Embedding-Free Transformer with Inference Spatial Reduction for Efficient Semantic Segmentation

Hyunwoo Yu*[1], Yubin Cho*[1,2], Beoungwoo Kang*[1], Seunghun Moon*[1], Kyeongbo Kong*[3], and Suk-Ju Kang†[1]

[1] Department of Electronics Engineering, Sogang University, South Korea
[2] AI Lab, CTO Division, LG Electronics, South Korea
[3] Department of Electrical & Electronics Engineering, Pusan National University, South Korea
{hyunwoo137, dbqls1219, beoungwoo, moonsh97, sjkang}@sogang.ac.kr
kbkong@pusan.ac.kr

Abstract. We present an Encoder-Decoder Attention Transformer, EDAFormer, which consists of the Embedding-Free Transformer (EFT) encoder and the all-attention decoder leveraging our Embedding-Free Attention (EFA) structure. The proposed EFA is a novel global context modeling mechanism that focuses on functioning the global non-linearity, not the specific roles of the query, key and value. For the decoder, we explore the optimized structure for considering the globality, which can improve the semantic segmentation performance. In addition, we propose a novel Inference Spatial Reduction (ISR) method for the computational efficiency. Different from the previous spatial reduction attention methods, our ISR method further reduces the key-value resolution at the inference phase, which can mitigate the computation-performance trade-off gap for the efficient semantic segmentation. Our EDAFormer shows the state-of-the-art performance with the efficient computation compared to the existing transformer-based semantic segmentation models in three public benchmarks, including ADE20K, Cityscapes and COCO-Stuff. Furthermore, our ISR method reduces the computational cost by up to 61% with minimal mIoU performance degradation on Cityscapes dataset. The code is available at https://github.com/hyunwoo137/EDAFormer.

Keywords: Semantic segmentation · Embedding-free self-attention · Inference spatial reduction

1 Introduction

Semantic segmentation, which aims to obtain the accurate pixel-wise prediction for the whole image, is one of the most fundamental tasks in the computer vision [28,37] and is widely used in various downstream applications [10,11,29].

*Equal Contribution
†Corresponding Author

From the CNN-based models [4,26,36,37,42,44] to the transformer-based models [21,35,48,50,51,53,60,66], semantic segmentation models have been introduced in different structures. However, compared to other tasks, the semantic segmentation has a large amount of computation, as it treats the high resolution images and requires the per-pixel prediction decoder. Therefore, it is a significant challenge to explore the efficient structure for this task.

With the great success of the Vision Transformer [19] (ViT), recent semantic segmentation models [5,17,23,38,40,41,45,55–57] mainly utilize the transformer-based structure to improve the performance by modeling the global context via the self-attention mechanism, and various advanced self-attention structures [6, 18, 20, 22, 25, 30, 31, 33, 39, 57, 61, 63] have been introduced. In this paper, we analyze the general self-attention mechanism as two parts. The first is that the input feature is assigned the specific roles as the query, key and value by embedding the input features through the linear projection with the learnable parameters. The second is functioning as a global non-linearity, which obtains the attention weight between the query and the key via the softmax and then projects the attention weight into the value. We focus on that the real important part of global context modeling is the global non-linear functioning, not the specific roles (*i.e.*, the query, key, and value) assigned to the input feature. We found that the simple but effective method, which removes the specific roles of the input feature, rather improves the performance. Therefore, we propose a novel self-attention structure, Embedding-Free Attention (EFA), which omits the embeddings of the query, key and value.

With this powerful module, we also propose a semantic segmentation model, Encoder-Decoder Attention Transformer (EDAFormer), which is composed of the proposed Embedding-Free Transformer (EFT) encoder and the all-attention decoder. For the encoder, we adopt the hierarchical structure, and leverage our EFA module in the transformer blocks that effectively extract the global context features. For the decoder, inspired by [23, 27, 65], our all-attention decoder not only leverages our EFA, which effectively extracts the global context, but also is explored which level features need more global attention in the decoder. We empirically found that the higher level feature is more effective to consider the global context. Therefore, we design the all-attention decoder that leverages the more number of EFA modules to the higher level feature.

In addition, this paper addresses the issue of requiring additional training in the different structures whenever lighter (or less lightweight) models for lower computation (or higher accuracy). This issue causes user inconvenience and limits the versatility of lightweight methodologies.

To solve this issue, we introduce a novel Inference Spatial Reduction (ISR) method that reduces the key-value resolution more at the inference phase than at the training phase. Our ISR exploits the Spatial Reduction Attention (SRA)-based structure in a completely different perspective with the existing SRA-based models [43, 50, 51, 53, 55], as we focus on making the reduction ratio different at training and inference. Through our method, the query learns a larger amount of the key and value information during training, and better copes with the reduced

key and value during inference. This has the following two advantages. (1) Our method reduces the computational cost with little degradation in performance. (2) Our method allows to selectively adjust various computational costs of one pretrained model.

We demonstrate the effectiveness of the proposed method in terms of the computational cost and performance on three public semantic segmentation benchmarks. Compared to the transformer-based semantic segmentation models, our model achieve the competitive performance in terms of the efficiency and the accuracy. Our contributions are summarized as follows:

- We propose a novel embedding-free attention structure that removes the specific roles of the query, key, and value but focuses on global non-linearity, thus achieving strong performance.
- We introduce a semantic segmentation model, EDAFormer, which is designed with the EFT encoder and the all-attention decoder. Our decoder exploits the more number of the proposed EFA module at the higher level to capture the global context more effectively.
- We propose a novel ISR method for the efficiency, which enables to reduce the computational cost with less degradation in performance at the inference phase and allows to selectively adjust the computational cost of the pretrained transformer model.
- Our EDAFormer outperforms the existing transformer-based semantic segmentation models in terms of the efficiency and the accuracy on three public semantic segmentation benchmarks.

2 Related Works

2.1 Attention for Global Context

The importance of modeling the global context has been demonstrated by the self-attention mechanism in the transformer. Beyond the general attention method, various attention methods have been studied. [50,51] proposed the spatial reduction attention mechanism, which reduces the key-value resolution for efficiency. [54] leveraged the pyramid pooling to reduce the key-value in multi-scale resolution. Based on the spatial reduction attention structure, [22,63,64] exploited the convolutional layer in the attention. The window-based attention method [34,35] considered the local window regions for efficiency. [12] proposed the local window attention with global attention. The convolution-based attention [14, 22, 53, 55] used the convolutional operation to consider local context with global context. The channel reduction attention method [27] reduced the query and key channels. However, all these self-attention methods are based on the query, key and value embeddings. Different from these methods, we propose the efficient Embedding-Free Attention module by focusing on that the global non-linearity is important in the attention mechanism.

