Tensor Low-Rank Reconstruction for Semantic Segmentation

Wanli Chen[1], Xinge Zhu[1], Ruoqi Sun[2], Junjun He[2,3], Ruiyu Li[4], Xiaoyong Shen[4], and Bei Yu[1]

[1] The Chinese University of Hong Kong
{wlchen,byu}@cse.cuhk.edu.hk,zx018@ie.cuhk.edu.hk
[2] Shanghai Jiao Tong University
ruoqisun7@sjtu.edu.cn
[3] ShenZhen Key Lab of Computer Vision and Pattern Recognition, SIAT-SenseTime Joint Lab, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
hejunjun@sjtu.edu.cn
[4] SmartMore
{ryli,xiaoyong}@smartmore.com

Abstract. Context information plays an indispensable role in the success of semantic segmentation. Recently, non-local self-attention based methods are proved to be effective for context information collection. Since the desired context consists of spatial-wise and channel-wise attentions, 3D representation is an appropriate formulation. However, these non-local methods describe 3D context information based on a 2D similarity matrix, where space compression may lead to channel-wise attention missing. An alternative is to model the contextual information directly without compression. However, this effort confronts a fundamental difficulty, namely the high-rank property of context information. In this paper, we propose a new approach to model the 3D context representations, which not only avoids the space compression but also tackles the high-rank difficulty. Here, inspired by tensor canonical-polyadic decomposition theory (*i.e*, a high-rank tensor can be expressed as a combination of rank-1 tensors.), we design a low-rank-to-high-rank context reconstruction framework (*i.e*, RecoNet). Specifically, we first introduce the tensor generation module (TGM), which generates a number of rank-1 tensors to capture fragments of context feature. Then we use these rank-1 tensors to recover the high-rank context features through our proposed tensor reconstruction module (TRM). Extensive experiments show that our method achieves state-of-the-art on various public datasets. Additionally, our proposed method has more than 100 times less computational cost compared with conventional non-local-based methods.

Keywords: Semantic Segmentation, Low-Rank Reconstruction, Tensor Decomposition

1 Introduction

Semantic segmentation aims to assign the pixel-wise predictions for the given image, which is a challenging task requiring fine-grained shape, texture and cate-

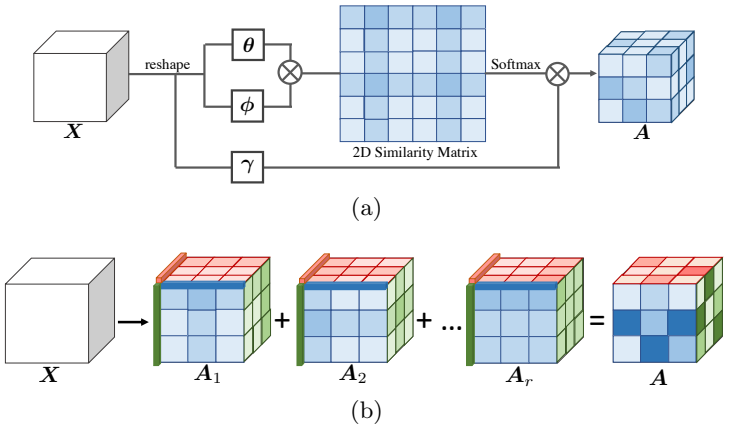


Fig. 1: (a) Non-local vs. (b) our proposed RecoNet, which is based on tensor low-rank reconstruction. Note that 2D similarity matrix exists in non-local based methods and our RecoNet is formed with all 3D tensors.

gory recognition. The pioneering work, fully convolutional networks (FCN) [31], explores the effectiveness of deep convolutional networks in segmentation task. Recently, more work achieves great progress from exploring the contextual information [1, 4, 5, 25, 33, 50], in which non-local based methods are the recent mainstream [16, 48, 51]. These methods model the context representation by rating the element-wise importance for contextual tensors. However, the context features obtained from this line lack of channel-wise attention, which is a key component of context. Specifically, for a typical non-local block, the 2D similarity map $\boldsymbol{A} \in \mathbb{R}^{HW \times HW}$ is generated by the matrix multiplication of two inputs with dimension of $H \times W \times C$ and $C \times H \times W$, respectively. It is noted that the channel dimension C is eliminated during the multiplication, which implies that only the spatial-wise attention is represented while the channel-wise attention is compressed. Therefore, these non-local based methods could collect fine-grained spatial context features but may sacrifice channel-wise context attention.

An intuitive idea tackling this issue is to construct the context directly instead of using the 2D similarity map. Unfortunately, this approach confronts a fundamental difficulty because of the high-rank property of context features [48]. That is, the context tensor should be high-rank to have enough capacity since contexts vary from image to image and this large diversity cannot be well-represented by very limited parameters.

Inspired by tensor canonical-polyadic decomposition theory [20], *i.e.*, a high-rank tensor can be expressed as a combination of rank-1 tensors, we propose a new approach of modeling high-rank contextual information in a progressive manner without channel-wise space compression. We show the workflow of non-local networks and RecoNet in Fig. 1. The basic idea is to first use a series of low-rank tensors to collect fragments of context features and then build them up

to reconstruct fine-grained context features. Specifically, our proposed framework consists of two key components, rank-1 tensor generation module (TGM) and high-rank tensor reconstruction module (TRM). Here, TGM aims to generate the rank-1 tensors in channel, height and width directions, which explore the context features in different views with low-rank constraints. TRM adopts tensor canonical-polyadic (CP) reconstruction to reconstruct the high-rank attention map, in which the co-occurrence contextual information is mined based on the rank-1 tensors from different views. The cooperation of these two components leads to the effective and efficient high-rank context modeling.

We tested our method on five public datasets. On these experiments, the proposed method consistently achieves the state-of-the-art, especially for PASCAL-VOC12 [13], RecoNet reaches the **top-1** performance. Furthermore, by incorporating the simple and clean low-rank features, our whole model has less computation consumption (more than **100** times lower than non-local) compared to other non-local based context modeling methods.

The contributions of this work mainly lie in three aspects:

- Our studies reveal a new path to the context modeling, namely, context reconstruction from low-rank to high-rank in a progressive way.
- We develop a new semantic segmentation framework RecoNet, which explores the contextual information through tensor CP reconstruction. It not only keeps both spatial-wise and channel-wise attentions, but also deals with high-rank difficulty.
- We conduct extensive experiments to compare the proposed methods with others on various public datasets, where it yields notable performance gains. Furthermore, RecoNet also has less computation cost, *i.e*, more than **100** times smaller than non-local based methods.

