Machine Vision and Applications (2023) 34:22
https://doi.org/10.1007/s00138-023-01373-7

ORIGINAL PAPER

Dual Context Network for real-time semantic segmentation

Hong Yin[1] · Wenbin Xie[1] · Jingjing Zhang[2] · Yuanfa Zhang[2] · Weixing Zhu[1] · Jie Gao[1] · Yan Shao[1] · Yajun Li[3]

Received: 20 January 2021 / Revised: 30 October 2022 / Accepted: 2 January 2023 / Published online: 19 January 2023

Abstract

Real-time semantic segmentation is a challenging task as both segmentation accuracy and inference speed need to be considered at the same time. In this paper, a Dual Context Network (DCNet) is presented to address this challenge. It contains two independent sub-networks: Region Context Network and Pixel Context Network. Region Context Network is main network with low-resolution input and features re-weighting module to achieve sufficient receptive field. Meanwhile, Pixel Context Network with location attention module is to capture the location dependencies of each pixel for assisting the main network to recover spatial detail. A contextual feature fusion is introduced to combine output features of these two sub-networks. The experiments show that DCNet can achieve high-quality segmentation while keeping a high speed. Specifically, for Cityscapes test dataset, it can achieve 76.1% Mean IOU with the speed of 82 FPS on a single GTX 2080Ti GPU when using ResNet50 as backbone and 71.2% Mean IOU with the speed of 142 FPS when using ResNet18 as backbone.

Keywords Real-time · Semantic segmentation · Feature fusion · Context information · Location attention

1 Introduction

Semantic segmentation is a fundamental and challenging task in computer vision. It aims to predict dense labels for all pixels. The research of this task can be applied to a variety of potential applications, such as autonomous driving, augmented reality and robot sensing, maintaining efficient inference speed and high segmentation accuracy to meet the high demand of these real applications.

Contrary to the development of full-precision semantic segmentation [33, 34, 37, 40, 44], it is necessary to accelerate the inference speed without sacrificing too much quality for real-time semantic segmentation. Recently, more and more real-time semantic segmentation approaches [4, 5, 22, 27] have been proposed to address the above issues. Some works [1, 2] prune the channels of the model to reduce calculation time, which will weaken the capabilities of the feature discriminative. In another way, some other works [5, 10] try to reduce the computation by restricting the input size, these methods are simple and effective, but they ignore how to recover the spatial information. Also, some real-time semantic segmentation approaches [4, 6] address this task by introducing a shallow lightweight network [23, 24, 35]. However, these lightweight networks lead to the degradation of performance as the weak representation ability.

From a macroperspective, the semantic segmentation task can be divided into two parts: region semantic prediction and pixel-level detail recovery. The whole network is real-time only if it can keep high-speed inference in both parts. Usually, a simple semantic segmentation model starts with a backbone network pretrained from image classification task which gradually downsamples the resolution and increases the number of channel of feature maps. The different stages have different feature discriminative abilities. There is richer spatial information in the lower stage, and more accurate semantic predictions can be achieved in the higher stage. To achieve a competitive combination of efficiency and prediction accuracy, some researchers employ the U-shape structure [1, 4, 11]. The U-shape structure recovers the loss of spatial information by fusing the feature maps of each stage of the backbone network. However, this structure will introduce huge extra computation. As another typical structure, feature reuse [12, 36] can enhance the learning capacity of the network and increase the receptive field with fewer parameters, but it is difficult to refine spatial detail (Fig. 1).

✉ Wenbin Xie
1171801987@qq.com

[1] Army Engineering University of PLA, Nanjing, China

[2] Naval Research Academy, Nanjing, China

[3] Nanjing University of Science and Technology, Nanjing 210094, China

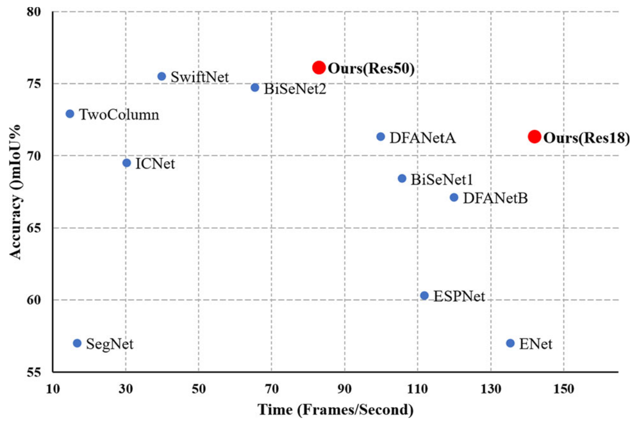


Fig. 1 Inference speed and mIoU performance on Cityscapes [8] test set. Existing real-time methods involved are SegNet [1], ENet [2], ICNet [5], TwoColumn [10], BiSeNet [4], ESPNet [3], DFANet [6], SwiftNet [20] and our DCNet (based on two backbone networks)

Based on comparing the previous structures with each other, it can be found that region semantic prediction depends on the region contextual dependencies. Meanwhile, the key point of the detail recovery is how to receive the category context of each pixel. In view of the above analysis, a Dual Context Network (DCNet) is presented with two parts: Region Context Network and Pixel Context Network. All these structures are visualized in Fig. 2.

First, the Region Context Network involves a backbone network which has strong ability to learn the feature representation. Considering that there are many layers in the backbone network, using 1/2-resolution image as input to reduce the inference time. Additionally, a feature re-weighting module is proposed to extract global context to weight the feature information in the low stage. Second, for Pixel Context Network, it adopts three convolutional layers to downsample the full-resolution image quickly. Then, a location attention module is connected to the 1/8 feature map. The Pixel Context Network aims to improve the small objects prediction and recover details like the object boundary by capturing location correlations in the large size feature maps. Finally, contextual feature fusion is designed to effectively fuse the features of two sub-networks. As shown in the experimental section, the proposed DCNet achieves impressive performance on Cityscapes [8], CamVid [9] and COCO-Stuff [29] benchmark datasets. Considering both the accuracy and inference speed, the proposed methods yield very competitive perform comparing with the real-time baseline methods, as shown in Fig. 1.

The proposed method is interesting in the following aspects:

A novel Dual Context Network (DCNet) is proposed with two sub-networks: Region Context Network and Pixel Context Network. These two sub-networks take different resolution images as input to accomplish semantic prediction and detail information recovery, respectively.

Motivated by the re-weighting module in SENet [14], this work developed a feature re-weighting module to integrate context information and a location attention module to model spatial interdependencies. In addition, the contextual feature fusion is applied to further improve the accuracy.

Experiments prove superior performance of our method through comparison with a number of state-of-the-art networks on the benchmarks of Cityscapes, CamVid and COCO-Stuff.

