SCIENCE CHINA
Information Sciences

February 2020, Vol. 63 120104:1–120104:16
https://doi.org/10.1007/s11432-019-2718-7

• RESEARCH PAPER •
Special Focus on Deep Learning for Computer Vision

CGNet: cross-guidance network for semantic segmentation

Zhijie ZHANG & Yanwei PANG*

School of Electrical and Information Engineering, Tianjin University, Tianjin 300072*, China*

Received 16 June 2019/Revised 13 November 2019/Accepted 29 November 2019/Published online 16 January 2020

Abstract Semantic segmentation is a fundamental task in image analysis. The issue of semantic segmentation is to extract discriminative features for distinguishing different objects and recognizing hard examples. However, most existing methods have limitations on resolving this problem. To tackle this problem, we identify the contributions of the edge and saliency information for segmentation and present a novel end-to-end network, termed cross-guidance network (CGNet) to leverage them to benefit the semantic segmentation. The edge and saliency detection network are unified into the CGNet, and model the intrinsic information among them, guiding the process of extracting discriminative features. Specifically, the CGNet attempts to extract segmentation, edge, and salient features, simultaneously. Then it transfers them into the cross-guidance module (CGM) to generate the pre-knowledge features based on the modeled information, optimizing the context feature extraction process. The proposed approach is extensively evaluated on PASCAL VOC 2012, PASCAL-Person-Part, and Cityscapes, and achieves state-of-the-art performance, demonstrating the superiority of the proposed approach.

Keywords semantic segmentation, fully convolutional networks, pyramid network, edge detection, saliency detection, cross-guidance

Citation Zhang Z J, Pang Y W. CGNet: cross-guidance network for semantic segmentation. Sci China Inf Sci, 2020, 63(2): 120104, https://doi.org/10.1007/s11432-019-2718-7

1 Introduction

Semantic segmentation, which involves classifying pixels into predefined categories, is used to predict the category, location of objects in an image. Thus, it plays an essential role in many applications [1], such as surveillance analysis, automatic driving, medical imaging analysis, action recognition, virtual fitting, and image/video retrieval.

Recently, semantic image segmentation methods based on fully convolutional networks (FCNs) [2] have achieved significant progress in segmentation performance, compared to traditional methods. Some representative ones are built upon well-designed spatial pyramid pooling (SPP) [3] module (e.g., PSPNet [4], DeepLab-v2 [5], DeepLab-v3 [6], DeepLab-v3+ [7]). However, these methods severely depend on the robustness of multiple receptive fields representations and lack enough object detail information, which leads to trivial segmentation regions and discontinuous boundaries.

Specifically, the semantic segmentation methods utilize global appearance models of foregrounds and backgrounds to identify the target regions, which preserves the homogeneity and semantic characteristics of objects. However, without fully utilizing edge information, most existing semantic segmentation methods have difficulty in reducing the uncertainties in detecting the boundary positions owing to the

* Corresponding author (email: pyw@tju.edu.cn)

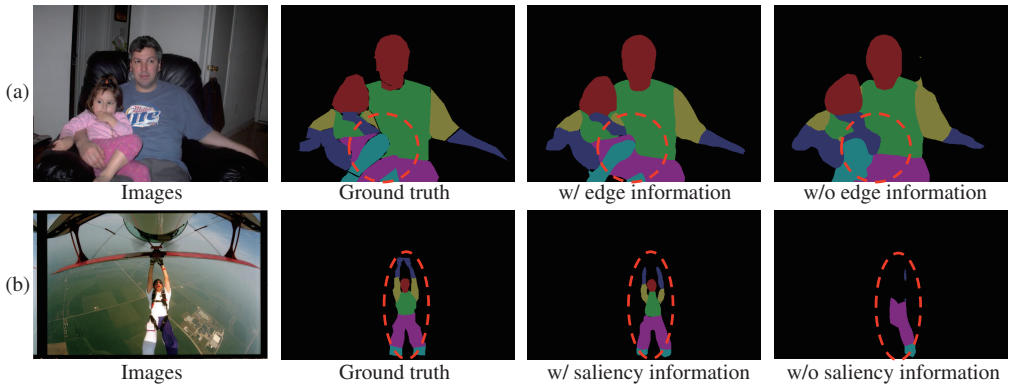


Figure 1 (Color online) Examples of segmentation results, using different settings on whether to utilize edge (a) or salient object (b) information.

similar appearances of the neighboring objects. They also need to apply additional time-consuming post-processing technologies (e.g., conditional random field (CRF) [8]) to refine the initial coarse segmentation results. In addition, hard examples or even easy examples are sometimes overshadowed by irrelevant objects, which can confuse the deep convolutional neural networks (DCNNs), because they are dominated by conspicuous objects during training, making them inconspicuous and difficult for the network to mine features on them. This problem can also be more critical when there are only a few training examples. Although there are methods try to alleviate this problem using well-designed loss function [9], or utilizing online hard example mining (OHEM) [10], we proposed to alleviate this problem from a feature perspective.