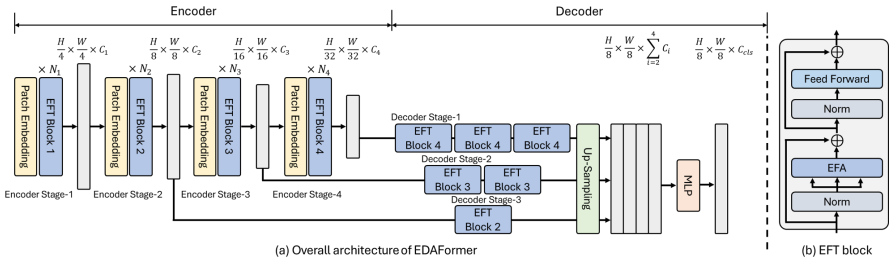

Fig. 1: (a) Overall architecture of the proposed EDAFormer, consisting of two main parts: an EFT encoder and an all-attention decoder. The encoder and decoder of EDAFormer are designed with the query, key and value embedding free attention structure. (b) Details of the EFT block that contains EFA module.

2.2 Transformer-based Semantic Segmentation

Since ViT [19] achieved the great performance in the image classification task, the transformer-based architectures have also been studied on the semantic segmentation, one of the most fundamental vision tasks. SETR [66] was the first semantic segmentation model to adopt the transformer architecture as a backbone with convolutional decoder. Beyond introducing the effective encoder structures, recent method [55] proposed the efficient encoder-decoder structures for the semantic segmentation. SegFormer [55] introduced a mix transformer encoder and a purely MLP-based decoder. FeedFormer [43] introduced a cross attention-based decoder to refer the low-level feature information of the transformer encoder. VWFormer [56] used the transformer encoder and exploited the window-based attention for considering the multi-scale representation in the decoder. We introduce the efficient Encoder-Decoder Attention TransFormer model for the semantic segmentation to effectively capture the global context at both the encoder and the decoder.

3 Proposed Method

This section introduces our Encoder-Decoder Attention Transformer (EDAFormer), which is composed of the Embedding-Free Transformer (EFT) encoder and the all-attention decoder. Additionally, we describe our Inference Spatial Reduction (ISR) method that can reduce the computational cost effectively.

3.1 Overall Architecture

EDAFormer. As shown in Fig. 1 (a), we leverage a hierarchical encoder structure, which is effective in the semantic segmentation task. When the input image is $I \in \mathbb{R}^{H\times W\times 3}$, the output feature of each stage is defined as $\mathbf{F}_i \in \mathbb{R}^{\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}\times C_i}$, where $i \in \{1,2,3,4\}$ denotes the index of the encoder stage, and C is the channel dimension. At each stage, the features are first downsampled by the patch embedding block before being input to the transformer block.

As illustrated in Fig. 1 (b), our transformer block structure of the encoder is composed of the Embedding-Free Attention (EFA) and the Feed-Forward Layer (FFL). As shown in Fig. 2 (b), our EFA module omits the linear projection for the query $\mathbf{Q}$, key $\mathbf{K}$ and value $\mathbf{V}$ embeddings, which are lightweight and effectively extracts the global context. Additionally, we adopt the spatial reduction attention (SRA) structure [51] to leverage our ISR in the inference phase. We use the non-parametric operations and the average pooling to reduce the key-value spatial resolution, which has less impact on performance with the spatial reduction in the inference phase. The EFA module is formulated as follows:

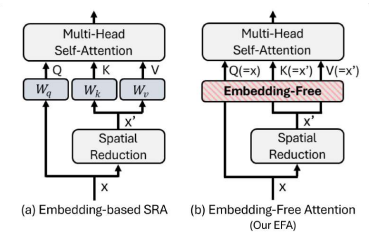

Fig. 2: Comparison of the previous method and our EFA.

$$\begin{aligned} \mathbf{Q} = \mathbf{x}_{in},\ \mathbf{K} = \mathbf{V} = \texttt{SR}(\mathbf{x}_{in}, R), \\ \mathbf{Att} = \texttt{softmax}(\mathbf{Q} \cdot \mathbf{K}^T / \sqrt{d_k}),\ \mathbf{x}_{out} = \mathbf{Att} \cdot \mathbf{V}, \end{aligned} \tag{1}$$

where $\texttt{SR}$ and R denote the spatial reduction via the average pooling and the reduction ratio, respectively. $\mathbf{x}_{in}$ is directly used as the query, and the spatial reduced features are used as the key-value. In the part where the softmax function is used for similarity scores between the query and the key, the global non-linearity can be applied to the input features, allowing the global context extraction without the specific roles of the query, key, and value. Then, the FFL is formulated as follows:

$$\texttt{FFL}(\mathbf{x}_{in}) = \texttt{Linear}((\texttt{DW}(\texttt{Linear}(\mathbf{x}_{in}))), \tag{2}$$

where $\texttt{DW}$ indicates the depth-wise convolution. As the EFA and FFL are connected sequentially, the whole process of our EFT block is formulated as:

$$\begin{aligned} \mathbf{z} &= \texttt{EFA}(\texttt{LN}(\mathbf{x}_{in})) + \mathbf{x}_{in}, \\ \mathbf{x}_{out} &= \texttt{FFL}(\texttt{LN}(\mathbf{z})) + \mathbf{z}, \end{aligned} \tag{3}$$

where $\mathbf{z}$ is the intermediate features, and $\texttt{LN}$ is a layer normalization. This embedding-free structure is effective for the classification and the semantic segmentation. In addition, we empirically find that our embedding-free structure is effective for our ISR in terms of considering the trade-off between the computation and the performance degradation.

All-attention decoder. As previous models [23, 58, 65] have demonstrated, applying the SRA to the encoder features in the decoder is effective for capturing the global semantic-aware features. We thus design an all-attention decoder, which consists of EFT blocks at all of the decoder stages. We also explore the optimal structure of the decoder for using EFT blocks. As a result, applying more attention blocks to the high-level features was effective for capturing globally

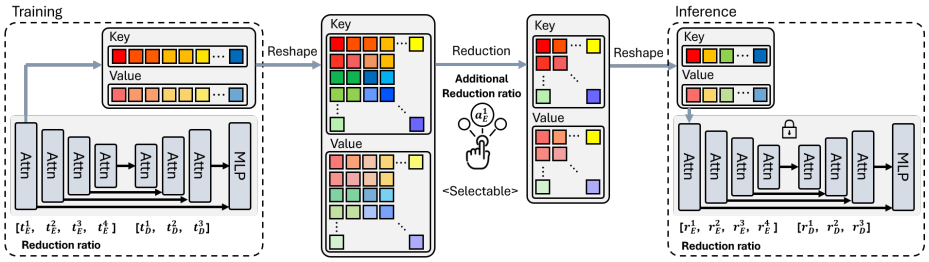

Fig. 3: Overview of our ISR method at the encoder stage-1. Our ISR applies the reduction ratio at the inference, reducing the key and value tokens selectively. This framework can be performed at every stage that contains the self-attention structure. It leads to flexibly reduce the computational cost without disrupting the spatial structure.

more semantic informative features. As shown in Fig. 1 (a), our decoder has a hierarchical structure that utilizes 3, 2, and 1 EFT blocks at the 1^{st} to 3^{rd} decoder stages, respectively. This structure is composed of a larger number of transformer blocks compared to the decoders of the previous transformer-based segmentation models, but has lower computational costs compared to previous models because the EFT block is lightweight.