2 Related Work

Tensor Low-rank Representation. According to tensor decomposition theory [20], a tensor can be represented by the linear combination of series of low-rank tensors. The reconstruction results of these low-rank tensors are the principal components of original tensor. Therefore, tensor low-rank representation is widely used in computer vision task such as convolution speed-up [21] and model compression [45]. There are two tensor decomposition methods: Tuker decomposition and CP decompostion [20]. For the Tuker decomposition, the tensor is decomposed into a set of matrices and one core tensor. If the core tensor is diagonal, then Tuker decomposition degrades to CP decomposition. For the CP decomposition, the tensor is represented by a set of rank-1 tensors (vectors). In this paper, we apply this theory for *reconstruction*, namely reconstructing high-rank contextual tensor from a set of rank-1 context fragments.

Self-Attention in Computer Vision. Self attention is firstly proposed in natural language processing (NLP) [8,10,37,43]. It serves as a global encoding method that can merge long distance features. This property is also important to computer vision tasks. Hu *et al.* propose SE-Net [18], exploiting channel

information for better image classification through channel wise attention. Woo *et al.* propose CBAM [39] that combines channel-wise attention and spatial-wise attention to capture rich feature in CNN. Wang *et al.* propose non-local neural network [38]. It catches long-range dependencies of a featuremap, which breaks the receptive field limitation of convolution kernel.

Context Aggregation in Semantic Segmentation. Context information is so important for semantic segmentation and many researchers pay their attention to explore the context aggregation. The initial context harvesting method is to increase receptive fields such as FCN [31], which merges feature of different scales. Then feature pyramid methods [4,5,50] are proposed for better context collection. Although feature pyramid collects rich context information, the contexts are not gathered adaptively. In other words, the importance of each element in contextual tensor is not discriminated. Self-attention-based methods are thus proposed to overcome this problem, such as EncNet [47], PSANet [51], APCNet [16], and CFNet [48]. Researchers also propose some efficient self-attention methods such as EMANet [22], CCNet [19], A^2Net [6], which have lower computation consumption and GPU memory occupation. However, most of these methods suffer from channel-wise space compression due to the 2D similarity map. Compared to these works, our method differs essentially in that it uses the 3D low-rank tensor reconstruction to catch long-range dependencies without sacrificing channel-wise attention.

3 Methodology

3.1 Overview

The semantic information prediction from an image is closely related to the context information. Due to the large varieties of context, a high-rank tensor is required for the context feature representation. However, under this constraint, modeling the context features directly means a huge cost. Inspired by the CP decomposition theory, although the context prediction is a high-rank problem, we can separate it into a series of low-rank problems and these low-rank problems are easier to deal with. Specifically, we do not predict context feature directly, instead, we generate its fragments. Then we build up a complete context feature using these fragments. The low-rank to high-rank reconstruction strategy not only maintains 3D representation (for both channel-wise and spatial-wise), but also tackles with the high-rank difficulty.

The pipeline of our model is shown in Fig. 2, which consists of low-rank tensor generation module (TGM), high-rank tensor reconstruction module (TRM), and global pooling module (GPM) to harvest global context in both spatial and channel dimensions. We upsample the model output using bilinear interpolation before semantic label prediction.

In our implementation, multiple low-rank perceptrons are used to deal with the high-rank problem, by which we learn parts of context information (*i.e*, context fragments). We then build the high-rank tensor via tensor reconstruction theory [20].

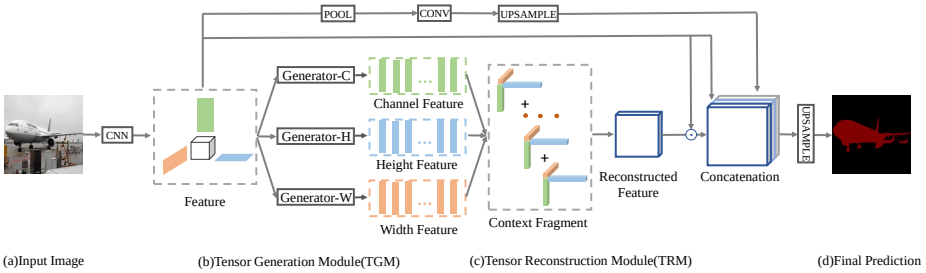


Fig. 2: The pipeline of our framework. Two major components are involved, *i.e*, Tensor Generation Module (TGM) and Tensor Reconstruction Module (TRM). TGM peroforms the low-rank tensor generation while TRM achieves the high-rank tensor reconstruction via CP construction theory.

Formulation: Assuming we have $3r$ vectors in C/H/W directions $\boldsymbol{v}_{ci} \in \mathbb{R}^{C\times1\times1}$, $\boldsymbol{v}_{hi} \in \mathbb{R}^{1\times H\times1}$ and $\boldsymbol{v}_{wi} \in \mathbb{R}^{1\times1\times W}$, where $i \in r$ and r is the tensor rank. These vectors are the CP decomposed fragments of $\boldsymbol{A} \in \mathbb{R}^{C\times H\times W}$, then tensor CP rank-r reconstruction is defined as:

$$\boldsymbol{A} = \sum_{i=1}^{r} \lambda_i \boldsymbol{v}_{ci} \otimes \boldsymbol{v}_{hi} \otimes \boldsymbol{v}_{wi}, \tag{1}$$

where λ_i is a scaling factor.

3.2 Tensor Generation Module

In this section, we first provide some basic definitions and then show how to derive the low-rank tensors from the proposed module.

Context Fragments. We define context fragments as the outputs of the tensor generation module, which indicates some rank-1 vectors $\boldsymbol{v}_{ci}$, $\boldsymbol{v}_{hi}$ and $\boldsymbol{v}_{wi}$ (as defined in previous part) in the channel, the height and the width directions. Every context fragment contains a part of context information.

Feature Generator. We define three feature generators: Channel Generator, Height Generator and Width Generator. Each generator is composed of Pool-Conv-Sigmoid sequence. Global pooling is widely used in previous works [29, 50] as the global context harvesting method. Similarly, here we use global average pooling in feature generators, obtaining the global context representation in C/H/W directions.

Context Fragments Generation. In order to learn fragments of context information across the three directions, we apply channel, height and width generator on the top of input feature. We repeat this process r times obtaining $3r$ learnable vectors $\boldsymbol{v}_{ci} \in \mathbb{R}^{C\times1\times1}$, $\boldsymbol{v}_{hi} \in \mathbb{R}^{1\times H\times1}$ and $\boldsymbol{v}_{wi} \in \mathbb{R}^{1\times1\times W}$, where $i \in r$. All vectors are generated using independent convolution kernels. Each of them learns a part of context information and outputs as context fragment. The TGM is shown in Fig. 3.

