CSA has both input-dependent kernel and learnable filter. It is designed for stronger representation capability in first stage of vision transformers.

3.2. Recursive Atrous Self-Attention (RASA)

Light-weight models are more efficient and more suitable for on-device applications [47]. However, their performances are limited by the small number of parameters even with advanced model architecture [60]. For light-weight models, we focus on enhancing their representation capabilities with marginal increase in the number of parameters. **Atrous Self-Attention (ASA).** Multi-scale features are beneficial in detecting or segmenting the objects [35, 67]. Atrous convolution [5, 6, 41] is proposed to capture the multi-scale context with the same amount of parameters as standard convolution. Weights sharing atrous convolution [42] is also demonstrated in boosting model performances. Unlike convolution, the feature response of self-attention is a weighted sum of the projected input vectors from all spatial locations. These weights are determined by

Architecture	Stage1	Stage2	Stage3	Stage4
SA Type	CSA	RASA	RASA	RASA
SA Kernel	3×3	Global	Global	Global
Layer Num.	2	2	2	2
Feature Res.	$\frac{H}{4} \times \frac{W}{4}$	$\frac{H}{8} \times \frac{W}{8}$	$\frac{H}{16} \times \frac{W}{16}$	$\frac{H}{32} \times \frac{W}{32}$
Feature Dim.	64	64	160	256
Heads Num.	2	2	5	8
MLP Ratio	4	8	4	4
SR Ratio	–	4	2	1

Table 2. Model architecture of Lite Vision Transformer (LVT). We follow the canonical four-stage design [21]. All stages consist of transformer blocks [18]. H, W are the input resolutions. SR [57] mechanism is adopted for the model efficiency. We propose CSA and RASA as two enhanced Self-Attention layers (SA) to process low and high level features, respectively, in the light-weight models.

the similarities between the queries and keys, and represent the strength of the relationship among any pair of feature vectors. Thus we add multi-scale information when generating these weights shown in Fig 4. Specifically, we upgrade the calculation of the query from a 1×1 convolution to the following operation:

$$Q = \sum_{r \in \{1,3,5\}} \text{SiLU}(\text{Conv}(\hat{Q}, W_q^{k=3}, r, g = d)) \quad (5)$$

where

$$\hat{Q} = \text{Conv}(X, W_q^{k=1}, r = 1, g = 1), \quad (6)$$

$$\text{SiLU}(m) = m \odot \text{sigmoid}(m). \quad (7)$$

$X, Q \in \mathbb{R}^{d \times H \times W}$ are the feature maps, and $W_q^k \in \mathbb{R}^{k^2 \times d \times d/g}$ is the kernel weight. H, W are the spatial dimensions. d is the feature channels. k, r and g represent the kernel size, dilation rate, and group number of the convolution. We first use the 1×1 convolution to apply linear projection. Then we apply three convolutions that have different dilation rates but a shared kernel to capture the multi-scale contexts. The parameter cost is further reduced by setting the group number equal to the feature channel number. The parallel features of different scales are then weighted summed. We employ a self-calibration mechanism that determines the weights for each scale by their activation strength. This can be implemented by the SiLU [22, 45]. By this design, the similarity calculation of the query and key between any pair of spatial locations in self-attention uses the multi-scale information.

Resursive Atrous Self-Attention (RASA). For light-weight models, we intend to increase their depths without increasing the parameter usage. Recursive methods have been proposed in many vision tasks for Convolutional Neural Networks (CNNs) including [29, 33, 42, 50]. Unlike