As shown in red circles of Figure 1(a), we found that it is not easy to make a distinction between the lower leg and the upper leg, or the arm and the leg without edge information, while utilizing the edge information can distinguish them. Illustration in Figure 1(b) shows that the salient objects sometimes can be recognized as inconspicuous objects when there is no saliency information. However, utilizing saliency information enables avoiding this problem. Herein, there is shared intrinsic information among the segmentation, edge, and saliency information: the edge information can provide explicitly constrain object configurations for segmentation features, while the saliency information makes inconspicuous objects properly emphasized.

Inspired by the above phenomenon, we propose a novel approach, called cross-guidance network (CGNet), to integrate the segmentation, edge, and saliency information into a unified network. There are two main sequential steps for the CGNet to segment objects, generating different features (i.e., segmentation features, edge features, salient objects features), facilitating the edge and salient features to guide the extraction of discriminative context features in a novel way, and thus enhancing segmentation features. In other words, the main contribution of the proposed method lies in the cross-guidance module. Although there are methods [11] using salient or edge information to assist segmentation, they all adopt multiple task fashion or directly fuse the input features with the edge or salient features. In our study, edge features and salient features are mutually constrained, and the extracted features are enhanced with the interdependent features, simultaneously. This way can better model the intrinsic relationship between the extracted features and the salient, edge information, and thus guide the feature extraction process. The proposed integration network yields more accurate segmentation results with a fast inference speed of 26 fps on a TITAN X (Pascal) GPU.

The main contributions of this study are three-fold:

- We reveal and utilize the intrinsic relationship among the segmentation, edge, and saliency information in the semantic segmentation task.
- The CGNet is proposed to model the segmentation, edge, and saliency information, guiding the process of extracting discriminative features and enhancing the segmentation results.
- The proposed approach achieves the state-of-the-art performance on the PASCAL VOC 2012 and PASCAL-Person-Part dataset.

2 Related work

Generally speaking, modern semantic segmentation methods all stem from FCNs [2]. These methods have achieved significant improvements in segmentation performance.

In semantic segmentaiton, a backbone network is the basis of extracting compact features. Popular backbone networks in semantic segmentation include GlaceNet [12], VGGNet [13], ResNet [14], DenseNet [15], and Xception [16]. Early studies in semantic segmentation widely used VGGNet to extract high-level features (e.g., FCN [2], SegNet [17], DeconvNet [18]). However, with an increasing demand for accuracy, VGGNet often has difficulty in meeting the requirements on larger segmentation datasets. To this end, ResNet [14] was introduced to semantic segmentation. This backbone reduces the probability of gradient disappearance during backpropagation, enabling it to generate more representative features. As such, it has been widely applied as the backbone network in prevalent semantic segmentation methods (e.g., DeepLab-v2 [5], PSPNet [4], DeepLab-v3 [6], DRN [19], RefineNet [20], EncNet [21], CCNet [22]). DenseNet (e.g., Tiramisu [23], DenseASPP [24]) and Xception (e.g., DeepLab-v3+ [7]) appear to be other popular backbone networks in recent methods. Taking efficiency into account, our approach is built upon the mostly used ResNet with a stride of 16, which is more applicable than methods using stride 8 or bigger.

In addition to the design of backbone network, a pyramid network has been approved to be an efficient way to deliver features with detailed information. Many state-of-the-art segmentation methods are based on SPP [3], which adopts parallel convolutional layers to generate features with different receptive fields (e.g., EncNet [21], PSPNet [4], DeepLab [5–7], DenseASPP [24]). In contrast to SPP-based methods, many other approaches (e.g., ExFuse [25], SegNet [17], ICNet [26], RefineNet [20], PAN [27], GCN [28], and [18, 23, 29]) attempt to employ a step-by-step decoder network in the form of a cascaded pyramid structure, to recover object details. Furthermore, Ref. [30] employs image pyramid network to extract contextual features for prediction.

Some efforts have been made to take advantage of extra information, such as object classification (e.g., EncNet [21]), which has a potential correlation with semantic segmentation. We find that there are two tasks related to semantic segmentation: edge detection and saliency detection. The edge detection methods can first identify object boundaries utilizing local gradient representations, and then separate the closed loop regions as the objects. Recently, deep convolutional networks methods [11, 31–34] employ holistically nested topology to solve this task, especially the HED [32], DeepContour [35]. However, these methods consume a high computational cost. Instead of using edge detection methods mentioned above, we propose an efficient way to detect object contours, slightly increasing the computational cost. We also follow the pix2pixHD [36] to generate edge labels for supervision.