In the all-attention decoder, the output features $\mathbf{F}_i$ of each encoder stage $i \in \{2, 3, 4\}$ are first fed into the EFT blocks in each decoder stage $j \in \{3, 2, 1\}$, where j denotes the index of the decoder stages. Then, the features $\widehat{\mathbf{F}}_j \in \mathbb{R}^{H_j \times W_j \times C_j}$ of each decoder stage are up-sampled to $H_2 \times W_2$ resolution using the bilinear interpolation. These up-sampled features $\mathbf{U}_j \in \mathbb{R}^{H_2 \times W_2 \times C_j}$ are then concatenated and passed to linear layers for fusion. Finally, the final prediction mask is projected into the number of classes C_{cls} mask by another linear layer. This process is formulated as:

$$\begin{aligned} \widehat{\mathbf{F}}_j &= \texttt{EFT}(\texttt{LN}(\mathbf{F}_i)) + \mathbf{F}_i, \ \forall i \\ \mathbf{U}_j = \texttt{Upsample}(\widehat{\mathbf{F}}_j), \ \mathbf{F}_c &= \texttt{Concat}(\mathbf{U}_j), \ \forall j \\ \mathbf{M} &= \texttt{Linear}(\texttt{Linear}(\mathbf{F}_c)), \end{aligned} \tag{4}$$

where $\mathbf{M} \in \mathbb{R}^{H_2 \times W_2 \times C_{cls}}$ is the final prediction mask.

3.2 Inference Spatial Reduction Method

Different from previous SRA, our inference spatial reduction (ISR) method reduces the key-value spatial resolution at the inference phase. Our method achieves the computational efficiency by changing the hyperparameter associated with the 'reduction ratio R' of the average pooling in the EFA module. Our ISR can be used in the self-attention structures because the self-attention has a special structure where reducing the resolution of key and value does not affect the shape of the input and output features. Due to this structure, the reduction

ratio can be adjusted during inference without affecting the resolution of the input and output features.

However, reducing the key and value resolution largely at training has the advantage of computational efficiency, but leads to the performance degradation because the query cannot consider enough information from the key and value. To address this issue, our ISR alleviates the trade-off gap between the computational cost and the accuracy by reducing the resolution of the key and value at inference. In this part, we describe that our ISR is applied to the our EDAFormer, which is the optimized architecture for applying our ISR effectively.

As shown in Fig. 1, our EDAformer uses the proposed transformer blocks in both encoder-decoder structures. Each pooling-based SRA used in each encoder stage and decoder stage has a corresponding reduction ratio setting that reduces the key and value resolution. At training as illustrated in Fig. 3, the reduction ratios t_E^i of each encoder stage are set to $\left[8,\ 4,\ 2,\ 1\right]$, which are the default setting of other previous models [50, 51, 55] using SRA. The reduction ratios t_D^j of the decoder stage that takes each encoder features are set to $\left[1,\ 2,\ 4\right]$, which are equal to the reduction ratios of the corresponding encoder stage. t_E and t_D denote the reduction ratio of the encoder and decoder at training, respectively. The computational complexity of the previous attention is as follows:

$$\Omega(\texttt{SRA}) = 2\frac{(hw)^2}{r^2}c, \tag{5}$$

where Ω and $\texttt{SRA}$ denote the computational complexity and the spatial reduction attention. h, w and c represent the height, the width and the channel of the features, respectively. r is the reduction ratio at training phase.

Under these reduction ratio settings, we train our EDAFormer to get pretrained weights. After that, at inference phase, it is possible to optionally adjust the inference computational reduction by selecting the reduction ratios at the discretion of the user. As shown in Fig. 3, r_E^i and r_D^j denote the reduction ratio of the encoder and decoder at inference, respectively. They are formulated as:

$$\begin{aligned} r_E^i &= t_E^i \times a_E^i,\ \forall i \\ r_D^j &= t_D^j \times a_D^j,\ \forall j \end{aligned} \tag{6}$$

where a_E^i and a_D^j denote the additional reduction ratio of the encoder and decoder at inference, respectively. After applying our ISR, the computational complexity is as follows:

$$\Omega(\texttt{ISR}(\texttt{SRA})) = 2\frac{(hw)^2}{r^2a^2}c, \tag{7}$$

where $\texttt{ISR}$ is the inference spatial reduction and a is the additional reduction ratio at inference. Therefore, one of the advantage of our ISR is that it is simple to obtain the computational reduction on the pretrained model without additional training. Our ISR reduces the performance degradation compared with reducing by r^2a^2 at training. Empirically, the optimal setting is $\left[16,8,2,1\right]$-$\left[2,4,8\right]$ in the encoder-decoder, which has the best reduction ratio of the performance degradation to the computational cost reduction.

Method	Params (M)	ADE20K GFLOPs ↓	ADE20K mIoU (%) ↑	Cityscapes GFLOPs ↓	Cityscapes mIoU (%) ↑	COCO-Stuff GFLOPs ↓	COCO-Stuff mIoU (%) ↑
Segformer-B0 [55]	3.8	8.4	37.4	125.5	76.2	8.4	35.6
FeedFormer [43]	4.5	7.8	39.2	107.4	77.9	-	-
VWFormer-B0 [56]	3.7	5.1	38.9	-	77.2	5.1	36.2
EDAFormer-T (w/o ISR)	4.9	5.6	42.3	151.7	78.7	5.6	40.3
EDAFormer-T (w/ ISR)	4.9	**4.7**	**42.1**	**94.9**	**78.7**	**4.7**	**40.3**
OCRNet [16]	70.5	164.8	45.6	1296.8	81.1	-	-
Swin UperNet-T [35]	60.0	236.0	44.4	-	-	-	-
ContrastiveSeg [49]	58.0	-	-	-	79.2	-	-
SenFormer [2]	144.0	179.0	46.0	-	-	-	-
Segformer-B2 [55]	27.5	62.4	46.5	717.1	81.0	62.4	44.6
ProtoSeg [68]	90.5	-	48.6	-	80.6	-	42.4
MaskFormer [9]	42.0	55.0	46.7	-	-	-	-
Mask2Former [8]	47.0	74.0	47.7	-	-	-	-
FeedFormer-B2 [43]	29.1	42.7	48.0	522.7	81.5	-	-
VWFormer-B2 [56]	27.4	38.5	48.1	-	81.7	38.5	45.2
EDAFormer-B (w/o ISR)	29.4	32.0	49.0	605.9	81.6	32.0	45.9
EDAFormer-B (w/ ISR)	29.4	**29.4**	**48.9**	**452.9**	**81.6**	**29.4**	**45.8**

Table 1: Comparison with the transformer-based state-of-the-art semantic segmentation model on three public datasets. GFLOPs are computed using 512×512 resolutions for ADE20K and COCO-Stuff, and 2048×1024 resolutions for Cityscapes.