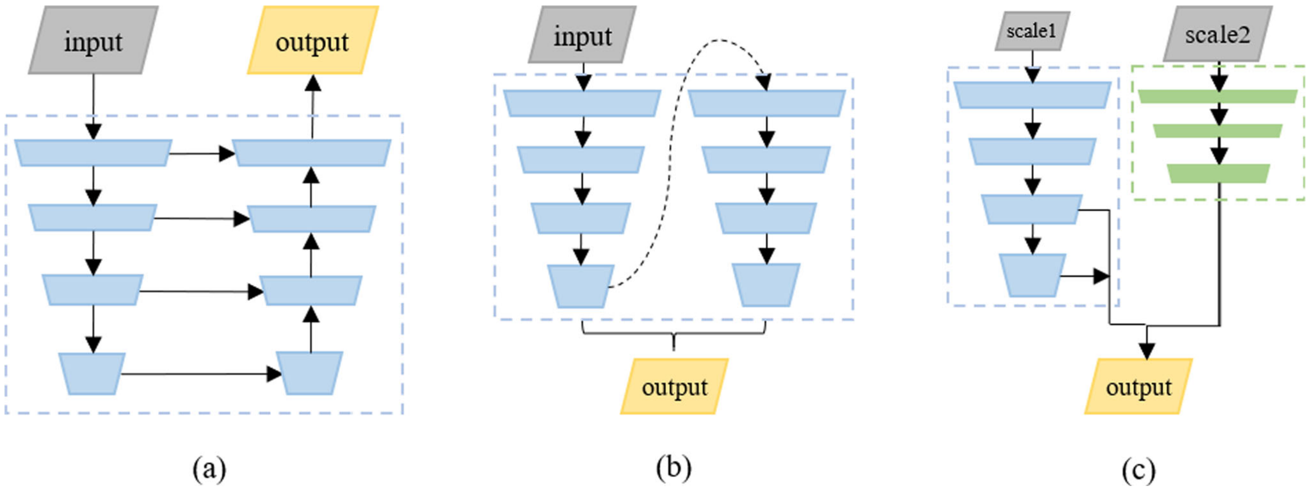


Fig. 2 Network structure comparison. **a** U-shape. **b** Feature reuse. **c** Our DCNet structure. Dashed boxes of different colors represent different networks; scale1 and scale2 are different resolutions of the same image

2 Related work

2.1 Real-time segmentation

In recent years, many methods have explored high-quality prediction under limited calculation in semantic segmentation. SegNet [1] designs a small encoder network and a corresponding decoder network to reduce the number of parameters. ENet [2] proposes a lightweight network by dropping the last stages to pursue high speed. ESPNet [3] is based on the efficient spatial pyramid, which is efficient in terms of computation. ICNet [5] employs an image cascade network to combine semantic information in low resolution and details in high resolution. DFANet [6] presents a network structure with multiple encoding streams to obtain a balance between the speed and accuracy. BiSeNet [4] introduces a spatial path with a shallow network to preserve the spatial information. Considering these methods synthetically, a high-performance backbone network with low-resolution image is used to obtain semantic information. Moreover, different from other methods, the detail information is improved by utilizing the location correlations of each pixel. Recent work [49] shows that carefully designed decoder can also improve the efficiency of the lightweight models, and important strategies include depth compression and kernel size selection. The strategies of the efficient segmentation models are summarized in Table.1.

2.2 Context information

The context information [43] plays an important role in deep learning, especially semantic segmentation task. Most methods generate a high-quality prediction by fusing different context information. The context information can be categorized into two kinds: (1) region context. ParseNet [12] applies global average pooling to obtain feature maps which represent the whole image. PSPNet [16] develops the Spatial Pyramid Pooling which aggregates context information of different size regions. Deeplab v3 [15] employs atrous convolution in parallel with multiple atrous rates to capture multi-scale context, called Atrous Spatial Pyramid Pooling. (2) Pixel context. Different from region context, pixel context represents the set of pixels which belong to the same category for each pixel. It takes more time to extract the pixel context than the region context, but the detail recovery can be greatly improved. OCNet [18] introduces an object context pooling to update the representation for each pixel.

Table 1 The strategies of the efficient segmentation models

	Light weight backbone	Efficient decoder	Multi-path fusion
SegNet[1]	√	√	
ENet [2]	√	√	
ESPNet [3]	√		
ICNet [5]			√
DFANet [6]			√
BiSeNet [4]			√
TwoCol.[10]			√
SwiftNet[20]	√		√
SFNet [48]	√		
TD Flows[49]		√	

In particular, understanding and utilizing context information under limited time is very critical for real-time semantic segmentation. In this paper, most semantic parts are harvested by capturing region context in a deep network. Considering inference speed, a shallow network is used to aggregate the pixel context for detail information recovery.

2.3 Attention mechanism

Attention mechanism has been widely used in various fields of deep learning in recent years, such as voice recognition, natural language processing and computer vision. For semantic segmentation, attention mechanism can determine which part of feature map needs more attention. SE-Net [14] learns a channel-wise attention by modeling the dependencies of convolutional features between channels. Furthermore, the work [13, 18, 38] explores effectiveness of non-local operation in space–time dimension based on the self-attention method. Inspired by attention mechanism, a feature re-weighting module is designed in the context network to recalibrate the low-level features with small receptive field. Besides, a location attention module is employed on the basis of self-attention to calculate the correlation between all locations.

2.4 Feature fusion

Feature fusion is widely employed in compute vision [41, 42] for different purposes. ICNet [5] proposes a cascade feature fusion to combine cascade features from different resolution inputs. BiSeNet [4] uses a specific feature fusion module to fuse features from different paths. Additionally, a lot of methods use dense connections to fuse low-level and high-level features together. Like the DFANet [6], they deploy two strategies to implement cross-level feature aggregation: sub-network aggregation and sub-stage aggregation. Recent works have shown that combination of contour information can effectively improve segmentation performance. EGNet [52] fuses complementarity salient edge and object information to locate salient objects with high-accuracy boundaries. SCG [53] combines the Mask R-CNN features with the saliency and contour features to supply pixel-wise saliency information and generic object contour prior to detect and segment generic objects. Motivated by these works, the proposed method designs a Pixel Context Network to addressed the edge details. In our model, the output features of the two sub-networks are different in level of context representation, and a context feature fusion unit is adopted to pursuit better accuracy.

3 Dual Context Network

The whole Dual Context Network is presented in this section from three aspects: (1) Region Context Network with feature re-weighting module; (2) Pixel Context Network with location attention module; (3) network architecture with contextual feature fusion.

3.1 Region Context Network (RCN)

In the task of semantic segmentation, most of modern methods focus on region context information, like the pyramid pooling module [16], atrous spatial pyramid pooling [15, 25, 39, 45], or dense U-shape [6, 19]. However, while more complex networks can achieve good accuracy, they are not applicable to realize real-time processing due to their larger computing requirements. Therefore, the Region Context Network is primarily designed with limited computing time.

In this sub-network, the ResNet [7] is used as the base recognition model. Compared with the lightweight model, ResNet has stronger high-level semantic extraction capabilities but more calculation time. In order to speed up the network, low-resolution input is used as an intuitive speedup strategy. As shown in the following experiments, 1/2 sized image is chosen as input because much time is saved, but less accuracy is sacrificed. According to our observation, the different stages of ResNet have different recognition abilities. The lower stage has more spatial information but poor semantic consistency, and the high stage has large receptive field, but the prediction is coarse. Based on this observation, atrous convolution is applied in the last two stages to capture multi-scale context information. Due to low-resolution input, this operation does not add much calculation time. Then, a global average pooling is connected at the tail of the ResNet, which introduces the consistency constraint as a guidance. Moreover, to refine the output feature of the low stage, a feature re-weighting module (FRM) is designed, as shown in Fig. 3b. Finally, the high-level features are bilinear upsampled by a factor of 2 and concatenated with the corresponding low-level features processed by FRM.