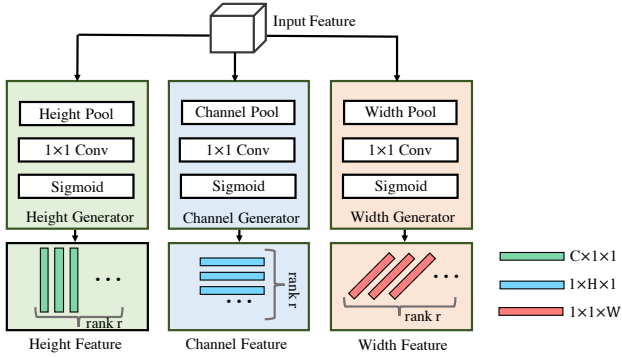


Fig. 3: Tensor Generation Module. Channel Pool, Height Pool and Width Pool are all global average pooling.

Non-linearity in TGM. Recalling that TGM generates $3r$ rank-1 tensors and these tensors are activated by Sigmoid function, which re-scales the values in context fragments to [0, 1]. We add the non-linearity for two reasons. Firstly, each re-scaled element can be regarded as the weight of a certain kind of context feature, which satisfy the definition of attention. Secondly, all the context fragments shall not be linear dependent so that each of them can represent different information.

3.3 Tensor Reconstruction Module

In this part, we introduce the context feature reconstruction and aggregation procedure. The entire reconstruction process is clean and simple, which is based on Equation (1). For a better interpretation, we first introduce the context aggregation process.

Context Aggregation. Different from previous works that only collect spatial or channel attention [47, 51], we collect attention distribution in both directions simultaneously. The goal of TRM is to obtain the 3D attention map $\boldsymbol{A} \in \mathbb{R}^{C\times H\times W}$ which keeps response in both spatial and channel attention. After that, context feature is obtained by element-wise product. Specifically, given an input feature $\boldsymbol{X} = \{x_1, x_2, \ldots, x_{CHW}\}$ and a context attention map $\boldsymbol{A} = \{a_1, a_2, \ldots, a_{CHW}\}$, the fine-grained context feature $\boldsymbol{Y} = \{y_1, y_2, \ldots, y_{CHW}\}$ is then given by:

$$\boldsymbol{Y} = \boldsymbol{A} \cdot \boldsymbol{X} \iff y_i = a_i \cdot x_i, i \in CHW. \tag{2}$$

In this process, every $a_i \in \boldsymbol{A}$ represents the extent that $x_i \in \boldsymbol{X}$ be activated.

Low-rank Reconstruction. The tensor reconstruction module (TRM) tackles the high-rank property of context feature. The full workflow of TRM is shown in Fig. 4, which consists of two steps, *i.e*, sub-attention map aggregation and global context feature reconstruction. Firstly, three context fragments $\boldsymbol{v}_{c1} \in \mathbb{R}^{C\times 1\times 1}$, $\boldsymbol{v}_{h1} \in \mathbb{R}^{1\times H\times 1}$ and $\boldsymbol{v}_{w1} \in \mathbb{R}^{1\times 1\times W}$ are synthesized into a rank-1 sub-attention map $\boldsymbol{A}_1$. This sub-attention map represents a part of 3D context feature, and we

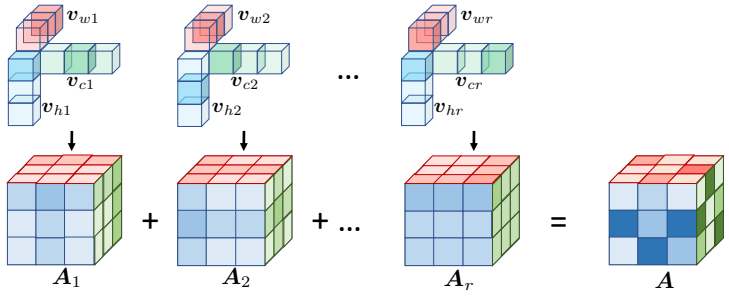


Fig. 4: Tensor Reconstruction Module (TRM). The pipeline of TRM consists of two main steps, *i.e*, sub-attention map generation and global context reconstruction. The processing from top to bottom (see ↓) indicates the sub-attention map generation from three dimensions (channel / height / width). The processing from left to right (see $\boldsymbol{A}_1 + \boldsymbol{A}_2 + \cdots + \boldsymbol{A}_r = \boldsymbol{A}$) denotes the global context reconstruction from low-rank to high-rank.

will show the visualization of some $\boldsymbol{A}_i, i \in [1, r]$ in experimental result part. Then, other context fragments are reconstructed following the same process. After that we aggregate these sub-attention maps using weighted mean:

$$\boldsymbol{A} = \sum_{i=1}^{r} \lambda_i \boldsymbol{A}_i. \tag{3}$$

Here $\lambda_i \in (0, 1)$ is a learnable normalize factor. Although each sub-attention map represents low-rank context information, the combination of them becomes a high-rank tensor. The fine-grained context features in both spatial and channel dimensions are obtained after Equation (3) and Equation (2).

3.4 Global Pooling Module

Global pooling module (GPM) is commonly used in previous work [48, 50]. It is composed of a global average pooling operation followed with a 1×1 convolution. It harvests global context in both spatial and channel dimensions. In our proposed model, we apply GPM for the further boost of network performance.

3.5 Network Details

We use ResNet [17] as our backbone and apply dilation strategy to the output of Res-4 and Res-5 of it. Then, the output stride of our proposed network is 8. The output feature of Res-5 block is marked as X. TGM+TRM and GPM are then added on the top of X. Following previous works [47, 50], we also use auxiliary loss after Res-4 block. We set the weight α to 0.2. The total loss $\mathcal{L}$ is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{main} + \alpha \mathcal{L}_{aux}. \tag{4}$$

Finally, we concatenate X with the context featuremap generated by TGM+TRM and the global context generated by GPM to make the final prediction.