4 Experiment

4.1 Experimental Settings

Datasets. ADE20K [67] is a challenging scene parsing dataset captured at indoors and outdoors. It consists of 150 semantic categories, and 20,210/2,000/3,352 images for training, validation, and testing. Cityscapes [13] is an urban driving scene dataset that contains 5,000 fine-annotated images with 19 semantic categories. It consists of 2,975/500/1,525 images in training, validation, and test sets. COCO-Stuff [3] is a challenging dataset, which contains 164,062 images labeled with 172 semantic categories.

Implementation details. The mmsegmentation codebase was used to train our model on 4 RTX 3090 GPUs. We pretrained our encoder on ImageNet-1K [15], and our decoder was randomly initialized. For classification and segmentation evaluation, we adopted Top-1 accuracy and mean Intersection over Union (mIoU), respectively. We applied the same training settings and data augmentation as PVTv2 [50] for ImageNet pretraining. We applied random horizontal flipping, random scaling with a ratio of 0.5-2.0 and random cropping with the size of 512×512, 1024×1024, and 512×512 for ADE20K, Cityscapes, and COCO-Stuff, respectively. The batch size was 16 for ADE20K and COCO-Stuff, and 8 for Cityscapes. We used the AdamW optimizer for 160K iterations on ADE20K, Cityscapes and COCO-Stuff.

4.2 Comparison with State-of-the-art Methods

Semantic segmentation. In Table 1, we compared our EDAFormer with the previous transformer-based methods on three public datasets. The comparison includes the parameter size, FLOPs, and mIoU performance. Our lightweight model, EDAFormer-T (w/ ISR), showed 42.1%, 78.7% and 40.3% mIoU, and our larger model, EDAFormer-B (w/ ISR), yielded 48.9%, 81.6% and 45.8% mIoU on each dataset. Compared to previous methods, both of our EDAFormer achieved the state-of-the-art performance with the efficient computation.

Models	Params (M)	GFLOPs	Top-1 Acc. (%)
RSB-ResNet-18 [26, 52]	12	1.8	70.6
PVTv2-B0 [51]	3.4	0.6	70.5
MiT-B0 [55]	3.7	0.6	70.5
EFT-T (Ours)	3.7	0.6	**72.3**
ResNet50 [26]	25.5	4.1	78.5
RSB-ResNet-152 [26, 52]	60.0	11.6	81.8
DeiT-S [47]	22.0	4.6	79.8
PVT-Small [50]	25.0	3.8	79.8
PVTv2-B2 [51]	25.4	4.0	82.0
MiT-B2 [55]	25.4	4.0	81.6
T2T-ViT-14 [62]	21.5	4.8	81.5
TNT-S [24]	23.8	4.8	81.5
ResMLP-S24 [46]	30.0	6.0	79.4
Swin-Mixer-T/D6 [35]	23.0	4.0	79.7
Visformer-S [7]	40.2	4.8	82.1
gMLP-S [32]	20.0	4.5	79.6
PoolFormer-S36 [59]	31.0	5.0	81.4
EfficientFormer-L3 [30]	31.3	3.9	82.4
FasterViT-0 [25]	31.4	3.3	82.1
EFT-B (Ours)	25.4	4.2	**82.4**

Table 2: Comparison with the previous models on ImageNet. GFLOPs were computed with 224×224.

EFT encoder on ImageNet. In Table 2, we compared our Embedding-Free Transformer (EFT) encoder with the existing models on ImageNet-1K classification. Our EFT achieved higher performance than other transformer models. This result indicates that our EFT backbone is effective in the classification task by considering the spatial information globally even without the embeddings of the query, key and value.

4.3 Effectiveness of our EFA at Decoder

To verify the effectiveness of considering the globality at the decoder, we compared the different operations at the Embedding-Free Attention (EFA) position of the EFT block in Table 3 (a). The applied operations are the local context operation (*i.e.*, DW Conv, Conv) and the global context operation (*i.e.*, w/ embedding attention, w/o embedding attention). Our w/o embedding structure improved 1.6% and 2.4% mIoU compared to the depth-wise convolution and the standard convolution, respectively. These results show that capturing the global context in the decoder is important for the mIoU performance improvement. While w/ embedding method outperformed the local context operation by capturing global context, our EFA further improved mIoU by 0.8% with the lightweight model parameter and FLOPs. This indicates that our EFA module better models the global context.

4.4 Structural Analysis of our All-attention Decoder

Our decoder, a {3-2-1} structure, is the hierarchical structure with six EFT blocks that assigns more attention blocks to high-level semantic features. In Table 3 (b), we verified the effectiveness of our decoder structure compared with three cases. The case of {2-2-2} structure assigned two EFT blocks equally to all decoder stages. The cases of {1-2-3}, {1-4-1} and our {3-2-1} allocated more EFT blocks to the decoder stage-3, 2 and 1, respectively. As a result, our {3-2-1} structure assigning more attention to higher level features shows better

(a) Effectiveness of our EFA for the decoder

Operation	Params (M)	ADE20K GFLOPs	ADE20K mIoU(%)
DW Conv	4.5	5.1	40.7
Conv	6.6	6.0	39.9
w/ embedding	5.7	6.1	41.5
w/o embedding	4.9	**5.6**	**42.3**

(b) Ablation on the number of EFA at each decoder stage

Stage-1	Stage-2	Stage-3	Params (M)	ADE20K GFLOPs	ADE20K mIoU(%)
2	2	2	4.6	5.7	41.5
1	2	3	4.2	5.8	40.6
1	4	1	4.4	5.7	40.5
3	**2**	**1**	4.9	**5.6**	**42.3**

Table 3: Ablation studies of our all-attention decoder structure on the validation set of ADE20K. Our EFT encoder is used as the backbone.