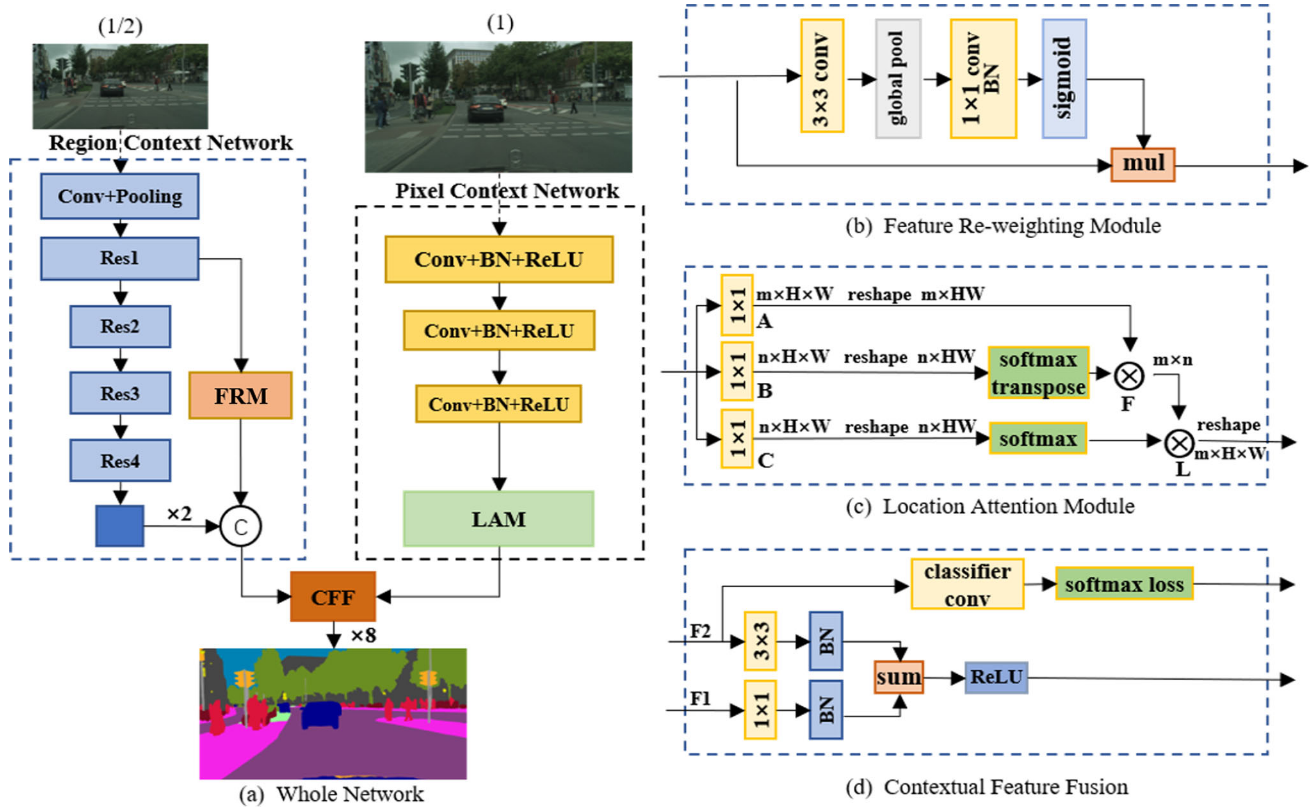
Feature re-weighting module (FRM): In the Region Context Network, it combines the features of low and high stages as the output of this sub-network. However, due to the small receptive field and the lack of spatial context guidance in the low stage, simply adding the channels will lead to poor semantic prediction. For this reason, a feature re-weighting module is designed to refine the low-level features. The first component of the module is a 3 × 3 convolution layer. This layer is used to unify the number of channel s to 256. Then the following is a global average pooling, which can capture global context information. 1 × 1 convolution and batch normalization are applied in front of the sigmoid activation. This operation can improve the nonlinear representation ability of the feature without changing the size of the receptive field. Finally, this module computes a weight vector, which reweights the low-level feature maps. In this way, it can reduce differences between different stages.

The propose FRM module is inspired by the SENet [14], but there are notable differences between these two methods:

1) The target of SENet is to enhance the channel-wise feature responses by modeling the interdependencies between channels, but the motivation of FRM is to calibrate the feature responses from different layers of network which have different receptive fields and spatial contexts.
2) A 3 × 3 convolution layer is added to the FRM branch before the pooling module, the target of this design is to extend the receptive field of low layer feature, so that the aggregated information can be merged with features from deeper layers more effectively.

3.2 Pixel Context Network (PCN)

Through the Region Context Network, most semantic predictions are harvested. However, this prediction is coarse because the boundary is blurry and some details are missing like the small objects. Thus, recovering and refining the coarse prediction is very important for improving accuracy. To address this problem, some full-precision models [17, 20] adopt dense design. However, the dense design is not suitable for real-time semantic segmentation because it is time-consuming. In this work, small objects and boundaries are considered as pixel-level structure. Therefore, the detail information can be recovered with the help of the pixel context. With this motivation, a Pixel Context Network is used to guide the feature learning. To avoid affecting the whole inference speed, this sub-network only contains three layers and an attention module, so it takes the full-resolution image as input. Each layer includes a 3 × 3 convolution with stride = 2, followed by batch normalization and ReLU. Thus, it can extract the 1/8-resolution feature map quickly, which encodes rich spatial information. After these layers, a location attention module is introduced, as shown in Fig. 3c. This module effectively captures the global information and distributes it to each location. With the support of the Pixel



(a) Whole Network

(b) Feature Re-weighting Module

(c) Location Attention Module

(d) Contextual Feature Fusion

Fig. 3 Overview of our Dual Context Network. **a** Network architecture. **b** Components of the feature re-weighting module. **c** Components of the location attention module. **d** Components of contextual feature fusion. "C" means concatenation; " × *N*" represents *N* times upsampling operation. "" denotes matrix multiplication