3.6 Relation to Previous Approaches

Compared with non-local and its variants that explore the pairwise relationship between pixels, the proposed method is essentially unary attention. Unary attention has been widely used in image classification such as SENet [18] and CBAM [39]. It is also broadly adopted in semantic segmentation such as DFN [44] and EncNet [47]. Apparently, SENet is the simplest formation of RecoNet. The 3D attention map of SENet $\boldsymbol{A}_{SE} \in \mathbb{R}^{C\times H\times W}$ is as Formula (5):

$$\begin{aligned} \boldsymbol{A}_{SE} &= \boldsymbol{v}_c \otimes \boldsymbol{v}_h \otimes \boldsymbol{v}_w, \\ \boldsymbol{v}_h &= \boldsymbol{e}, \\ \boldsymbol{v}_w &= \boldsymbol{e}^\top, \\ \boldsymbol{e} &= \{1, 1, 1, \dots 1\}. \end{aligned} \tag{5}$$

RecoNet degenerates to SENet by setting tensor rank $r = 1$. Meanwhile, $\boldsymbol{v}_h = \boldsymbol{e}$ and $\boldsymbol{v}_w = \boldsymbol{e}^\top$. From Formula (5), it is observed that the weights in H and W directions are the same, which implies that SENet only harvests channel attention while sets the same weights in spatial domain. EncNet [47] is the updated version of SENet, which also uses the same spatial weights. Different spatial weights are introduced in CBAM, which extends Formula (5) to Equation (6).

$$\boldsymbol{A}_{CBAM} = \boldsymbol{v}_c \otimes \boldsymbol{v}_{h,w}, \quad \boldsymbol{v}_{h,w} \in \mathbb{R}^{1\times H\times W}. \tag{6}$$

Here $\boldsymbol{A}_{CBAM} \in \mathbb{R}^{C\times H\times W}$ is the 3D attention map of CBAM. The spatial attention is considered in CBAM. However, single rank-1 tensor $\boldsymbol{A}_{CBAM}$ can not represent complicated context information. Considering an extreme case, the spatial attention is CP-decomposed into 2 rank-1 tensors $\boldsymbol{v}_h \in \mathbb{R}^{1\times H\times 1}$ and $\boldsymbol{v}_w \in \mathbb{R}^{1\times 1\times W}$. Then, $\boldsymbol{A}_{CBAM}$ becomes a sub-attention map of RecoNet.

Simple but effective is the advantage of unary attentions, but they are also criticized for not being able to represent complicated features or for being able to represent features only in one direction (spatial/channel). RecoNet not only takes the advantage of simplicity and effectiveness from unary attention, but also delivers comprehensive feature representations from multi-view (*i.e*, spatial and channel dimension).

4 Experiments

Many experiments are carried out in this section. We use five datasets: PASCAL-VOC12, PASCAL-Context, COCO-Stuff, ADE20K and SIFT-FLOW to test the performance of RecoNet.

4.1 Implementation Details

RecoNet is implemented using Pytorch [32]. Following previous works [14, 47], synchronized batch normalization is applied. The learning rate scheduler is $lr = base\_lr \times (1 - \frac{iter}{total\_iters})^{power}$. We set $base\_lr$ to 0.001 for PASCAL-VOC12, PASCAL-Context and COCO-Stuff datasets. The $base\_lr$ for ADE20K and SIFT-FLOW is 0.01 and 0.0025. Here we set $power$ to 0.9. SGD optimizer is applied with weight decay 0.0001 and momentum 0.9. We train ADE20K and COCO-Stuff for 120 epochs and 180 epochs respectively. For other datasets, we train 80 epochs. The batch size we set for all datasets is 16 and all input images are randomly cropped into 512×512 before putting into neural network. The data augmentation method we use is the same with previous works [47, 50]. Specifically, we randomly flip and scale the input images (0.5 to 2).

We use multi-scale and flip evaluation with input scales [0.75, 1, 1.25, 1.5, 1.75, 2.0] times of original scale. The evaluation metrics we use is mean Intersection-over-Union (mIoU).

4.2 Results on Different Datasets

PASCAL-VOC12. We first test RecoNet using PASCAL-VOC12 [13] dataset, a golden benchmark of semantic segmentation, which includes 20 object categories and one background class. The dataset contains 10582, 1449, 1456 images for training, validation and testing. Our training set contains images from PASCAL augmentation dataset. The results are shown in Table 1. RecoNet reaches 85.6% mIoU, surpassing current best algorithm using ResNet-101 by 1.2%, which is a large margin.

Following previous work [14–16, 47, 48], we use COCO-pertained model during training. We first train our model on MS-COCO [27] dataset for 30 epochs, where the initial learning rate is set to 0.004. Then the model is fine-tuned on PASCAL augmentation training set for another 80 epochs. Finally, we fine-tune our model on original VOC12 train+val set for extra 50 epochs and the initial lr is set to 1e-5. The results in Table 2 show that RecoNet-101 outperforms current state-of-the-art algorithms with the same backbone. Moreover, RecoNet also exceeds state-of-the-art methods that use better backbone such as Xception [7]. By applying ResNet-152 backbone, RecoNet reaches 89.0% mIoU without adding extra data. The result is now in the **1st** place of the PASCAL-VOC12 challenge[5].

PASCAL-Context. [42] is a densely labeled scene parsing dataset includes 59 object and stuff classes plus one background class. It contains 4998 images for training and 5105 images for testing. Following previous works [16, 47, 48], we evaluate the dataset with background class (60 classes in total). The results are shown in Table 3. RecoNet performs better than all previous approaches that use non-local block such as CFNet and DANet, which implies that our proposed context modeling method is better than non-local block.

[5] http://host.robots.ox.ac.uk:8080/anonymous/PXWAVA.html

Table 1: Results on PASCAL-VOC12 w/o COCO-pretrained model

	FCN [31]	PSPNet [50]	EncNet [47]	APCNet [16]	CFNet [48]	DMNet [15]	RecoNet
aero	76.8	91.8	94.1	95.8	95.7	**96.1**	93.7
bike	34.2	71.9	69.2	75.8	71.9	**77.3**	66.3
bird	68.9	94.7	**96.3**	84.5	95.0	94.1	95.6
boat	49.4	71.2	**76.7**	76.0	76.3	72.8	72.8
bottle	60.3	75.8	86.2	80.6	82.8	78.1	**87.4**
bus	75.3	95.2	96.3	96.9	94.8	**97.1**	94.5
car	74.7	89.9	90.7	90.0	90.0	**92.7**	92.6
cat	77.6	95.9	94.2	96.0	95.9	96.4	**96.5**
chair	21.4	39.3	38.8	42.0	37.1	39.8	**48.4**
cow	62.5	90.7	90.7	93.7	92.6	91.4	**94.5**
table	46.8	71.7	73.3	75.4	73.0	75.5	**76.6**
dog	71.8	90.5	90.0	91.6	93.4	92.7	**94.4**
horse	63.9	94.5	92.5	95.0	94.6	95.8	**95.9**
mbike	76.5	88.8	88.8	90.5	89.6	91.0	**93.8**
person	73.9	89.6	87.9	89.3	88.4	90.3	**90.4**
plant	45.2	72.8	68.7	75.8	74.9	76.6	**78.1**
sheep	72.4	89.6	92.6	92.8	**95.2**	94.1	93.6
sofa	37.4	**64**	59.0	61.9	63.2	62.1	63.4
train	70.9	85.1	86.4	88.9	**89.7**	85.5	88.6
tv	55.1	76.3	73.4	79.6	78.2	77.6	**83.1**
mIoU	62.2	82.6	82.9	84.2	84.2	84.4	**85.6**

COCO-Stuff. [2] is a challenging dataset which includes 171 object and stuff categories. The dataset provides 9000 images for training and 1000 images for testing. The outstanding performance of RecoNet (as shown in Table 4) illustrates that the context tensor we modeled has enough capacity to represent complicated context features.