As for saliency detection, it is highly related to semantic segmentation, because it aims to assign each pixel to different pre-defined categories, and most recent state-of-the-art saliency detection approaches are also FCN-based. From [37], DHSNet [38] is the first FCN-based method for salient object detection. The methods [39–42] following DHSNet make significant progress on yielding more accurate and reasonable saliency detection results. In our method, we adopt a simple way to detect saliency. Furthermore, the proposed approach tries to explore the edge and saliency information when they are involved in semantic segmentation and find an efficient way to enhance segmentation performance.

3 Method

We propose a novel deep fully convolutional network for semantic segmentation, called CGNet. The key idea of CGNet is to guide the process of extracting discriminating features by properly modeling the segmentation, edge, and saliency information. The idea is implemented by stacking five parts: a backbone network, a pyramid attentive module, an edge detection head, a saliency detection head, and a cross-guidance module, and thus forms an end-to-end network for semantic segmentation. In this section, we will first describe the integral pipeline of the proposed network. Then, we will clarify the proposed

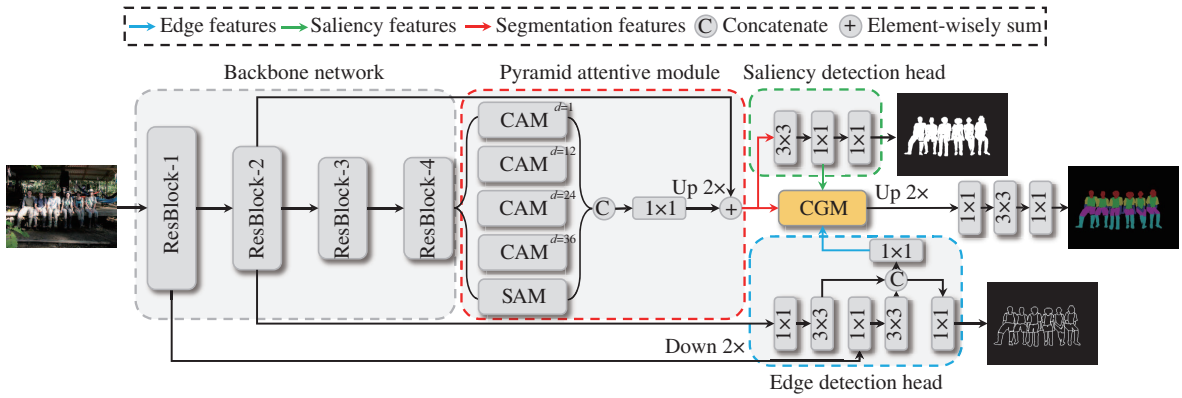


Figure 2 (Color online) Illustration of the proposed CGNet, which includes the main backbone network with a pyramid attentive module, a cross-guidance module (CGM), an edge detection head and a saliency detection head. 'ResBlock' denotes the residual convolutional block in ResNet [14], while '1×1', '3×3','d', 'Up', and 'Down' denote the convolutional layer with kernel size 1, convolutional layer with kernel size 3, dilated (atrous) rates of convolutional kernel, upsampling using non-parameterized bilinear interpolation, and downsampling, respectively. 'CAM' and 'SAM' refer to channel attentive module and spatial attentive module, respectively.

modules in detail. Finally, the loss function in CGNet will be elaborated.

3.1 Network architecture

The proposed CGNet is illustrated in Figure 2. Formally, given an input image $\boldsymbol{I} \in \mathbb{R}^{H \times W \times C}$, where H, W denote the height, and width of the image, a backbone network B (ResNet-101, with output stride 16, dilation rate 2 for the last residual block, is adopted in all our experiments) is first employed to extract the representation r_b of the image:

$$r_b^i = F_B(I; \boldsymbol{W}_B), \tag{1}$$

where $\boldsymbol{W}_B$ denotes the parameters of models, i represents the output features of i-th residual block in backbone network. For example, r_b^1 represents the outputs of ResBlock-1.