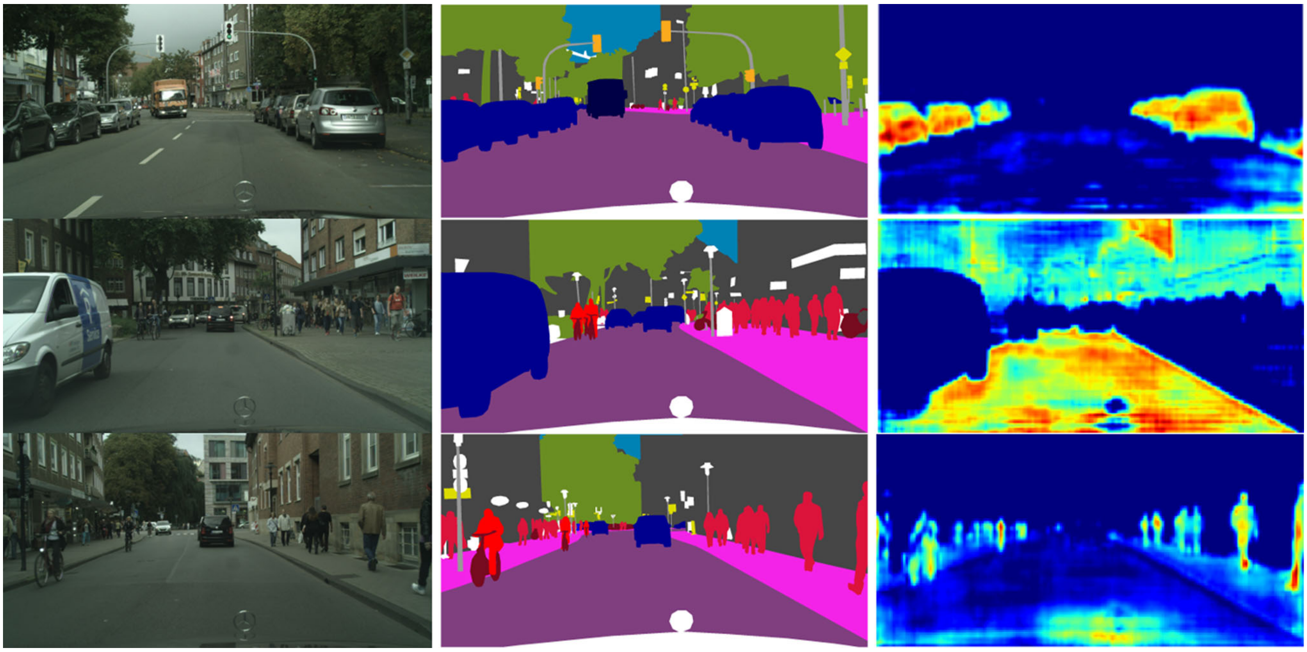


Fig. 4 Illustrations of maps with the support of Pixel Context Network. For each row, it shows an input image, the ground truth and the pixel context maps of the three categories

Context Network, pixels belonging to the same category can complement each other, as shown in Fig. 4.

Location attention module: Inspired by [13, 21, 26], the location attention module (LAM) is designed to capture long-range relations of each pixel to enhance the representative capability of each location, as illustrated in Fig. 3c. The 1/8-resolution feature map is fed into a 1 × 1 convolution layers with batch normalization and ReLU to obtain three new feature maps A, B and C, respectively. $A \in \mathbb{R}^{m\times H\times W}$ and $\{B, C\} \in \mathbb{R}^{n\times H\times W}$, where m and n represent

$$B''_{i,j} = \frac{exp(B'_{i,j})}{\sum_{j=1}^{N} exp(B'_{i,j})} \tag{1}$$

$$C''_{i,j} = \frac{exp(C'_{i,j})}{\sum_{j=1}^{N} exp(C'_{i,j})} \tag{2}$$

the channel number of the feature map, and H and W represent the height and width of the feature map, respectively. Firstly, all the three tensors are flattened along the last dimension and reshaped as $A' \in \mathbb{R}^{m\times N}$ and $\left\{B', C'\right\} \in \mathbb{R}^{n\times N}$, where $N = H \times W$ is the spatial size of the feature map. Then, Softmax is applied along each row of feature maps B' and C' to normalized the feature response of each individual channel:

where $B'_{i,j}$ and $C'_{i,j}$ are the elements of feature maps B' and C' at position (i,j), and $B''_{i,j}$ and $C''_{i,j}$ are the elements of normalized feature maps $\left\{B'', C''\right\} \in \mathbb{R}^{n\times N}$. The attention operation is defined as:
where $A'' \in \mathbb{R}^{m\times N}$ is the attention augmented feature map which is further reshape back to a 3-dimensional tensor $A^o \in \mathbb{R}^{m\times H\times W}$.

$$A'' = A'(B'')^T C'' \tag{3}$$

The explanation of Eq. (3) is straightforward. Based on the law of association $A'(B'')^T C'' = A'\left(\left(B''\right)^T C''\right)$, the second term on the right side $\left(B''\right)^T C'' \in \mathbb{R}^{N\times N}$ is a correlation matrix which captures the similarities of all the spatial pixel pairs. Therefore, $A'\left(\left(B''\right)^T C''\right)$ means the features in A' are enhanced by the pixels with similar feature response. Regardless of distance, pixels of the same category achieve mutual gains, which is important for recovering and refining detailed information. It should be noted that for efficiency reason, the learning phase still follows Eq. (3) to avoid the storage of the $N \times N$ correlation matrix.

3.3 Network architecture

With the Region Context Network and the Pixel Context Network, the Dual Context Network is presented for real-time semantic segmentation as illustrated in Fig. 3a.

In the Region Context Network, the pre-trained ResNet is used as the backbone network. 1/2-resolution image is fed into the backbone network to speed up without much precision sacrifice. The atrous convolution is utilized in the last two stages to acquire region context information for high-level feature maps. Then the feature re-weighting module changes the weights of low-level channels to enhance the consistency. Besides, a shallow network is designed with only three convolution layers and a location attention module, which captures pixel context information for each pixel. Therefore, it results in limited computation with full-resolution input. With the support of both sub-networks, the whole network can achieve real-time inference speed and high performance.

Contextual feature fusion: The features of the two sub-networks focus on different levels of context information. Thus, it cannot simply sum up these features. In the Region Context Network, the context is defined over rigid rectangle regions and includes pixels of various categories. However, the context of the Pixel Context Network gathers the pixels of the same category. Therefore, a contextual feature fusion is proposed to combine these features. As shown in Fig. 3d, a 1 × 1 projection convolution is applied for F_1 (the output feature of Region Context Network) and a 3 × 3 atrous convolution for F_2 (the output feature of Pixel Context Network) to make them have the same number of channels. Then, two batch normalization layers are utilized to normalize these two features, respectively. Finally, the fused feature can be obtained followed by an element-wise sum and a ReLU. Meanwhile, before fusing, an auxiliary loss is adopted for F_2 to enhance learning.

Discussion. The targets of RCN and PCN sub-networks are different. Therefore, they have different specially designed structures to achieve their respective targets. RCN is to generate the coarse segmentation using the context captured by the large perspective fields; thus, it has longer inference path but low-resolution input, which guarantee the efficiency of the subnetwork. Meanwhile, PCN is used to capture the complementary details (boundaries and small objects); therefore, it needs full-resolution input to preserve the details but shorter inference path to reduce the computation cost.

4 Experiments

The proposed method is effective for high-resolution images; the architecture is evaluated on Cityscapes, CamVid and

COCO-Stuff benchmarks. The datasets and the implementation details are introduced firstly. Then, Cityscapes is adopted to investigate the effects of each component of our proposed method. Next, the accuracy and speed results are reported on Cityscapes, CamVid and COCO-Stuff datasets compared with other full-precision models and real-time models.