SIFT-Flow. [28] is a dataset that focuses on urban scene, which consists of 2488 images in training set and 500 images for testing. The resolution of images is 256×256 and 33 semantic classes are annotated with pixel-level labels. The result in Table 5 shows that the proposed RecoNet outperforms previous state-of-the-art methods.

ADE20K. [53] is a large scale scene parsing dataset which contains 25K images annotated with 150 semantic categories. There are 20K training images, 2K validation images and 3K test images. The experimental results are shown in Table 6. RecoNet shows better performance than non-local based methods such as CCNet [19]. The superiority on result means RecoNet can collect richer context information.

4.3 Ablation Study

In this section, we perform the thorough ablation experiments to investigate the effect of different components in our method and the effect of different rank number. These experiments provide more insights of our proposed method. The

Table 2: Results on PASCAL-VOC w. COCO-pretrained model

Method	Backbone	mIoU
CRF-RNN [52]		74.7
DPN [30]		77.5
Piecewise [26]		78.0
ResNet38 [41]		84.9
PSPNet [50]	ResNet-101	85.4
DeepLabv3 [4]	ResNet-101	85.7
EncNet [47]	ResNet-101	85.9
DFN [44]	ResNet-101	86.2
CFNet [48]	ResNet-101	87.2
EMANet [22]	ResNet-101	87.7
DeeplabV3+ [5]	Xception	87.8
DeeplabV3+ [5]	Xception+JFT	89.0
RecoNet	ResNet-101	**88.5**
RecoNet	ResNet-152	**89.0**

Table 3: Results on PASCAL-Context test set with background (60 classes)

Method	Backbone	mIoU
FCN-8s [31]		37.8
ParseNet [29]		40.4
Piecewise [26]		43.3
VeryDeep [40]		44.5
DeepLab-v2 [3]	ResNet-101	45.7
RefineNet [25]	ResNet-152	47.3
PSPNet [50]	ResNet-101	47.8
MSCI [24]	ResNet-152	50.3
Ding *et al.* [12]	ResNet-101	51.6
EncNet [47]	ResNet-101	51.7
DANet [14]	ResNet-101	52.6
SVCNet [11]	ResNet-101	53.2
CFNet [48]	ResNet-101	54.0
DMNet [15]	ResNet-101	54.4
RecoNet	ResNet-101	**54.8**

Table 4: Results on COCO-Stuff test set (171 classes)

Method	Backbone	mIoU
FCN-8s [31]		22.7
DeepLab-v2 [3]	ResNet-101	26.9
RefineNet [25]	ResNet-101	33.6
Ding *et al.* [12]	ResNet-101	35.7
SVCNet [11]	ResNet-101	39.6
DANet [14]	ResNet-101	39.7
EMANet [22]	ResNet-101	39.9
RecoNet	ResNet-101	**41.5**

Table 5: Results on SIFT-Flow test set

Method	pixel acc.	mIoU
Sharma et al. [35]	79.6	-
Yang et al. [42]	79.8	-
FCN-8s [31]	85.9	41.2
DAG-RNN+CRF [36]	87.8	44.8
Piecewise [26]	88.1	44.9
SVCNet [11]	89.1	46.3
RecoNet	**89.6**	**46.8**

Table 6: Results on ADE20K *val* set

Method	Backbone	mIoU
RefineNet [25]	ResNet-152	40.70
PSPNet [50]	ResNet-101	43.29
DSSPN [23]	ResNet-101	43.68
SAC [34]	ResNet-101	44.30
EncNet [47]	ResNet-101	44.65
CFNet [48]	ResNet-50	42.87
CFNet [48]	ResNet-101	44.89
CCNet [19]	ResNet-101	45.22
RecoNet	ResNet-50	**43.40**
RecoNet	ResNet-101	**45.54**

experiments are conducted on PASCAL-VOC12 *validation* set and more ablation studies can be found in supplementary material.

Different Components. In this part, we design several variants of our model to validate the contributions of different components. The experimental settings are the same with previous part. Here we have three main components, including global pooling module (GPM) and tensor low-rank reconstruction module inducing TGM and TRM. For fairness, we fix the tensor rank $r = 64$. The influence of each module is shown in Table 7. According to our experiment results, tensor low-rank reconstruction module contributes 9.9% mIoU gain in network performance and the pooling module also improves mIoU by 0.6%. Then we use the auxiliary loss after Res-4 block. We finally get 81.4% mIoU by using

Table 7: Ablation study on different components. The experiments are implemented using PASCAL-VOC12 validation dataset. FT represents fine-tune on PASCAL-VOC12 original training set

Method	TGM+TRM	GPM	Aux-loss	MS/Flip	FT	mIoU %
ResNet-50						68.7
ResNet-50	√					78.6
ResNet-50	√	√				79.2
ResNet-50	√	√	√			79.8
ResNet-101	√	√	√			81.4
ResNet-101	√	√	√	√		82.1
ResNet-101	√	√	√	√	√	82.9

Table 8: Ablation study on ten-sor rank. The results are obtained by using ResNet101 backbone and multi-scale evaluation

Method	Tensor Rank	mIoU %
RecoNet	16	81.2
RecoNet	32	81.8
RecoNet	48	81.4
RecoNet	**64**	**82.1**
RecoNet	80	81.6
RecoNet	96	81.0
RecoNet	128	80.7

Table 9: Results on PASCAL-VOC12 *val* set. RecoNet achieves the best performance with relatively small cost

Method	SS	MS/Flip	FLOPs
ResNet-101	-	-	190.6G
DeepLabV3+ [5]	79.45	80.59	+84.1G
PSPNet [50]	79.20	80.36	+77.5G
DANet [14]	79.64	80.78	+117.3G
PSANet [51]	78.71	79.92	+56.3G
CCNet [19]	79.51	80.77	+65.3G
EMANet [22]	80.09	81.38	+43.1G
RecoNet	**81.40**	**82.13**	**+41.9G**

GPM and TGM+TRM together. The result shows that the tensor low-rank reconstruction module dominants the entire performance.