Models	Top-1 (%)	Params (M)	FLOPs (G)
MobileNetV2 [47]	71.9	3.5	0.3
PVTv2-B0 [56]	70.5	3.4/3.7	0.6
LVT	74.8	3.4/5.5	0.9
ResNet50 [21]	76.1	25.6	4.1
ResNeXt50-32x4d [61]	77.6	25.0	4.3
SENet50 [25]	77.7	28.1	3.9
RegNetY-4G [43]	80.0	21.0	4.0
DeiT-Small/16 [51]	79.9	22.1	4.6
CPVT-S-GAP [13]	81.5	23.0	–
T2T-ViT _t -14 [64]	81.7	21.5	6.1
DeepViT-S [70]	82.3	27.0	6.2
ViP-Small/7 [23]	81.5	25.0	–
PVTv1-Small [57]	79.8	24.5	3.8
TNT-S [20]	81.5	23.8	5.2
CvT-13 [59]	81.6	20.0	4.5
CoaT-Lite Small [62]	81.9	20.0	4.0
Twins-SVT-S [12]	81.7	24.0	2.9
CrossViT-15 [3]	82.3	28.2	6.1
CvT-21 [59]	82.5	32.0	7.1
BoTNet-S1-59 [49]	81.7	33.5	7.3
RegNetY-8G [43]	81.7	39.0	8.0
T2T-ViT _t -19 [64]	82.2	39.2	9.8
Swin-T [38]	81.2	28.0	4.5
PVTv2-B2 [56]	82.0	24.8/25.4	4.0
LVT_scaled up	83.3	24.8/32.2	5.2

Table 3. ImageNet Classification. Params: encoder (transferable to other tasks) / encoder + head. Following MobileNetV2 [47] and PVTv2-B0 [56], we limit the parameter size of the encoder less than 3.5M. In order to compare LVT with other classifiers, we scale LVT to the size of the canonical network, ResNet50 [21]. We can observe that LVT shows superior performances.

these methods, we propose a recursive method for self-attention. The design follows the pipeline of the standard recurrent networks [19, 28]. Together with atrous self-attention (ASA), we propose recursive atrous self-attention (RASA), and its formula can be written in the following:

$$\begin{aligned} x_{t+1} &= \text{ASA}(\mathbf{F}(X_t, h_{t-1})) \\ h_{t-1} &= X_{t-1} \\ X_t &= \text{ASA}(\mathbf{F}(X_{t-1}, h_{t-2})) \end{aligned} \quad (8)$$

where t is the step and $h \in \mathbb{R}^{d \times H \times W}$ the hidden state. We take **ASA** as the non-linear activation function. The initial hidden state $h_{-1} = \mathbf{0}$. $\mathbf{F}(X, h) = W_F X + U_F h$ is the linear function combining the input and hidden state. W_F, U_F are the projection weights. However, we empirically find that setting $W_F = 1, U_F = 1$ provides the best performances and avoids introducing extra parameters. We set the recursion depth as two in order to limit the computation cost.

Method	Encoder	mIoU	Params (M)	FLOPs (G)	FPS (512)
FCN [39]	MobileNetV2 [47]	19.7	9.8	39.6	64.4
PSPNet [67]	MobileNetV2 [47]	29.6	13.7	52.9	57.7
DeepLabV3+ [7]	MobileNetV2 [47]	34.0	15.4	69.4	43.1
SegFormer [60]	MiT-B0 [60]	37.4	3.8	8.4	50.5
SegFormer [60]	LVT	39.3	3.9	10.6	45.5

Table 4. Mobile ADE20K semantic segmentation. We report the results for the single-scale input. The FPS is calculated on the 2,000 images whose short sides are rescaled to 512 with the aspect ratio unchanged. It is observed that LVT achieves significant performance gain compared with previous SOTA mobile semantic segmentation models.

Method	Backbone	COCO val			COCO test-dev			Params (M)	FLOPs (G)	FPS (1333, 800)
		PQ	PQ th	PQ st	PQ	PQ th	PQ st			
Panoptic FPN [30]	MobileNetV2 [47]	36.3	42.9	26.4	36.4	43.0	26.5	4.1	32.9	35.8
Panoptic FPN [30]	PVTv2-B0 [56]	41.3	47.5	31.9	41.2	47.7	31.5	5.3	49.7	23.5
Panoptic FPN [30]	LVT	42.8	49.5	32.6	43.0	49.9	32.6	5.4	56.4	20.4

Table 5. Mobile COCO panoptic segmentation. The FPS is calculated on the 2,000 high-resolution images. They are rescaled such that the maximum length does not exceed 1333 and the minimum length 800. The aspect ratio is kept. It is observed that LVT achieves significant performance improvement over previous SOTA mobile encoders for panoptic segmentation.