4.1 Dataset and training details

Dataset: The Cityscapes is a large, diverse set of stereo video sequences recorded in streets from 50 different cities, it contains 5,000 finely annotated images and 19,998 images with coarse annotation, and all images have a high resolution of 2048 × 1024. This dataset contains 30 classes, and 19 of them are used in training and testing. Following the standard setting of Cityscapes, the finely annotated images are split into training, validation and testing sets with 2,975, 500 and 1,525 images, respectively.

The CamVid is another street scene dataset, and it contains 701 images extracted from high-resolution video sequences with resolution up to 960 × 720. For comparison with prior work, the dataset is partitioned into 367, 101 and 233 images for training, validation and testing, respectively. For evaluation, 11 semantic classed are used.

The COCO-Stuff contains 91 stuff classes and 1 class 'unlabeled.' It is labeled dataset based on MS-COCO [30] for stuff segmentation in context. Following the split in [29], 9 K images are used for training and 1 K for testing.

Implementation: The experiments are conducted based on platform PyTorch. All experiments are on a workstation with Nvidia 2080Ti cards. Our testing uses only one card. The mini-batch stochastic gradient descent (SGD) is used with batch size 16, momentum 0.9 and weight decay 0.0001 in training. The base learning rate is 0.01 and the 'poly' learning rate policy is adopted with power 0.9. All experiments are trained for totally 200 epochs with two GPU cards.

The dataset is augmented in training process, such as mean subtraction, random horizontal flip and random scale. The scales contain {0.5, 1.0, 1.25, 1.5, 2.0}. Finally, the image (Cityscapes) is cropped into 960 × 960 for training. All backbone networks are pretrained on ImageNet dataset.

4.2 Ablation study

In this subsection, each component of our proposed model is investigated step by step. The following experiments are applied to the Cityscapes validation dataset.

Backbone network: Backbone network is very important for model accuracy and speed, the large backbone networks take much time to extract complex features, and however, the small backbone networks lead to degradation of segmentation accuracy. To prove the effectiveness of each module, a large backbone network (ResNet50) and a small backbone network (ResNet18) are implemented, and both of them are modified networks with atrous convolution. The detailed architectures of these two networks are summarized in Table 2.

Table 2 The modified ResNet architecture

Layer name	ResNet18	ResNet50
conv1	7 × 7, 64, stride 2	
conv2_x	3 × 3 max pool, stride 2	
	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$ stride 1	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$ stride 1
conv3_x	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$ stride 2	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$ stride 2
conv4_x	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$ stride 1 dilation 2	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$ stride 1 dilation 2
conv5_x	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$ stride 1 dilation 4	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$ stride 1 dilation 4

Building blocks are shown in brackets with the number of blocks stacked. Dilation means convolution with a dilated filter in conv4_x and conv5_x

Ablation for image resolution: Except the backbone network, image resolution is also the most critical factor that influences speed. A simple method is to feed the small-resolution images into backbone network for accelerating feature extraction. As shown in Table 3, the image is downsampled with ratios 1, 1/2 and 1/4 and then put the images into ResNet50 and ResNet18. The prediction output is directly upsampled to the original size. When using scaling ratio 0.25, the inference time is reduced a lot compared to the scale 1, but the segmentation accuracy also decreases by a large margin, from 68.3% to 63.3% with ResNet50, from 63.2% to 58.1% with ResNet18. When using scaling ratio 0.5, the FPS only increases by 15 and 10 compared to the 0.25 case, and the accuracy only decreases by 0.9% and 0.8% compared to the 1 case. In the following experiments, ResNet50 and ResNet18

Table 3 Accuracy and speed analysis of different image resolutions: ResNet50 and ResNet18 on Cityscapes validation dataset

Model	Scale	Time(ms)	Frame(fps)	mIoU(%)
ResNet50	1.0	11	91	68.3
ResNet50	0.5	8	125	67.4
ResNet50	0.25	7	140	63.3
ResNet18	1.0	5	200	63.2
ResNet18	0.5	4	250	62.4
ResNet18	0.25	4	260	58.1

Table 4 Performance comparison of each component in our DCNet on Cityscapes validation dataset

Method	mIoU(%)
RCN(ResNet50)	67.4
RCN(ResNet50) + FRM	69.1
RCN(ResNet50) + FRM + PCN	70.5
RCN(ResNet50) + FRM + PCN(LAM)	72.9
RCN(ResNet50) + FRM + PCN(LAM) + CFF	74.2
RCN(ResNet18)	62.4
RCN(ResNet18) + FRM	63.7
RCN(ResNet18) + FRM + PCN	65.4
RCN(ResNet18) + FRM + PCN(LAM)	68.6
RCN(ResNet18) + FRM + PCN(LAM) + CFF	69.7

RCN: Region Context Network. The bracket represents the backbone network; FRM: Feature Re-WEIGHTING MODULE; PCN: Pixel Context Network; LAM: location attention module; **CFF**: contextual feature fusion

with scaling ratio 0.5 are taken as our basic unit to test the performance of the other components.

Ablation for Feature Re-weighting Module: To expand the receptive field of low-level feature maps in Region Context Network, a feature re-weighting module is designed. This module is connected to the output feature of the conv2_x layer to help upsampling. As shown in Table 4, the feature re-weighting module enhances consistency of the low stage (Fig. 5). This improves the performance from 67.4 to 69.1% and 62.4 to 63.7% over evaluation.

Ablation for Location Attention Module: In order to recover the detail information, a Pixel Context Network is introduced. The core part of this network is the location attention module, which captures long-range relations of each pixel. As shown in Table 4, the PCN without LAM can improve the performance by about 1%. However, the PCN with LAM makes the whole network more accurate, 3.8% increase of the ResNet50 and 4.9% increase of the ResNet18, which indicates the effect of this module. Figure 6 shows that the PCN with LAM can recover more detail information, e.g., some traffic signs, poles and the object boundary.

Ablation for contextual feature fusion: After obtaining the output features from these two sub-networks, it is necessary to fuse them effectively. Considering that these two sub-networks focus on different levels of context, region level for RCN and pixel level for PCN, the contextual feature fusion is proposed. As shown in Table 4, the effect of this design and a simple sum are evaluated. The mIoU is increased to 74.2% and 69.7%, respectively, which explains the features of the two sub-networks belong to different levels.

4.3 Comparison with the state-of-the-arts

In this subsection, in order to have a better understanding of the limitation and benefit of the proposed model, DCNet is compared with other popular models on Cityscapes, CamVid and COCO-Stuff. Our model is tested on a single GTX 2080 Ti GPU and reports the average time of all testing images. In the test process, there is no any testing augmentation.