Tensor Rank. Tensor rank r determines the information capacity of our reconstructed attention map. In this experiment, we use ResNet101 as the backbone. We sample r from 16 to 128 to investigate the effect of tensor rank. An intuitive thought is that the performance would be better with the increase of r. However, our experiment results on Table 8 illustrates that the larger r does not always lead to a better performance. Because we apply TGM+TRM on the input feature $X \in \mathbb{R}^{512\times64\times64}$, which has maximum tensor rank 64. An enormous r may increase redundancy and lead to over-fitting, which harms the network performance. Therefore, we choose $r = 64$ in our experiments.

Comparison with Previous Approaches. In this paper, we use deep-base ResNet as our backbone. Specifically, we replace the first 7×7 convolution in ResNet with three consequent 3×3 convolutions. This design is widely adopted in semantic segmentation and serves as the backbone network of many prior works [19, 22, 47, 48, 50]. Since the implementation details and backbones vary in different algorithms. In order to compare our method with previous approaches in absolutely fair manner, we implemented several state-of-the-art algorithms (listed in Table 9) based on our ResNet101 backbone and training setting. The

Table 10: Computational cost and GPU occupation of TGM+TRM. FLOPs (FLoating point Operations). We use tensor rank $r = 64$ for evaluation

Method	Channel	FLOPs	GPU Memory
Non-Local [38]	512	19.33G	88.00MB
APCNet [16]	512	8.98G	193.10MB
RCCA [19]	512	5.37G	41.33MB
A^2Net [6]	512	4.30G	25.00MB
AFNB [54]	512	2.62G	25.93MB
LatentGNN [49]	512	2.58G	44.69MB
EMAUnit [22]	512	2.42G	24.12MB
TGM+TRM	512	**0.0215G**	**8.31MB**

results are shown in Table 9. We compare our method with feature pyramid approaches such as PSPNet [50] and DeepLabV3+ [5]. The evaluation results show that our algorithm not only surpass these method in mIoU but also in FLOPs. Also, we compare our method with non-local attention based algorithms such as DANet [14] and PSANet [51]. It is noticed that our single-scale result outperforms their multi-scale results, which implies the superiority of our method. Additionally, we compare RecoNet with other low-cost non-local methods such as CCNet [19] and EMANet [22], where RecoNet achieves the best performance with relatively small cost.

4.4 Further Discussion

We further design several experiments to show computational complexity of the proposed method, and visualize some sub-attention maps from the reconstructed context features.

Computational Complexity Analysis. Our proposed method is based on the low-rank tensors, thus having large advantage on computational consumption. Recalling that non-local block has computational complexity of $\mathcal{O}(CH^2W^2)$. On the TGM stage, we generates a series of learnable vectors using 1×1 convolutions. The computational complexity is $\mathcal{O}(C^2 + H^2 + W^2)$ while on the TRM stage, we reconstruct the high-rank tensor from these vectors and the complexity is $\mathcal{O}(CHW)$ for each rank-1 tensor. Since $CHW >> C^2 > H^2 = W^2$, the total complexity is $\mathcal{O}(rCHW)$, which is much smaller than non-local block. Here r is the tensor rank. Table 10 shows the FLOPs and GPU occupation of TGM+TRM. From the table we can see that the cost of TGM+TRM is neglegible compared with other non-local based methods. Our proposed method has about **900** times less FLOPs and more than **100** times less FLOPs compared with non-local block and other non-local-based methods, such as A^2Net [6] and LatentGNN [49]. Besides of these methods, we calculate the FLOPs and GPU occupation of RCCA, AFNB and EMAUnit, which is core component of CCNet [19], AsymmetricNL [54] and EMANet [22]. It can be found that TGM+TRM has the lowest computational overhead.

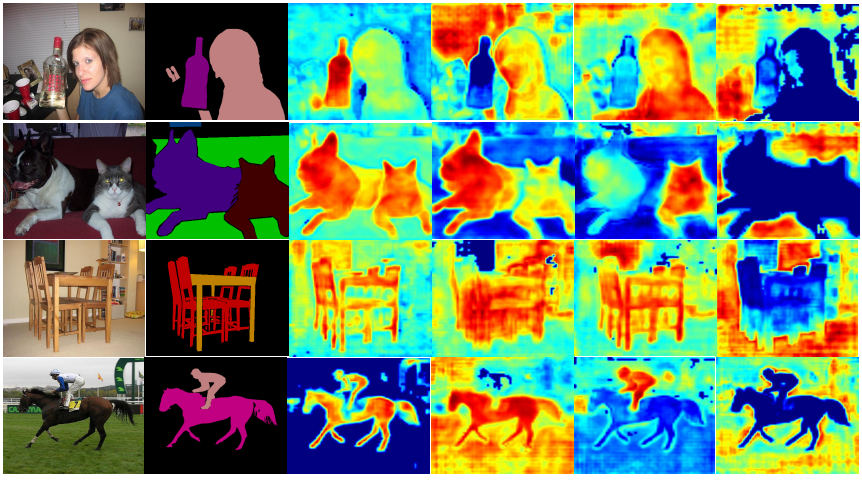


Fig. 5: Visualization of sub-attention map. From left to right are Image, Ground Truth, $\boldsymbol{A}_i \cdot \boldsymbol{X}$, $\boldsymbol{A}_j \cdot \boldsymbol{X}$, $\boldsymbol{A}_k \cdot \boldsymbol{X}$, and $\boldsymbol{A}_l \cdot \boldsymbol{X}$. It can be found that sub-attention maps mainly focus on the different parts of image.

Visualization. In our proposed method, context features are constructed by the linear combination of sub-attention maps, *i.e*, $\boldsymbol{A}_i \cdot \boldsymbol{X}$. Therefore, we visualize their heat maps to check the part of features they activate. We randomly select four sub-attention maps $\boldsymbol{A}_i \cdot \boldsymbol{X}$, $\boldsymbol{A}_j \cdot \boldsymbol{X}$, $\boldsymbol{A}_k \cdot \boldsymbol{X}$, $\boldsymbol{A}_l \cdot \boldsymbol{X}$, as shown in Fig. 5. We can see that different sub-attention maps activate different parts of the image. For instance, for the last case, the four attention maps focus on the foreground, the horse, the person, and the background, respectively, which implies that the low-rank attention captures the context fragments and RecoNet can catch long-range dependencies.