$[r_E^1, r_E^2, r_E^3, r_E^4]$-$[r_D^1, r_D^2, r_D^3]$ Reduction ratio Train	Inference	Params (M)	ADE20K GFLOPs ↓	ADE20K mIoU (%) ↑	Cityscapes GFLOPs ↓	Cityscapes mIoU (%) ↑	COCO-Stuff GFLOPs ↓	COCO-Stuff mIoU (%) ↑
(a) EDAFormer-T with the different reduction ratio at inference.								
[8, 4, 2, 1]-[1, 2, 4]	[8, 4, 2, 1]-[1, 2, 4]†	4.9	5.6	42.3	151.7	78.7	5.6	40.3
	[8, 4, 2, 1]-[2, 4, 8]	4.9	5.3 (-5.4%)	42.2 (-0.1)	133.6 (-11.9%)	78.7 (-0.0)	5.3 (-5.4%)	40.3 (-0.0)
	[**16, 8, 2, 1**]-[**2, 4, 8**]	4.9	4.7 (-16.1%)	42.1 (-0.2)	94.9 (-37.4%)	78.7 (-0.0)	4.7 (-16.1%)	40.3 (-0.0)
	[16, 8, 4, 2]-[2, 4, 8]	4.9	4.1 (-26.8%)	41.3 (-1.0)	59.1 (-61.0%)	78.1 (-0.6)	4.1 (-26.8%)	39.1 (-1.2)
	[16, 8, 4, 2]-[2, 4, 8]⋆	4.9	4.1 (-26.8%)	42.1 (-0.2)	59.1 (-61.0%)	78.6 (-0.1)	4.1 (-26.8%)	40.2 (-0.1)
(b) EDAFormer-B with the different reduction ratio at inference.								
[8, 4, 2, 1]-[1, 2, 4]	[8, 4, 2, 1]-[1, 2, 4]†	29.4	32.0	49.0	605.9	81.6	32.0	45.9
	[8, 4, 2, 1]-[2, 4, 8]	29.4	31.3 (-2.2%)	48.9 (-0.1)	569.0 (-6.1%)	81.6 (-0.0)	31.3 (-2.2%)	45.8 (-0.1)
	[**16, 8, 2, 1**]-[**2, 4, 8**]	29.4	29.4 (-8.1%)	48.9 (-0.1)	452.9 (-25.3%)	81.6 (-0.0)	29.4 (-8.1%)	45.8 (-0.1)
	[16, 8, 4, 2]-[2, 4, 8]	29.4	26.6 (-16.9%)	48.3 (-0.7)	298.1 (-50.8%)	81.4 (-0.2)	26.6 (-16.9%)	45.0 (-0.9)
	[16, 8, 4, 2]-[2, 4, 8]⋆	29.4	26.6 (-16.9%)	48.7 (-0.3)	298.1 (-50.8%)	81.6 (-0.0)	26.6 (-16.9%)	45.7 (-0.2)

Table 4: Computation and performance of our model on three standard benchmarks. † indicates that the same reduction ratio is applied at training and inference. ⋆ indicates the fine-tuning. **Bold** is optimal inference reduction ratio for our EDAFormer.

performance of 0.8%, 1.7%, 1.8% mIoU compared to {2-2-2},{1-2-3}, and {1-4-1}, respectively. These results indicate that allocating the additional attention layers to the higher level features, which contain richer semantic information, is more effective for semantic segmentation performance.

4.5 Effectivness of our ISR in our EDAFormer

In Table 4, we verified the effectiveness of our Inference Spatial Reduction (ISR) method in the proposed EDAFormer-T and EDAFormer-B, and empirically found the optimal reduction ratio. At training, our EDAFormer was trained with the base setting of [8,4,2,1]-[1,2,4]. At inference, We experimented on applying our ISR to only decoder (*i.e.*[8,4,2,1]-[2,4,8]), part of the encoder-decoder (*i.e.*[16,8,2,1]-[2,4,8]), and all of the encoder-decoder (*i.e.*[16,8,4,2]-[2,4,8]). The setting of [16,8,2,1]-[2,4,8] showed the optimal performance for improving the computational efficiency compared to the accuracy degradation. Compared to EDAFormer-T with the base setting, EDAFormer-T with the optimal setting reduced the computation by 16.1%, 37.4% and 16.1% on ADE20K, Cityscapes and COCO-Stuff, respectively. The performance dropped by only 0.2% mIoU on ADE20K and did not drop on COCO-Stuff and Cityscapes. Furthermore, EDAFormer-B reduced the computation by 8.1% with only 0.1% mIoU degradation on ADE20K and COCO-Stuff, and reduced the computation by 25.3% without performance degradation on Cityscapes. These results indicate that our ISR

Method	$[r_E^1, r_E^2, r_E^3, r_E^4]$-$[r_D^1, r_D^2, r_D^3]$ Reduction ratio Train	Inference	COCO-Stuff GFLOPs	mIoU(%)
(a) Comparisons of our models with and without our ISR method				
EDAFormer-T				
w/o ISR	[16, 8, 2, 1]-[2, 4, 8]	[16, 8, 2, 1]-[2, 4, 8]	4.7	39.8
w/ ISR	[8, 4, 2, 1]-[1, 2, 4]	[16, 8, 2, 1]-[2, 4, 8]	4.7	**40.3**
EDAFormer-B				
w/o ISR	[16, 8, 2, 1]-[2, 4, 8]	[16, 8, 2, 1]-[2, 4, 8]	29.4	45.3
w/ ISR	[8, 4, 2, 1]-[1, 2, 4]	[16, 8, 2, 1]-[2, 4, 8]	29.4	**45.8**

Method	Reduction ratio $[r_E^1, r_E^2, r_E^3, r_E^4]$-$[r_D^1, r_D^2, r_D^3]$	Params (M)	Cityscapes GFLOPs	mIoU(%)
(b) Effectiveness of our EFA structure for our ISR				
w/ embedding	[8, 4, 2, 1]-[1, 2, 4]	5.7	153.5	78.7
	[8, 4, 2, 1]-[2, 4, 8]	5.7	134.7	78.5 (-0.2)
	[8, 4, 2, 1]-[3, 6, 9]	5.7	131.2	78.2 (-0.5)
	[8, 4, 2, 1]-[4, 8, 12]	5.7	130.0	77.9 (-0.8)
w/o embedding	[8, 4, 2, 1]-[1, 2, 4]	4.9	151.7	78.7
	[8, 4, 2, 1]-[2, 4, 8]	4.9	133.6	78.7 (-0.0)
	[8, 4, 2, 1]-[3, 6, 9]	4.9	130.3	78.7 (-0.0)
	[8, 4, 2, 1]-[4, 8, 12]	4.9	129.1	78.6 (-0.1)

Table 5: (a) Ablation for mIoU (%) performance comparisons of our models with and without our ISR method on COCO-Stuff. (b) Ablation for the effectiveness of our EFA structure for our ISR on Cityscapes *val.*

method is simple, yet significantly reduces the computational cost with little performance degradation. In addition, our method showed the impressive effectiveness by only adjusting the reduction ratio at the inference without fine-tuning. Our ISR is effective without the fine-tuning, but we trained the models with 40K iterations for fine-tuning to further compensate for performance degradation at higher reduction ratio of [16,8,4,2]-[2,4,8]. As a result, EDAFormer-T showed a 0.2% drop in mIoU on ADE20K, and 0.1% drops in mIoU on Cityscapes and COCO-Stuff. EDAFormer-B showed 0.3% and 0.2% drops in mIoU on ADE20K and COCO-Stuff, and no drop in mIoU on Cityscapes.

4.6 Comparison between the model with and without ISR.

In Table 5 (a), we compared our w/ ISR with w/o ISR, which used the same reduction ratio of [16,8,2,1]-[2,4,8] at both training and inference. Our EDAFormer with our ISR was trained with the reduction ratio of [8,4,2,1]-[1,2,4] and adjusted the ratio to [16,8,2,1]-[2,4,8] at inference. Despite the same computation at inference phase, the result with our ISR showed better mIoU than the case w/o ISR, with both 0.5% improvements for our EDAFormer-T and EDAFormer-B, respectively. Therefore, our model w/ ISR, which considers enough information of the key and value during training, can achieve better performance than the model that cannot consider enough information by reducing the resolution of the key and value during training.