Baselines: The baselines are categorized as full-precision models and efficient models. The full-precision models refer to the popular models that have good performance but longer running time. Specifically, the baseline full-precision models include: PSPNet [16], DenseASPP [17], DANet [26], Deeplabv3 + [25], Panoptic-DeepLab[46] and FNA + + [47]. Efficient models refer to some lightweight networks with real-time inference speed, such as SegNet [1], ENet [2], ESPNet [3], ICNet [5], TwoColumn [10], BiSeNet [4], DFANet [6], SwiftNet [20] and SFNet [48]. It should be noted that there are different structures even for the same models. For example, BiSeNet has Xception39 and ResNet18 backbones, and DFANet has backbone A and backbone B. Also. More specifically, for Panoptic-Deeplab[46] the quoted mIoUs are obtained with different training data, and for FNA + + [47] the quoted mIoUs are obtained with DeepLabv3 and DeepLabv3 +, respectively. The proposed models also have two backbones: ResNet50 and ResNet18.

Cityscapes: The DCNet is evaluated against the state-of-the-art on Cityscapes firstly. To have a more straightforward comparison for all methods, the mIoU and FPS are presented in Table 5 and Fig. 1.

1. From an accuracy perspective, the performance of a full-precision model is still superior to that of an efficient model. The best full-precision model is Panoptic-DeepLab, and 84.2% can be obtained with extra training data (Mapillary Vistas); the worst result of the full-precision models is 78.4% (PSPNet), which is about 2% higher than the best result of the efficient model (76.1% from ours). However, the inference speed of the full-precision models is very slow, and the DCNet is 70 and 121 times faster than DenseASPP, which owns fastest speed in full-precision models.
2. From an efficient perspective, our DCNet is very competitive. As can be observed, based on the ResNet50, the accuracy performance of the proposed method outperforms state-of-the-art methods, while the inference speed is kept comparable. It achieves mIoU 76.1% with 82 FPS inference speed on Cityscapes test set, which is very close to the competitive baseline model SFNet (DF1) in terms of both efficiency and accuracy. Additionally, when the backbone network is decreased to ResNet18, the inference speed of DCNet is increased to 142 FPS corresponding with still mIoU 71.2%, which is comparable with the previous methods DFANet A. Our model has

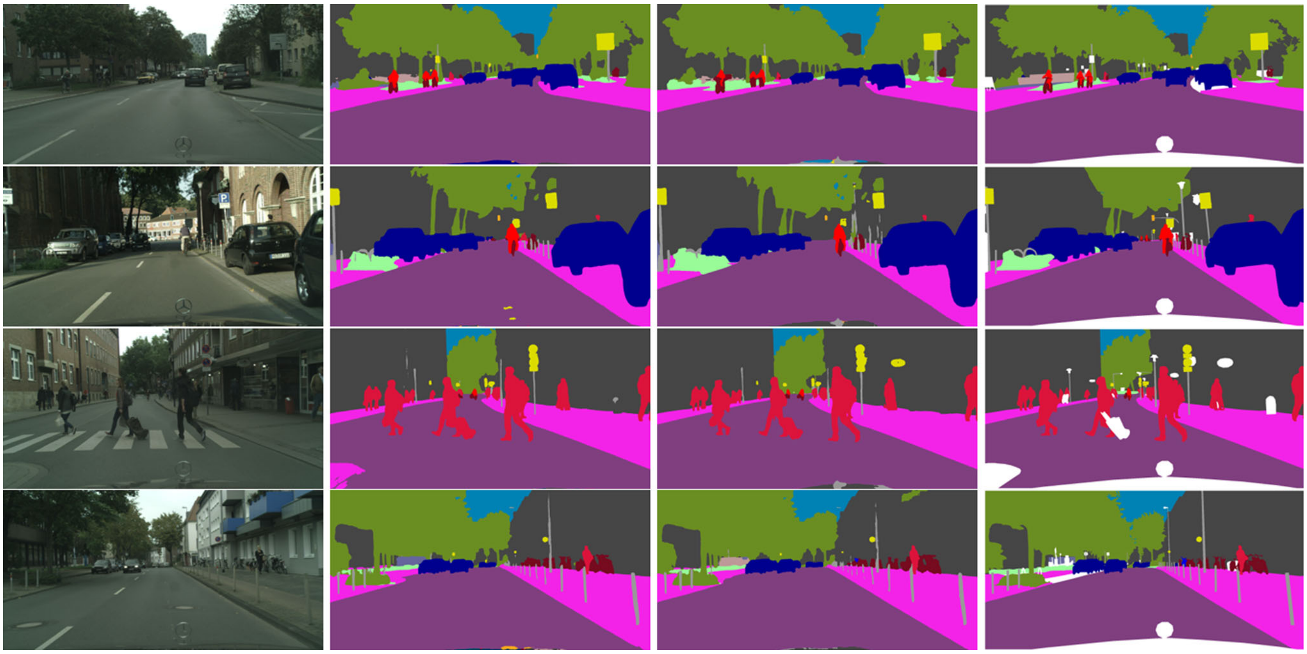


Fig. 5 Results of DCNet on Cityscapes validation set. Each line is followed by input images, output of ResNet50, output of ResNet18 and ground truth

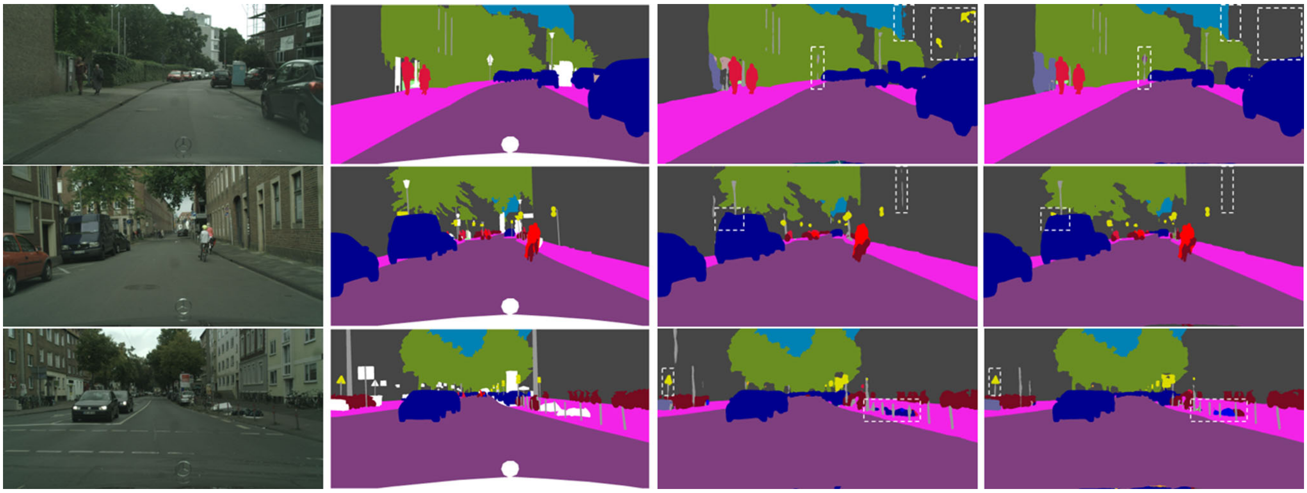


Fig. 6 Example results of the output before adding the Pixel Context Network and after adding the Pixel Context Network. Each line is followed by input images, ground truth, output without PCN and output with PCN