3.3. Model Architecture

The architecture of LVT is shown in Tab. 2. We adopt the standard four-stage design [21]. Four Overlapped Patch Embedding layers [60] are employed. The first one down-samples the image into stride-4 resolution. The other three downsample the feature maps to the resolution of stride-8, stride-16, and stride-32. All stages comprise the transformer blocks [18]. Each block contains the self-attention layer followed by an MLP layer. CSA is embedded in the stage-1 while RASA in the other stages. They are enhanced self-attention layers proposed to process local and global features in LVT.

4. Experiments

4.1. ImageNet Classification

Dataset. We perform image recognition experiments on ILSVRC2012 [46], a popular subset of the ImageNet database [17]. The training and validation sets contain 1.3M and 50K images, respectively. There are 1,000 object categories in total. The classes are distributed approximately and strictly uniformly in the training and validation sets.

Settings. The training setting follows previous conventions. We use AdamW as the optimizer [40]. Following previous works [27, 51], the learning rate is scaled based on the batch size with the formula being $lr = \frac{\text{batch_size}}{1024} \times lr_base$. We set lr_base as 1.6×10^{-3} . The weight decay of 5×10^{-2} is adopted. Stochastic depth with drop path rate being 0.1 is employed [26]. In total, there are 300 training epochs. Following [65], we use CutOut [68], RandAug [15], and Token Labeling [27] as the data augmentation methods. Class

attention layer [52] is used as the post stage. Both in the training and testing phase, the input resolution is 224×224 .

Results. The results are shown in Tab. 3. We limit the encoder size less than 3.5M, following MobileNet [47] and PVTv2-B0 [56]. The encoder is our design focus as it is the backbone used by other complex tasks like detection and segmentation. In order to compare LVT with other standard models, we scale LVT to the size of ResNet50 [21], a canonical backbone of vision models. The high performances of LVT for image recognition is demonstrated.

4.2. Mobile ADE20K Semantic Segmentation

Dataset. We perform semantic segmentation task on the challenging ADE20K dataset [69]. There are 150 categories in total, including 35 stuff classes and 115 discrete objects. The training and validation sets contain 20,210 and 2,000 images, respectively.

Settings. Previous conventions are followed. We adopt the Segformer framework [60] and use the MLP decoder. The LVT encoder is pretrained on ImageNet-1K without extra data. The decoder is trained from scratch. We use mmsegmentation [14] as the codebase. The AdamW optimizer [40] with the initial learning rate being 6×10^{-5} is used. The weight decay is set as 1×10^{-2} . The poly learning rate schedule with power being 1 is employed. There are 160K training iterations in total and the batch size is 16. For data augmentation, we randomly resize the image with ratio 0.5 – 2.0 and then perform random cropping of size 512×512 . Horizontal flipping with probability 0.5 is applied. During evaluation, we perform single-scale test.

Results. The results are summarized in Tab. 4. The FLOPs

Tasks	ImageNet Classification				ADE20K Semantic Segmentation		
Methods	Accuracy (%)		Params (M)	FLOPs (G)	mIoU	Params (M)	FLOPs (G)
	Top-1	Top-5					
VOLO-D0 [65]	74.6	92.5	3.9	1.9	39.3	3.24	12.28
VOLO-D0 + CSA	75.2	92.9	4.0	1.9	40.0	3.39	12.38
VOLO-D0 + CSA + RASA	75.6	92.9	4.0	2.2	41.0	3.40	15.70

Table 6. Ablation studies. VOLO is used as the base network to add CSA and RASA, because VOLO uses the self-attention with 3×3 kernel in the first stage. By this comparison, the performance gain from local self-attention to Convolutional Self-Attention (CSA) can be clearly illustrated. It is demonstrated that both CSA and RASA significantly contributes to the performance improvement.

is calculated with the input resolution 512×512 . The FPS is calculated on 2000 images on a single NVIDIA V100 GPU. During inference, the images are resized such that the short side is 512. We only use single-scale testing. The model is compact. Together with the decoder, the parameters are less than 4M. We can observe that LVT demonstrates the best performance among all previous mobile methods for semantic segmentation.