4.7 Effectiveness of Embedding-Free Structure for ISR

To verify the effectiveness of our embedding-free structure for ISR. We experiment the ablated model that w/ embedding attention is adopt to our EFA position in all-attention decoder. We also compared with the ablated model (*i.e.*, w/ embedding) by applying our ISR to the decoder stages in Table 5 (b). The w/ embedding structure showed the gradual performance degradation as the reduction ratio increased, and the reduction ratio of [8,4,2,1]-[4,8,12] showed the performance decrease of 0.8% mIoU. However, our structure showed no performance degradation up to the reduction ratio of [8,4,2,1]-[3,6,9], and only a 0.1% drop in mIoU at the reduction ratio of [8,4,2,1]-[4,8,12]. This indicate that our w/o embedding structure is effective with proposed ISR method.

Method	Reduction ratio $[r_E^1, r_E^2, r_E^3, r_E^4]$-$[r_D^1, r_D^2, r_D^3]$	Cityscapes mIoU (%) ↑	FPS (img/s) ↑
(a) Comparison of different spatial reduction methods for our ISR			
w/o ISR	[8, 4, 2, 1]-[1, 2, 4]	78.7	10.2
Bipartite matching [1]	[11.2, 5.6, 2.8, 1.4]-[1.4, 2.8, 5.6]	78.7	10.5
Max pooling	[16, 8, 2, 1]-[2, 4, 8]	78.4	13.3
Overlapped pooling	[16, 8, 2, 1]-[2, 4, 8]	78.7	13.2
Average pooling	[16, 8, 2, 1]-[2, 4, 8]	78.7	13.3
(b) Inference speed improvement by increasing the reduction ratio			
Average pooling	[8, 4, 2, 1]-[1, 2, 4]	78.7	10.2
	[8, 4, 2, 1]-[2, 4, 8]	78.7	11.0 (+7.8%)
	[16, 8, 2, 1]-[2, 4, 8]	78.7	13.2 (+29.4%)
	[16, 8, 4, 2]-[2, 4, 8]	78.1	15.0 (+47.1%)

Table 6: (a) Performance and inference speed of our ISR with different spatial reduction methods. (b) Inference speed by increasing the reduction ratio.

Models	Params (M)	GFLOPs ↓	mIoU (%) ↑
CvT [53]	21.0	365.5	80.1
CvT + ISR	21.0	222.6 (-39.1%)	79.8 (-0.3)
MViT [60]	32.0	1435.6	80.5
MViT + ISR	32.0	838.0 (-41.6%)	80.3 (-0.2)
LVT [57]	5.0	132.1	79.6
LVT + ISR	5.0	86.1 (-34.8%)	79.5 (-0.1)
Swin [35]	36.2	272.2	79.7
Swin + ISR	36.2	208.0 (-23.6%)	79.0 (-0.7)
DaViT [17]	36.2	304.8	81.3
DaViT + ISR	36.2	242.0 (-20.6%)	80.9 (-0.4)
PVTv2 [51]	4.8	121.8	78.6
PVTv2 + ISR	4.8	63.4 (-47.9%)	78.3 (-0.3)
MiT [55]	4.9	117.4	78.2
MiT [55] + ISR	4.9	59.0 (-49.7%)	77.6 (-0.6)
SegFormer [55]	3.8	125.5	76.2
SegFormer + ISR	3.8	82.5 (-34.3%)	75.6 (-0.6)
FeedFormer [43]	4.5	107.5	77.9
FeedFormer + ISR	4.5	68.8 (-36.0%)	77.4 (-0.5)
EDAFormer (**Ours**)	4.9	151.7	78.7
EDAFormer + ISR (**Ours**)	4.9	94.9 (-37.4%)	78.7 (-0.0)

Table 7: Applying our ISR without finetuning to various transformer-based models on Cityscapes *val*.

4.8 Comparison of Spatial Reduction Methods for ISR

In Table 6 (a), We experimented to compare which method is better in terms of the mIoU and inference speed (FPS) for the key-value spatial reduction. The bipartite matching-based pooling had no mIoU degradation even though it was applied to every encoder-decoder stage. However, the bipartite matching can reduce maximum 50% of tokens, which corresponds to a reduction ratio of $r = 1.4$ ($\approx \sqrt{2}$). This is because it divides the tokens into two sets and merges them. In addition, this method has the additional latency caused by the matching algorithm. Therefore, the bipartite matching showed similar FPS compared to w/o ISR even though they reduce the computation of the attention. The max pooling showed a drop of 0.3% mIoU, and the overlapped pooling was slightly slower than the average pooling. Therefore, we adopted the average pooling method to reduce the tokens, which is a simple operation for general purposes and is most effective in terms of performance with inference speed.

4.9 Inference Speed Enhancement

In Table 6 (b), we represented the inference speed (FPS) comparisons of various reduction ratios. We measured the inference speed by using a single RTX 3090 GPU without any additional accelerating techniques. Compared to base setting, applying our ISR shows 29.4% and 47.1% FPS improvements in the reduction ratios of [16,8,2,1]-[2,4,8] and [16,8,4,2]-[2,4,8], respectively. The inference speed became faster as the computational cost was reduced by increasing the reduction ratio. These results indicate that the the computational reduction by our ISR leads to the improvement of the actual inference speed.

4.10 Applying ISR to Various Transformer-based Models

Our ISR can be universally applied not only to our EDAFormer, but also to other transformer-based models by using the additional spatial reduction at the

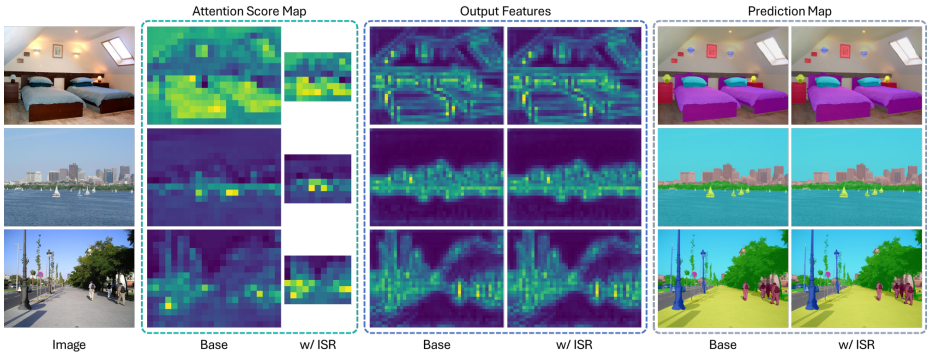

Fig. 4: Visualization of the attention score map, output features, and prediction map on ADE20K. 'Base' represents our EDAFormer trained with the base reduction ratio of [8,4,2,1]-[1,2,4]. 'w/ ISR' represents our EDAFormer applied our ISR method.

inference. To verify generalizability of our ISR, we applied ours to various models in Table 7. The transformer-based backbones are trained with our decoder for the semantic segmentation task. For the convolutional self-attention models (*i.e.*, CvT [53], MViT [60] and LVT [57]), our ISR significantly reduced computation by 34.8∼41.6% with 0.1∼0.3% performance degradation. Our method also showed the effective computational reduction with less performance degradation for window-based attention models (*i.e.*, Swin [35] and DaViT [17]), spatial reduction attention-based models (*i.e.*, PVTv2 [51] and MiT [55]) and segmentation models (*i.e.*, SegFormer [55] and FeedFormer [43]). The result for FeedFormer using the cross-attention decoder showed that our method is also effective in the cross-attention mechanism. These results indicate that our ISR framework can be effectively extended to various transformer-based architecture using different attention methods, and our EDAFormer is especially the optimized architecture for applying our ISR effectively.