Table 5 Accuracy and speed analysis on Cityscapes test dataset. In our method, the small size represents the input of the Region Context Network and the large size represents the input of the Pixel Context Network

Model type	Method	Input Size	Time(ms)	Frame(fps)	mIoU(%)
Full-precision models	PSPNet[16]	713 × 713	1288	0.78	78.4
	DenseASPP[17]	1024 × 2048	850	1.17	80.6
	DANet[26]	1024 × 2048	4500	0.22	81.5
	Deeplabv3 + [25]	1024 × 2048	2800	0.35	82.1
	Panoptic-DL[46]	1025 × 2049	462	2.16	79.4/84.2
	FNA + + [47]	1024 × 2048	987	1.01	78.4
Efficient models	SegNet[1]	640 × 360	60	16.7	57
	ENet[2]	640 × 360	7	135.4	57
	ESPNet[3]	1024 × 512	9	112	60.3
	ICNet[5]	1024 × 2048	33	30.3	69.5
	TwoColumn[10]	512 × 1024	68	14.7	72.9
	BiSeNet1[4]	768 × 1536	13	72.3	68.4
	BiSeNet2[4]	768 × 1536	21	45.7	74.7
	DFANet A[6]	1024 × 1024	10	100	71.3
	DFANet B[6]	1024 × 1024	8	120	67.1
	SwiftNet[20]	1024 × 2048	25	39.9	75.5
	SFNet(DF1)[48]	1024 × 2048	11	90.9	74.5
	Ours(Res50)	512 × 1024 (1024 × 2048)	12	83.3	**76.1**
	Ours(Res18)	512 × 1024 (1024 × 2048)	7	**142.9**	71.2

The best results are marked as bold

(For the speed comparison, all the models are evaluated using a single Nvidia 2080Ti GPU; for Panoptic-DL [46], the inference time evaluation is based on the detectron2 implementation of R52-DC5 backbone; for FNA + + [47], the inference time evaluation is based on the DeepLabv3 adapted model; for SFNet [48], the DF1 backbone is evaluated.)

two resolutions of input image, and 512 × 1024 input is fed into the Region Context Network and only downsampled 8 times. The Pixel Context Network receives 1024 × 2048 input; although the resolution is high, there are a few layers in this sub-network. Some visual results of the proposed DCNet are shown in Fig. 5. Our model can produce high-performance results on Cityscapes using the proposed method.

Other datasets: The proposed DCNet is tested on CamVid dataset which contains images with a resolution of 720 × 960. The same setting is adopted as [28]. According to Table 6, our model can achieve the mIoU of 70.8% and the FPS of 91 based on ResNet50. Also, it achieves 66.2% mIoU and 166 FPS while using ResNet18. These results support that the DCNet is good at the street scene dataset. Additionally, Table 7 shows the testing results of COCO-Stuff dataset, and the input image sizes are different for different models. DCNet still performs satisfyingly regarding common thing and stuff understanding. The experiments verify the effectiveness of our method on different datasets.

Table 6 Accuracy and speed analysis on CamVid test dataset on resolution 720 × 960. "-" indicates that the corresponding result is not provided by the method

Model type	Method	Frame(fps)	mIoU(%)
Efficient models	SegNet[1]	46	46.4
	ICNet[5]	27.8	67.1
	ENet[2]	–	51.3
	BiSeNet1[4]	–	65.6
	BiSeNet2[4]	–	68.7
	DFANet A[6]	120	64.7
	DFANet B[6]	160	59.3
	Ours(Res50)	91	**70.8**
	Ours(Res18)	**166**	66.2

The best results are marked as bold

Table 7 Accuracy and speed analysis on COCO-Stuff test dataset on resolution 640 × 640

Model type	Method	Frame(fps)	mIoU(%)
Full-precision models	FCN[31]	5.9	22.7
	DeepLab[32]	8.1	26.9
	PSPNet[16]	6.6	**32.6**
Efficient models	ICNet[5]	35.7	29.1
	Ours(Res50)	**101**	29.8

The best results are marked as bold

5 Conclusion

In this paper, a novel Dual Context Network is proposed for real-time semantic segmentation which improves the performance of accuracy and speed. The DCNet contains two sub-networks: Region Context Network (RCN) and Pixel Context Network (PCN). The RCN utilizes the region context to obtain the semantic prediction, and the PCN recovers the detail information with relations of each pixel. The results on Cityscapes, CamVid and COCO-Stuff datasets are presented to prove the effectiveness of our method. Furthermore, even though the efficient method is proposed in the context of semantic segmentation, its dual encoder–decoder structure and lightweight design can be easily extended to other related applications, such as salient object detection [50, 51]. In addition, weakly supervised object localization and segmentation [54–56] are two interesting yet challenging tasks, and how to extend the proposed method to a weakly supervised learning framework is going to be investigated in our future work.

References

1. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell **39**(12), 2481–2495 (2017)
2. Paszke, A., Chaurasia, A., Kim,S., Culurciello, E.: Enet: a deep neural network architecture for real-time semantic segmentation (2016)
3. Mehta, S., Rastegari, M., Caspi, A., Shapiro, L., Hajishirzi, H.: Espnet: efficient spatial pyramid of dilated convolutions for semantic segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 552–568 (2018)
4. Yu, C., Wang, J., Peng, C. Gao, C., Yu, G., Sang, N.: Bisenet: bilateral segmentation network for real-time semantic segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 325–341 (2018)
5. Zhao, H., Qi, X., Shen, X., Shi, J., Jia, J.: Icnet for real-time semantic segmentation on high-resolution images. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 405–420 (2018)
6. Li, H., Xiong, P., Fan, H., Sun, J.: Dfanet: deep feature aggregation for real-time semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9522–9531 (2019).
7. 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)
8. Cordts, M. et al.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3213–3223 (2016)
9. Brostow, G.J., Fauqueur, J., Cipolla, R.J.: Semantic object classes in video: a high-definition ground truth database. Pattern Recognit. Lett. **30**(2), 88–97 (2009)
10. Wu, Z., Shen, C., Hengel, A.: Real-time semantic image segmentation via spatial sparsity (2017)
11. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 234–241. Springer (2015)
12. Liu, W., Rabinovich, A., Berg, A. C.: Parsenet: Looking wider to see better (2015)
13. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7794–7803 (2018)
14. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
15. Chen, L.-C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation (2017)
16. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2881–2890 (2017).
17. Yang, M., Yu, K., Zhang, C., Li, Z.,. Yang, K: Denseaspp for semantic segmentation in street scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3684–3692 (2018)
18. Yuan, Y., Wang, J.: Ocnet: Object context network for scene parsing (2018)
19. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Learning a discriminative feature network for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1857–1866 (2018)
20. Orsic, M., Kreso, I., Bevandic, P., Segvic, S.: In defense of pre-trained imagenet architectures for real-time semantic segmentation of road-driving images. In: Proceedings of the IEEE Conference

on Computer Vision and Pattern Recognition, pp. 12607–12616 (2019)