4.3. Mobile COCO Panoptic Segmentation

Dataset. We perform panoptic segmentation on COCO dataset. The 2017 split is employed. It has 118K training images and 5K validation images. On average, each image contains 3.5 categories and 7.7 instances. We choose panoptic segmentation to evaluate our methods as it unifies the object recognition, detection, localization, and segmentation at the same time. We aim to thoroughly evaluate the performance of our model.

Settings. The panoptic FPN [30] framework is adopted. All the models are trained in this framework for fair comparisons. We use mmdetection [4] as the codebase. AdamW [40] optimizer with the initial learning rate 3×10^{-4} is used. The weight decay is 1×10^{-4} . The $3 \times$ schedule is employed. There are 36 training epochs in total, the learning rate is decayed by 10 times after 24 and 33 epochs. We adopt multi-scale training. During training, the images are randomly resized. The maximum length does not exceed 1333. The maximum allowable length of the short side is randomly sampled in the range of 640 – 800. Random horizontal flipping with probability 0.5 is applied. During testing, we perform single-scale testing.

Results. The results are shown in Tab. 5. The FLOPs is calculated on the input resolution 1200×800 . During the inference, all the images are resized such that the large side is not larger than 1333 and the short side is less than 800. The FLOPs are calculated on 2000 high-resolution images with a single NVIDIA V100 GPU. The whole model including the decoder takes less than 5.5M parameters. We can observe the superiority of LVT compared with the previous SOTA encoders for mobile panoptic segmentation.



Figure 5. Mobile COCO panoptic segmentation.

5. Ablation Studies

5.1. Recursion Times of RASA

In this section, we investigate the relationship between the recursion times and model performances. The experiments are performed on ImageNet classification. We set the recursion time from 1 to 4. The results are summarized in Tab. 7.

Models	Top-1 (%)	Top-5 (M)	Params (G)	FLOPs
LVT_R1	73.9	92.1	3.4/5.5	0.8
LVT_R2	74.8	92.6	3.4/5.5	0.9
LVT_R3	74.6	92.5	3.4/5.5	1.0
LVT_R4	74.9	92.6	3.4/5.5	1.1

Table 7. Relationship of recursion times and performances on ImageNet Classification. *R* means recursion times. The performance increases dramatically with two iterations. Considering the efficiency, we use LVT_R2 in the main experiments.

5.2. Contributions of CSA and RASA

Settings. In this section, we study the performance contributions of Convolutional Self-Attention (CSA) and Recursive Atrous Self-Attention. To this end, we build our model via the recently proposed VOLO which employed small kernel self-attention in the first stage. As VOLO is demonstrated as a powerful backbone in image recognition and semantic segmentation, we perform experiments on ImageNet and ADE20K. In order to perform the comparisons in the mobile setting, we scale VOLO to have a parameter size 4.0M. Specifically, we set the layer number of each stage to be 2, and adjust the feature dimensions to be 96, 192, 192, 192. All the other settings are kept unchanged.

Results. The results are reported in Tab. 6. For ImageNet classification, the input resolution in both the training and testing is 224×224 . For ADE20K semantic segmentation, we insert the VOLO and LVT with the MLP decoder, following the Segformer framework [60]. During testing, the short side of the image is resized to 512. It is observed both CSA and RASA significantly contribute to the performance gain.

6. Conclusion

In this work, we propose a powerful light-weight transformer backbone, Lite Vision Transformer (LVT). It consists of two novel Self-attention layers: Convolutional Self-Attention (CSA) and Recursive Atrous Self-Attention (RASA). They are used in the first and the last three stages of LVT to process low and high level features. We demonstrate the strong performances of LVT compared with previous mobile methods in tasks of visual recognition, semantic segmentation, and panoptic segmentation.