4.11 Visualization of Features

Fig. 4 visualized the features and prediction maps of the EDAFormer-B decoder stage-2 before and after applying the ISR. Firstly, we visualized the attention score maps representing the similarity score between the query and key. When ISR was applied, the resolution of the attention score map was reduced because the resolution of the key was reduced. Compared to the similarity scores without applying the ISR, the similarity scores between the query and key applying the ISR were well maintained. In other words, the attention regions before and after applying ISR were similar, even though we reduce the key tokens rather than the attention score map. Therefore, this means that applying our ISR can maintain the semantic similarity scores in the global regions.

Secondly, we compared the output features after operating between the attention score map and values. Surprisingly, the output features before and after

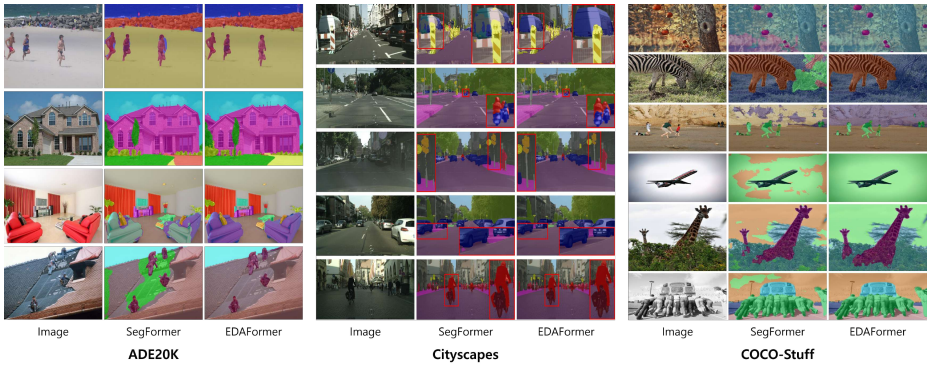

Fig. 5: Qualitative results on ADE20K, Cityscapes, and COCO-Stuff. Compared to SegFormer, the predictions of our EDAFormer are more precise for various categories.

applying ISR showed almost the same results. Therefore, these results indicate that the information obtained from the self-attention operation is maintained even though the spatial reduction is applied to the key and value in inference. Thirdly, when comparing the prediction maps, the results before and after applying the ISR are almost same. This means that the effect of ISR can be applied not only to the decoder stage-2, but also to the whole EDAFormer network.

4.12 Qualitative Results

In Fig. 5, we visualized our segmentation predictions on ADE20K, Cityscapes and COCO-Stuff, compared with the embedding-based transformer model (*i.e.* SegFormer [55]). Our EDAFormer better predicted the finer details near object boundaries. Our model also better segmented the large regions (*e.g.*, road, roof and truck) than SegFormer. Furthermore, our model predicted the objects of the same category (*e.g.*, sofa) that were far apart more precisely than SegFormer. This indicates that our embedding-free attention structure can capture enough global spatial information.

5 Conclusion

In this paper, we present an efficient transformer-based semantic segmentation model, EDAFormer, which leverages the proposed embedding-free attention module. The embedding-free attention structure can rethink the self-attention mechanism in the aspect of modeling the global context. In addition, we propose the novel inference spatial reduction framework for the efficiency, which changes the condition between train-inference phases. We hope that our attention mechanism and framework could further research efforts in exploring the lightweight and efficient transformer-based semantic segmentation model.

Acknowledgements

This work was supported by Samsung Electronics Co., Ltd (IO201218-08232-01) and the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (No. RS-2024-00414230) and MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2024-RS-2023-00260091) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation) and National Supercomputing Center with supercomputing resources including technical support(KSC-2023-CRE-0444).

References

1. Bolya, D., Fu, C.Y., Dai, X., Zhang, P., Feichtenhofer, C., Hoffman, J.: Token merging: Your vit but faster. arXiv preprint arXiv:2210.09461 (2022)
2. Bousselham, W., Thibault, G., Pagano, L., Machireddy, A., Gray, J., Chang, Y.H., Song, X.: Efficient self-ensemble for semantic segmentation. arXiv preprint arXiv:2111.13280 (2021)
3. Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1209–1218 (2018)
4. Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs (2017)
5. Chen, X., Liu, Z., Tang, H., Yi, L., Zhao, H., Han, S.: Sparsevit: Revisiting activation sparsity for efficient high-resolution vision transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2061–2070 (2023)
6. Chen, Y., Dai, X., Chen, D., Liu, M., Dong, X., Yuan, L., Liu, Z.: Mobile-former: Bridging mobilenet and transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5270–5279 (2022)
7. Chen, Z., Xie, L., Niu, J., Liu, X., Wei, L., Tian, Q.: Visformer: The vision-friendly transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 589–598 (2021)
8. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1290–1299 (2022)
9. Cheng, B., Schwing, A., Kirillov, A.: Per-pixel classification is not all you need for semantic segmentation pp. 17864–17875 (2021)
10. Cho, Y., Kang, S.: Class attention transfer for semantic segmentation. In: 2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS). pp. 41–45. IEEE (2022)
11. Cho, Y., Yu, H., Kang, S.J.: Cross-aware early fusion with stage-divided vision and language transformer encoders for referring image segmentation. IEEE Transactions on Multimedia (2023)
12. Chu, X., Tian, Z., Wang, Y., Zhang, B., Ren, H., Wei, X., Xia, H., Shen, C.: Twins: Revisiting the design of spatial attention in vision transformers pp. 9355–9366 (2021)

13. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3213–3223 (2016)
14. Dai, Z., Liu, H., Le, Q.V., Tan, M.: Coatnet: Marrying convolution and attention for all data sizes pp. 3965–3977 (2021)
15. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
16. Ding, M., Xiao, B., Codella, N., Luo, P., Wang, J., Yuan, L.: Object-contextual representations for semantic segmentation. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16. pp. 173–190 (2020)
17. Ding, M., Xiao, B., Codella, N., Luo, P., Wang, J., Yuan, L.: Davit: Dual attention vision transformers. In: Proceedings of the European conference on computer vision (ECCV). pp. 74–92 (2022)
18. Dong, X., Bao, J., Chen, D., Zhang, W., Yu, N., Yuan, L., Chen, D., Guo, B.: Cswin transformer: A general vision transformer backbone with cross-shaped windows. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12124–12134 (2022)
19. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
20. Gong, C., Wang, D., Li, M., Chen, X., Yan, Z., Tian, Y., Liu, Q., Chandra, V.: Nasvit: Neural architecture search for efficient vision transformers with gradient conflict aware supernet training. In: International Conference on Learning Representations (2021)
21. Graham, B., El-Nouby, A., Touvron, H., Stock, P., Joulin, A., Jégou, H., Douze, M.: Levit: a vision transformer in convnet's clothing for faster inference. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 12259–12269 (2021)
22. Guo, J., Han, K., Wu, H., Tang, Y., Chen, X., Wang, Y., Xu, C.: Cmt: Convolutional neural networks meet vision transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12175–12185 (2022)
23. Guo, M.H., Lu, C.Z., Hou, Q., Liu, Z., Cheng, M.M., Hu, S.M.: Segnext: Rethinking convolutional attention design for semantic segmentation. arXiv preprint arXiv:2209.08575 (2022)
24. Han, K., Xiao, A., Wu, E., Guo, J., XU, C., Wang, Y.: Transformer in transformer pp. 15908–15919 (2021)
25. Hatamizadeh, A., Heinrich, G., Yin, H., Tao, A., Alvarez, J.M., Kautz, J., Molchanov, P.: Fastervit: Fast vision transformers with hierarchical attention. In: The Twelfth International Conference on Learning Representations (2024), https://openreview.net/forum?id=kB4yBiNmXX
26. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
27. Kang, B., Moon, S., Cho, Y., Yu, H., Kang, S.J.: Metaseg: Metaformer-based global contexts-aware network for efficient semantic segmentation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 434–443 (2024)

28. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25** (2012)
29. Li, L., et al.: Semantic hierarchy-aware segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)
30. Li, Y., Yuan, G., Wen, Y., Hu, E., Evangelidis, G., Tulyakov, S., Wang, Y., Ren, J.: Efficientformer: Vision transformers at mobilenet speed. arXiv preprint arXiv:2206.01191 (2022)
31. Lin, W., Wu, Z., Chen, J., Huang, J., Jin, L.: Scale-aware modulation meet transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6015–6026 (2023)
32. Liu, H., Dai, Z., So, D., Le, Q.V.: Pay attention to mlps. Advances in Neural Information Processing Systems **34**, 9204–9215 (2021)
33. Liu, H., Jiang, X., Li, X., Bao, Z., Jiang, D., Ren, B.: Nommer: Nominate synergistic context in vision transformer for visual recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12073–12082 (2022)
34. Liu, Z., Hu, H., Lin, Y., Yao, Z., Xie, Z., Wei, Y., Ning, J., Cao, Y., Zhang, Z., Dong, L., Wei, F., Guo, B.: Swin transformer v2: Scaling up capacity and resolution. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12009–12019 (2022)
35. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
36. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11976–11986 (2022)
37. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
38. Lu, C., de Geus, D., Dubbelman, G.: Content-aware token sharing for efficient semantic segmentation with vision transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23631–23640 (2023)
39. Mehta, S., Rastegari, M.: Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer. arXiv preprint arXiv:2110.02178 (2021)
40. Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12179–12188 (2021)
41. Rao, Y., Zhao, W., Liu, B., Lu, J., Zhou, J., Hsieh, C.J.: Dynamicvit: Efficient vision transformers with dynamic token sparsification pp. 13937–13949 (2021)
42. RONNEBERGER, O., FISCHER, P., BROX, T.: U-net: Convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference. pp. 234–241 (2021)
43. Shim, J.h., Yu, H., Kong, K., Kang, S.J.: Feedformer: revisiting transformer decoder for efficient semantic segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 2263–2271 (2023)
44. Tan, M., Le, Q.: Efficientnetv2: Smaller models and faster training. In: International conference on machine learning. pp. 10096–10106. PMLR (2021)

45. Tang, Q., Zhang, B., Liu, J., Liu, F., Liu, Y.: Dynamic token pruning in plain vision transformers for semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 777–786 (2023)
46. Touvron, H., Bojanowski, P., Caron, M., Cord, M., El-Nouby, A., Grave, E., Izacard, G., Joulin, A., Synnaeve, G., Verbeek, J., et al.: Resmlp: Feedforward networks for image classification with data-efficient training. IEEE Transactions on Pattern Analysis and Machine Intelligence (2022)
47. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International conference on machine learning. pp. 10347–10357. PMLR (2021)
48. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
49. Wang, W., et al.: Exploring cross-image pixel contrast for semantic segmentation. In: ICCV. pp. 7303–7313 (2021)
50. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 568–578 (2021)
51. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pvt v2: Improved baselines with pyramid vision transformer. Computational Visual Media **8**(3), 415–424 (2022)
52. Wightman, R., Touvron, H., Jégou, H.: Resnet strikes back: An improved training procedure in timm. arXiv preprint arXiv:2110.00476 (2021)
53. Wu, H., Xiao, B., Codella, N., Liu, M., Dai, X., Yuan, L., Zhang, L.: Cvt: Introducing convolutions to vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22–31 (2021)
54. Wu, Y.H., Liu, Y., Zhan, X., Cheng, M.M.: P2t: Pyramid pooling transformer for scene understanding (2022)
55. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. Advances in Neural Information Processing Systems **34**, 12077–12090 (2021)
56. Yan, H., Wu, M., Zhang, C.: Multi-scale representations by varing window attention for semantic segmentation. In: The Twelfth International Conference on Learning Representations (2024), https://openreview.net/forum?id=lAhWGOkpSR
57. Yang, C., Wang, Y., Zhang, J., Zhang, H., Wei, Z., Lin, Z., Yuille, A.: Lite vision transformer with enhanced self-attention. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11998–12008 (2022)
58. Yu, H., Shim, J.h., Kwak, J., Song, J.W., Kang, S.J.: Vision transformer-based retina vessel segmentation with deep adaptive gamma correction. In: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 1456–1460. IEEE (2022)
59. Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., Yan, S.: Metaformer is actually what you need for vision. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10819–10829 (2022)
60. Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z.H., Tay, F.E., Feng, J., Yan, S.: Multiscale vision transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6824–6835 (2021)
61. Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z.H., Tay, F.E., Feng, J., Yan, S.: Tokens-to-token vit: Training vision transformers from scratch on imagenet. In:

Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 558–567 (2021)
62. Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z.H., Tay, F.E., Feng, J., Yan, S.: Tokens-to-token vit: Training vision transformers from scratch on imagenet. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 558–567 (2021)
63. Zhang, Q., Yang, Y.B.: Rest: An efficient transformer for visual recognition pp. 15475–15485 (2021)
64. Zhang, Q., Yang, Y.B.: Rest v2: simpler, faster and stronger pp. 36440–36452 (2022)
65. Zhang, W., Huang, Z., Luo, G., Chen, T., Wang, X., Liu, W., Yu, G., Shen, C.: Topformer: Token pyramid transformer for mobile semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12083–12093 (2022)
66. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6881–6890 (2021)
67. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ade20k dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 633–641 (2017)
68. Zhou, T., et al.: Rethinking semantic segmentation: A prototype view. In: CVPR. pp. 2582–2593 (2022)