21. Lin, Z. et al.: A structured self-attentive sentence embedding (2017)
22. Romera, E., Alvarez, J.M., Bergasa, L.M., Arroyo, R.: "Erfnet: efficient residual factorized convnet for real-time semantic segmentation. IEEE Trans. Intell. Transp. Syst. **19**(1), 263–272 (2017)
23. Howard A. et al.: Searching for mobilenetv3. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1314–1324
24. Chollet, F.: Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1251–1258 (2017)
25. Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV), pp. 801–818 (2018)
26. Fu, J. et al.: Dual attention network for scene segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3146–3154 (2019)
27. Zhang, H. et al.: Context encoding for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7151–7160 (2018)
28. Sturgess, P., Alahari, K., Ladicky, L., Torr, P. H.: Combining appearance and structure from motion features for road scene understanding (2009)
29. 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)
30. Lin, T.-Y. et al.: Microsoft coco: common objects in context. In: European Conference on Computer Vision, pp. 740–755. Springer (2014)
31. 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)
32. 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 Trans. Pattern Anal. Mach. Intell. **40**(4), 834–848 (2017)
33. Wang, P. et al.: Understanding convolution for semantic segmentation. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1451–1460, IEEE (2018)
34. Ding, H., Jiang, X., Shuai, B., Qun Liu, A., Wang, G.: Context contrasted feature and gated multi-scale aggregation for scene segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2393–2402 (2018)
35. Ma, N., Zhang, X., Zheng, H.-T., Sun, J.: Shufflenet v2: practical guidelines for efficient CNN architecture design. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 116–131 (2018)
36. Pohlen, T., Hermans, A., Mathias, M., Leibe, B.: Full-resolution residual networks for semantic segmentation in street scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4151–4160 (2017)
37. Pan, X., Shi, J., Luo, P., Wang, X., Tang, X.: Spatial as deep: Spatial CNN for traffic scene understanding. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
38. Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., Liu, W.: Ccnet: Criss-cross attention for semantic segmentation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 603–612 (2019)
39. Li, H., Xiong, P. An, J., Wang, L.: Pyramid attention network for semantic segmentation. arXiv 2018
40. Zheng, S. et al.: Conditional random fields as recurrent neural networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1529–1537 (2015)
41. Liu, Y., Cao, S., Lasang, P., Shen, S.: Modular lightweight network for road object detection using a feature fusion approach. IEEE Trans. Syst. Man Cybern. Syst. pp 1–13 (2019)
42. Khelifi, L., Mignotte, M.: A novel fusion approach based on the global consistency criterion to fusing multiple segmentations. IEEE Trans. Syst. Man Cybern. Syst. **47**(9), 2489–2502 (2017)
43. Yuan, X., Cao, X., Hao, X., Chen, H., Wei, X.: Vehicle detection by a context-aware multichannel feature pyramid. IEEE Trans. Syst. Man Cybern. Syst. **47**(7), 1348–1357 (2017)
44. Li, Y., Guo, Y., Kao, Y., He, R.: Image piece learning for weakly supervised semantic segmentation. IEEE Trans. Syst. Man Cybern. Syst. **47**(4), 648–659 (2017)
45. Si, J., Zhang, H., Li, C., Guo, J.: Spatial pyramid-based statistical features for person re-identification: a comprehensive evaluation. IEEE Trans. Syst. Man Cybern. Syst. **48**(7), 1140–1154 (2018)
46. Chen, B., Collins, M., Zhu, Y., Liu, T., Huang, T., Adam, H., Chen, L.: Panoptic-deeplab: a simple, strong, and fast baseline for bottom-up panoptic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2020)
47. Fang, J., Sun, Y., Zhang, Q., Peng, K., Li, Y., Liu, W., Wang, X.: FNA++: fast network adaptation via parameter remapping and architecture search. IEEE Trans. Pattern Anal. Mach. Intell. **43**(9), 2990–3004 (2021)
48. Li, X., You, A., Zhu, Z., Zhao, H., Yang, M., Yang, K., Tan, S., Tong, Y.: Semantic flow for fast and accurate scene parsing. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)
49. Fang, C., Tian., H., Zhang, D, Zhang, Q., Han, J., Han, J.: Densely nested top-down flows for salient object detection. Sci. China Inf. Sci. (2022)
50. Liu, N., Han, J.: DHSNet: deep hierarchical saliency network for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)
51. Liu, N., Han, J., Yang, M.: PiCANet: learning pixel-wise contextual attention for saliency detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018)
52. Zhao, J., Liu, J.-J., Fan, D.-P., Cao, Y., Yang, J., Cheng, M.-M.: EGNet: Edge guidance network for salient object detection. In: IEEE/CVF International Conference on Computer Vision (ICCV), pp. 8779–8788 (2019)
53. Liu, N., Zhao, W., Shao, L., Han, J.: SCG: saliency and contour guided salient instance segmentation. IEEE Trans. Image Process. **30**, 5862–5874 (2021)
54. Zhang, D., Zeng, W., Guo, G., Fang, C., Cheng, L., Han. J.: Weakly supervised semantic segmentation via alternative self-dual teaching. arXiv preprint arXiv:2112.09459 (2021)
55. Zhang, D., Zeng, W., Yao, J., Han, J.: Weakly supervised object detection using proposal-and semantic-level relationships. IEEE Trans. Pattern Anal. Mach. Intell. **44**(6), 3349–3363 (2022)
56. Zhang, D., Han, J., Yang, L., Xu, D.: SPFTN: a joint learning framework for localizing and segmenting objects in weakly labeled videos. IEEE Trans. Pattern Anal. Mach. Intell. **42**(2), 475–489 (2020)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Hong Yin was born in 1967 and received the Ph.D. degree from Nanjing University of Science and Technology in 2007, Nanjing, China, in 2017. His current research interests include pattern recognition, deep learning and semantic segmentation.

Wenbin Xie was born in 1980 and received the Ph.D. degree from PLA University of Science and Technology, Nanjing, China, in 2007; his research interests include computer vision and combat experiments.

Jingjing Zhang is an engineer at the Naval Research Academy. She was born in 1979 and graduated from the Naval Engineering University in 2018 with a Ph.D. His research interests include systems engineering, communications engineering and information.

Yuanfa Zhang was born in 1989. He obtained his bachelor's degree from PLA National University of Defense Technology in 2011. At present, he serves as a mid-level engineer at Naval Research Academy. His research focuses on command automation and computer science.

Weixing Zhu was born in 1978. He received the M.S. degree in 2006 and Ph.D. degree in 2012 from PLA University of Science and Technology. Now, he is an associate professor of Army Engineering University of PLA. His main research focuses on Big Data, Military Requirements and Software Engineering.

Jie Gao received the master degree of Computer Technology from Academy of Armored Force Engineering, Beijing, China, in 2020. Her research direction is information technology processing.

Yan Shao received the B.A. degree in the Art of Broadcasting and Presentation from Communication University of China, Nanjing, in 2010. Her current research interests include data research and arrangement.

Yajun Li received the B.E. degree in school of information engineering form Nanjing XiaoZhuang University, Nanjing, China, in 2017. He is currently pursuing the M.E. degree in computer science from the Nanjing University of Science and Technology, Nanjing, China. His current research interests include pattern recognition, deep learning and semantic segmentation.