5 Conclusion

In this paper, we propose a tensor low-rank reconstruction for context features prediction, which overcomes the feature compression problem that occurred in previous works. We collect high-rank context information by using low-rank context fragments that generated by our proposed tensor generation module. Then we use CP reconstruction to build up high-rank context features. We embed the fine-grained context features into our proposed RecoNet. The state-of-the-arts performance on different datasets and the superiority on computational consumption show the success of our context collection method.

6 Appendix

6.1 More Experimental Results

We conduct experiments on Cityscapes dataset [9], which is a famous scene segementation dataset that includes 19 semantic classes. It provides 2975/500/1525 images for training, validation and testing. Since the training setting of Cityscapes is very distinct to the implementation details that presented in main paper, we put the results in supplementary materials.

The input images are cropped into 512 × 1024 before input. The batch size we use is 8. Initially, the learning rate $lr = 0.01$. SGD optimizer with momentum = 0.9 and weight decay = 0.0005 is applied for training. The evaluation metrics and data augmentation strategies we use are the same as main paper. For the evaluation on *val*/*test* set, we train 40K/100K iterations on *train*/*train* + *val* set respectively. The testing results are shown on Table 11, which collects current state-of-the-art attention based methods. RecoNet get better performance than these approaches. The online hard example mining (OHEM) strategy is not used in our implementation since it is time consuming. The result is avaliable on the website.[6]

Table 11: Results on Cityscapes *test* set

Method	Backbone	mIoU
PSANet [51]	ResNet-101	80.1
CFNet [48]	ResNet-101	79.6
AsymmetricNL [54]	ResNet-101	81.3
CCNet [19]	ResNet-101	81.4
DANet [14]	ResNet-101	81.5
ACFNet [46]	ResNet-101	81.8
RecoNet	ResNet-101	**82.3**

In order to validate the consistency of RecoNet, we conduct additional ablation experiments on Cityscapes dataset. The tensor rank is set to $r = 64$ for ablation. In Table 12, it can be found that TGM+TRM contributes 5.8 % mIoU improvement (73.1% to 78.9%), which dominates the other modules. The experimental results show that RecoNet is consistent on different datasets.

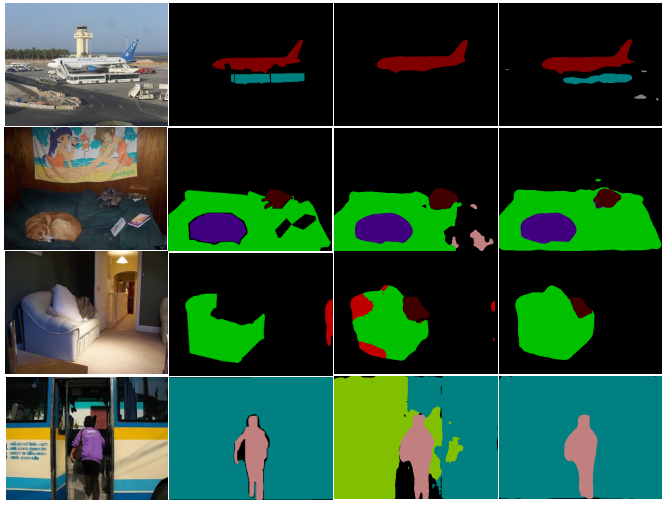
6.2 More Visualization

Fig. 6 shows some results of RecoNet-101 on PACAL-VOC12 *validation* dataset. The figure shows that RecoNet has a better qualitative result, especially in the boundary, which also demonstrates its effectiveness of context modeling.

[6] https://www.cityscapes-dataset.com/anonymous-results/?id=7c7bfabc1026a9fd07b348bfd311c56a57ba0369969f3bd9fd9f036ce49a2934

Table 12: Ablation study on different components. The experiments are implemented using Cityscapes validation set

Method	TGM+TRM	GPM	Aux-loss	MS/Flip	mIoU %
ResNet-50				√	73.1
ResNet-50	√			√	78.9
ResNet-50	√	√		√	79.4
ResNet-50	√	√	√	√	79.8
ResNet-101	√	√	√		80.5
ResNet-101	√	√	√	√	81.6



(a) Image (b) GT (c) FCN (d) RecoNet

Fig. 6: Qualitative results on PASCAL-VOC12 *validation* dataset.

References

1. Badrinarayanan, V., Kendall, A., Cipolla, R.: SegNet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE TPAMI **39**(12), 2481–2495 (2017) 2
2. Caesar, H., Uijlings, J., Ferrari, V.: COCO-Stuff: Thing and stuff classes in context. In: Proc. CVPR. pp. 1209–1218 (2018) 10
3. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE TPAMI **40**(4), 834–848 (2018) 11
4. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017) 2, 4, 11
5. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proc. ECCV. pp. 801–818 (2018) 2, 4, 11, 12, 13