Limitations. LVT is a light-weight model. The natural limitation is the weak representation power compared with models with large number of parameters. The focus of this work is on the mobile models. Our future work includes scaling LVT to the large powerful backbones.

Acknowledgements. The authors gratefully acknowledge supports from ONR N00014-21-1-2812.



Figure 6. Visual effectiveness of CSA and RASA in LVT.

References

- [1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014. 2
- [2] Irwan Bello, Barret Zoph, Ashish Vaswani, Jonathon Shlens, and Quoc V Le. Attention augmented convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3286–3295, 2019. 2
- [3] Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. *arXiv preprint arXiv:2103.14899*, 2021. 2, 5
- [4] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 7
- [5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014. 4
- [6] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 4
- [7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 6
- [8] Yinpeng Chen, Xiyang Dai, Dongdong Chen, Mengchen Liu, Xiaoyi Dong, Lu Yuan, and Zicheng Liu. Mobileformer: Bridging mobilenet and transformer. *arXiv preprint arXiv:2108.05895*, 2021. 3
- [9] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. *arXiv preprint arXiv:2112.01527*, 2021. 3
- [10] Bowen Cheng, Alexander G Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *arXiv preprint arXiv:2107.06278*, 2021. 1
- [11] Ho Kei Cheng, Jihoon Chung, Yu-Wing Tai, and Chi-Keung Tang. Cascadepsp: toward class-agnostic and very high-resolution segmentation via global and local refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8890–8899, 2020. 2
- [12] 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*, 1(2):3, 2021. 2, 5
- [13] 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. 2, 5
- [14] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. <https://github.com/open-mmlab/mmdetection>, 2020. 6
- [15] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc 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. 6
- [16] Zihang Dai, Hanxiao Liu, Quoc V Le, and Mingxing Tan. Coatnet: Marrying convolution and attention for all data sizes. *arXiv preprint arXiv:2106.04803*, 2021. 2, 3
- [17] J. Deng, W. Dong, R. Socher, L.J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition*, 2009. 6
- [18] 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. *arXiv preprint arXiv:2010.11929*, 2020. 1, 2, 3, 5, 6
- [19] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990. 2, 5
- [20] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *arXiv preprint arXiv:2103.00112*, 2021. 2, 5
- [21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 3, 5, 6
- [22] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 5
- [23] Qibin Hou, Zihang Jiang, Li Yuan, Ming-Ming Cheng, Shuicheng Yan, and Jiashi Feng. Vision permutator: A permutable mlp-like architecture for visual recognition. *arXiv preprint arXiv:2106.12368*, 2021. 5
- [24] Han Hu, Zheng Zhang, Zhenda Xie, and Stephen Lin. Local relation networks for image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3464–3473, 2019. 2
- [25] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 5
- [26] Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *European conference on computer vision*, pages 646–661. Springer, 2016. 6
- [27] Zihang Jiang, Qibin Hou, Li Yuan, Daquan Zhou, Yujun Shi, Xiaojie Jin, Anran Wang, and Jiashi Feng. All tokens matter: Token labeling for training better vision transformers. *arXiv preprint arXiv:2104.10858*, 2021. 6
- [28] Michael I Jordan. Serial order: A parallel distributed processing approach. In *Advances in psychology*, volume 121, pages 471–495. Elsevier, 1997. 2, 5
- [29] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1637–1645, 2016. 2, 5