6. Chen, Y., Kalantidis, Y., Li, J., Yan, S., Feng, J.: Aˆ 2-Nets: Double attention networks. In: Proc. NIPS. pp. 352–361 (2018) 4, 13
7. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Proc. CVPR. pp. 1251–1258 (2017) 9
8. Chorowski, J.K., Bahdanau, D., Serdyuk, D., Cho, K., Bengio, Y.: Attention-based models for speech recognition. In: Proc. NIPS. pp. 577–585 (2015) 3
9. 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) 15
10. Cui, Y., Chen, Z., Wei, S., Wang, S., Liu, T., Hu, G.: Attention-over-attention neural networks for reading comprehension. In: Proc. ACL (2017) 3
11. Ding, H., Jiang, X., Shuai, B., Liu, A.Q., Wang, G.: Semantic correlation promoted shape-variant context for segmentation. In: Proc. CVPR. pp. 8885–8894 (2019) 11
12. Ding, H., Jiang, X., Shuai, B., Qun Liu, A., Wang, G.: Context contrasted feature and gated multi-scale aggregation for scene segmentation. In: Proc. CVPR. pp. 2393–2402 (2018) 11
13. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) challenge. International Journal of Computer Vision **88**(2), 303–338 (2010) 3, 9
14. Fu, J., Liu, J., Tian, H., Fang, Z., Lu, H.: Dual attention network for scene segmentation. arXiv preprint arXiv:1809.02983 (2018) 9, 11, 12, 13, 15
15. He, J., Deng, Z., Qiao, Y.: Dynamic multi-scale filters for semantic segmentation. In: Proc. ICCV. pp. 3562–3572 (2019) 9, 10, 11
16. He, J., Deng, Z., Zhou, L., Wang, Y., Qiao, Y.: Adaptive pyramid context network for semantic segmentation. In: Proc. CVPR. pp. 7519–7528 (2019) 2, 4, 9, 10, 13
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. CVPR. pp. 770–778 (2016) 7
18. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proc. CVPR. pp. 7132–7141 (2018) 3, 8
19. Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., Liu, W.: CCNet: Criss-cross attention for semantic segmentation. In: Proc. ICCV. pp. 603–612 (2019) 4, 10, 11, 12, 13, 15
20. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM Review (SIREV) **51**(3) (2009) 2, 3, 4
21. Lebedev, V., Ganin, Y., Rakhuba, M., Oseledets, I., Lempitsky, V.: Speeding-up convolutional neural networks using fine-tuned CP-decomposition. In: Proc. ICLR (2015) 3
22. Li, X., Zhong, Z., Wu, J., Yang, Y., Lin, Z., Liu, H.: Expectation-maximization attention networks for semantic segmentation. In: Proc. ICCV. pp. 9167–9176 (2019) 4, 11, 12, 13, 14
23. Liang, X., Xing, E., Zhou, H.: Dynamic-structured semantic propagation network. In: Proc. CVPR. pp. 752–761 (2018) 11
24. Lin, D., Ji, Y., Lischinski, D., Cohen-Or, D., Huang, H.: Multi-scale context intertwining for semantic segmentation. In: Proc. ECCV. pp. 603–619 (2018) 11
25. Lin, G., Milan, A., Shen, C., Reid, I.: RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. Proc. CVPR pp. 1925–1934 (2017) 2, 11
26. Lin, G., Shen, C., Van Den Hengel, A., Reid, I.: Efficient piecewise training of deep structured models for semantic segmentation. In: Proc. CVPR. pp. 3194–3203 (2016) 11

27. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: Proc. ECCV. pp. 740–755 (2014) 9
28. Liu, C., Yuen, J., Torralba, A.: SIFT Flow: Dense correspondence across scenes and its applications. IEEE TPAMI **33**(5), 978–994 (2011) 10
29. Liu, W., Rabinovich, A., Berg, A.C.: ParseNet: Looking wider to see better. arXiv preprint arXiv:1506.04579 (2015) 5, 11
30. Liu, Z., Li, X., Luo, P., Loy, C.C., Tang, X.: Semantic image segmentation via deep parsing network. In: Proc. ICCV. pp. 1377–1385 (2015) 11
31. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proc. CVPR. pp. 3431–3440 (2015) 2, 4, 10, 11
32. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in PyTorch. In: NIPS Workshop (2017) 9
33. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: Proc. MICCAI. pp. 234–241 (2015) 2
34. Rui, Z., Sheng, T., Zhang, Y., Li, J., Yan, S.: Scale-adaptive convolutions for scene parsing. In: Proc. ICCV. pp. 2031–2039 (2017) 11
35. Sharma, A., Tuzel, O., Liu, M.Y.: Recursive context propagation network for semantic scene labeling. In: Proc. NIPS (2014) 11
36. Shuai, B., Zup, Z., Wang, B., Wang, G.: Scene segmentation with DAG-recurrent neural networks. IEEE TPAMI **40**(6), 1480–1493 (2018) 11
37. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Proc. NIPS. pp. 5998–6008 (2017) 3
38. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proc. CVPR. pp. 7794–7803 (2018) 4, 13
39. Woo, S., Park, J., Lee, J.Y., So Kweon, I.: CBAM: Convolutional block attention module. In: Proc. ECCV. pp. 3–19 (2018) 4, 8
40. Wu, Z., Shen, C., Hengel, A.v.d.: Bridging category-level and instance-level semantic image segmentation. arXiv preprint arXiv:1605.06885 (2016) 11
41. Wu, Z., Shen, C., Van Den Hengel, A.: Wider or deeper: Revisiting the resnet model for visual recognition. Pattern Recognition **90**, 119–133 (2019) 11
42. Yang, J., Price, B., Cohen, S., Yang, M.H.: Context driven scene parsing with attention to rare classes. In: Proc. CVPR. pp. 3294–3301 (2014) 9, 11
43. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: Hierarchical attention networks for document classification. In: Proc. NAACL. pp. 1480–1489 (2016) 3
44. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Learning a discriminative feature network for semantic segmentation. In: Proc. CVPR. pp. 1857–1866 (2018) 8, 11
45. Yu, X., Liu, T., Wang, X., Tao, D.: On compressing deep models by low rank and sparse decomposition. In: Proc. CVPR. pp. 7370–7379 (2017) 3
46. Zhang, F., Chen, Y., Li, Z., Hong, Z., Liu, J., Ma, F., Han, J., Ding, E.: Acfnet: Attentional class feature network for semantic segmentation. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019) 15
47. Zhang, H., Dana, K., Shi, J., Zhang, Z., Wang, X., Tyagi, A., Agrawal, A.: Context encoding for semantic segmentation. In: Proc. CVPR. pp. 7151–7160 (2018) 4, 6, 7, 8, 9, 10, 11, 12
48. Zhang, H., Zhang, H., Wang, C., Xie, J.: Co-occurrent features in semantic segmentation. In: Proc. CVPR. pp. 548–557 (2019) 2, 4, 7, 9, 10, 11, 12, 15

49. Zhang, S., He, X., Yan, S.: LatentGNN: Learning efficient non-local relations for visual recognition. In: Proc. ICML. pp. 7374–7383 (2019) 13
50. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proc. CVPR. pp. 2881–2890 (2017) 2, 4, 5, 7, 9, 10, 11, 12, 13
51. Zhao, H., Zhang, Y., Liu, S., Shi, J., Change Loy, C., Lin, D., Jia, J.: PSANet: Point-wise spatial attention network for scene parsing. In: Proc. ECCV. pp. 267–283 (2018) 2, 4, 6, 12, 13, 15
52. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., Torr, P.H.: Conditional random fields as recurrent neural networks. In: Proc. ICCV. pp. 1529–1537 (2015) 11
53. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ADE20K dataset. In: Proc. CVPR. pp. 633–641 (2017) 10
54. Zhu, Z., Xu, M., Bai, S., Huang, T., Bai, X.: Asymmetric non-local neural networks for semantic segmentation. In: Proc. ICCV. pp. 593–602 (2019) 13, 15