- [30] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6399–6408, 2019. 6, 7
- [31] Yawei Li, Kai Zhang, Jie Zhang Cao, Radu Timofte, and Luc Van Gool. Localvit: Bringing locality to vision transformers. *arXiv preprint arXiv:2104.05707*, 2021. 2
- [32] Zhiqi Li, Wenhai Wang, Enze Xie, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, Tong Lu, and Ping Luo. Panoptic segformer. *arXiv preprint arXiv:2109.03814*, 2021. 1, 3
- [33] Ming Liang and Xiaolin Hu. Recurrent convolutional neural network for object recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3367–3375, 2015. 2, 5
- [34] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1925–1934, 2017. 2
- [35] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 4
- [36] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2
- [37] Yudong Liu, Yongtao Wang, Siwei Wang, TingTing Liang, Qijie Zhao, Zhi Tang, and Haibin Ling. Cbnet: A novel composite backbone network architecture for object detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 11653–11660, 2020. 2
- [38] 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 arXiv:2103.14030*, 2021. 1, 2, 3, 5
- [39] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 6
- [40] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6, 7
- [41] George Papandreou, Iasonas Kokkinos, and Pierre-André Savalle. Modeling local and global deformations in deep learning: Epitomic convolution, multiple instance learning, and sliding window detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 390–399, 2015. 4
- [42] Siyuan Qiao, Liang-Chieh Chen, and Alan Yuille. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10213–10224, 2021. 2, 4, 5
- [43] Ilija Radosavovic, Raj Prateek Kosaraju, Ross Girshick, Kaiming He, and Piotr Dollár. Designing network design spaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10428–10436, 2020. 5
- [44] Prajit Ramachandran, Niki Parmar, Ashish Vaswani, Irwan Bello, Anselm Levskaya, and Jonathon Shlens. Stand-alone self-attention in vision models. *arXiv preprint arXiv:1906.05909*, 2019. 2
- [45] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Swish: a self-gated activation function. *arXiv preprint arXiv:1710.05941*, 7:1, 2017. 5
- [46] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 2, 6
- [47] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 1, 2, 4, 5, 6
- [48] Bharat Singh and Larry S Davis. An analysis of scale invariance in object detection snip. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3578–3587, 2018. 2
- [49] Aravind Srinivas, Tsung-Yi Lin, Niki Parmar, Jonathon Shlens, Pieter Abbeel, and Ashish Vaswani. Bottleneck transformers for visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16519–16529, 2021. 2, 5
- [50] Ying Tai, Jian Yang, and Xiaoming Liu. Image super-resolution via deep recursive residual network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3147–3155, 2017. 5
- [51] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 2, 5, 6
- [52] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. *arXiv preprint arXiv:2103.17239*, 2021. 6
- [53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 1, 2
- [54] Ziyu Wan, Jingbo Zhang, Dongdong Chen, and Jing Liao. High-fidelity pluralistic image completion with transformers. *arXiv preprint arXiv:2103.14031*, 2021. 1
- [55] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5463–5474, 2021. 1
- [56] 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. [1](#), [2](#), [5](#), [6](#)
- [57] 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. [1](#), [2](#), [3](#), [5](#)
- [58] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018. [2](#)
- [59] 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. [2](#), [3](#), [5](#)
- [60] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *arXiv preprint arXiv:2105.15203*, 2021. [1](#), [2](#), [4](#), [6](#), [8](#)
- [61] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. [5](#)
- [62] Weijian Xu, Yifan Xu, Tyler Chang, and Zhuowen Tu. Co-scale conv-attentional image transformers. *arXiv preprint arXiv:2104.06399*, 2021. [2](#), [5](#)
- [63] Chenglin Yang, Siyuan Qiao, Adam Kortylewski, and Alan Yuille. Locally enhanced self-attention: Rethinking self-attention as local and context terms. *arXiv preprint arXiv:2107.05637*, 2021. [2](#)
- [64] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zihang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. *arXiv preprint arXiv:2101.11986*, 2021. [2](#), [5](#)
- [65] Li Yuan, Qibin Hou, Zihang Jiang, Jiashi Feng, and Shuicheng Yan. VOLO: Vision outlooker for visual recognition. *arXiv preprint arXiv:2106.13112*, 2021. [1](#), [3](#), [4](#), [6](#), [7](#)
- [66] Hengshuang Zhao, Jiaya Jia, and Vladlen Koltun. Exploring self-attention for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10076–10085, 2020. [2](#)
- [67] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. [4](#), [6](#)
- [68] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13001–13008, 2020. [6](#)
- [69] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. [2](#), [6](#)
- [70] Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Zihang Jiang, Qibin Hou, and Jiashi Feng. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021. [5](#)