There are two streams for backbone features r_b: (1) The pyramid attentive module P is empolyed on the r_b^4 to extract segmentation features r_p:

$$r_p = F_P(r_b^4; \boldsymbol{W}_P) \in \mathbb{R}^{h \times w \times c}; \tag{2}$$

(2) The edge detection head E is built upon r_b^1 and r_b^2 to detect the edge of objects and obtain features with edge informaiton r_e:

$$r_e = F_E(r_b^1, r_b^2; \boldsymbol{W}_E) \in \mathbb{R}^{h \times w \times c}. \tag{3}$$

In addition, as we have the robust segmentation features r_p, the saliency detection head S can be built upon it to detect the salient objects and obtain saliency features r_s:

$$r_s = F_S(r_p; \boldsymbol{W}_S) \in \mathbb{R}^{h \times w \times c}. \tag{4}$$

With these features, the cross-guidance module (CGM) is easy to integrate the edge information and the saliency information for modeling the correlation with segmentation information, and thus guide the network to extract discriminative features. We formulate the guidance-based features r_{cgm} as

$$r_{\text{cgm}} = F_{\text{CGM}}(r_p, r_e, r_s; \boldsymbol{W}_{\text{CGM}}) \in \mathbb{R}^{h \times w \times c}. \tag{5}$$

Details of these modules will be introduced in Subsection 3.2. With the guidance-based features, we can generate more accurate segmentation results. Notably, the network is unified, resulting in a streamlined system that better lends itself to fast processing.

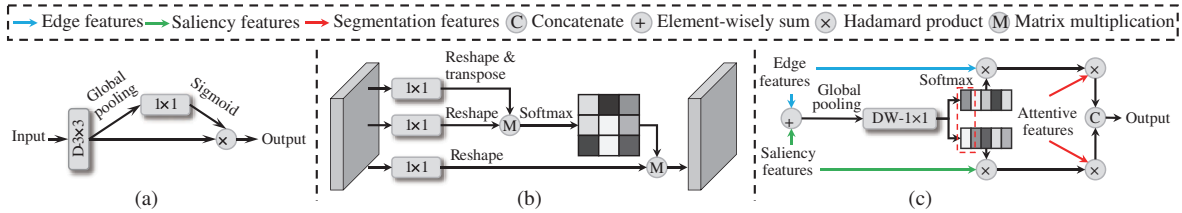


Figure 3 (Color online) Illustration of the proposed modules. '1 × 1', '3 × 3', 'D-3 × 3' and 'DW-1 × 1' denote the convolutional layer with kernel size 1, convolutional layer with kernel size 3, dilated convolutional layer [19] with kernel size 3, and depth-wise convolutional layer [16] with kernel size 1, respectively. (a) Channel attentive module; (b) spatial attentive module; (c) cross-guidance module.

3.2 Method details

Pyramid attentive module. For the task of semantic segmentation, most modern approaches consider enhancing the feature representation ability by exploring contextual information. To achieve this goal, these methods attempt to enlarge the receptive fields and aggregate the multiple receptive fields features. A pyramid network with different receptive fields has been approved to be an efficient way to extract context information. PSPNet [4] and ASPP [5] are the mostly used pyramid networks for obtaining contextual information. However, owing to the lack of attention on what is necessary for the local and global context features, we argue that the context information they extracted does not have a robust representation for more challenging segmentation tasks.

To remedy this problem, we propose a pyramid attentive module to simultaneously capture the local and global context information, and extract more representative features. As shown in Figure 2, four parallel dilated convolutional layers [19] (shown in Figure 3(a)) with different dilated rates (i.e., 1, 12, 24, 36) are used to extract local context information in multiple receptive fields. Furthermore, an attention mechanism (shown in Figure 3(a)) based on SENet [43], is applied to each output along the channel dimension to focus on the valuable context information for prediction. This is done because not all high-level layers are activated during training. Using the channel attentive module enables reducing the impacts of the irrelevant information.

It is noted that DANet [44] proposed the channel attention module (CAM) and position attention module (PAM). These two modules are based on self-attention mechanism. In our work, the SAM is based on self-attention mechanism. All methods which adopt self-attention mechanism in spatial dimension are similar. Different with PAM, our SAM adopts one convolution in the end to generate the output rather than fusion with input used in PAM. However, our CAM is based on SE module. The CAM adopts dilated convolution as the ASPP does to extract features, and then the learned weights of each channel multiply with the extracted features to achieve the channel attention mechanism. The pyramid form of the SAM and CAM benefits the utilization of attention mechanism, and it is totally different from the DANet.

In addition to the local extractor based on the channel attention, we integrate the spatial attentive module inspired by non-local [45], to the pyramid network to capture the global context information. As shown in Figure 3(b), the input features are first passed through a 3×3 convolution to arrange the features, and then the spatial matrix operation, which reshapes and transposes the features shape from $h \times w \times c$ to $(h \cdot w) \times c$ and $c \times (h \cdot w)$, is applied on them to get $(h \cdot w) \times (h \cdot w)$ features. The following softmax operation is employed on the reshaped features to conduct the pixel relationship map. The matrix multiplication is then employed on the pixel relationship map with its shape being $(h \cdot w) \times (h \cdot w)$ and the input features with shape $(h \cdot w) \times c$ to capture the spatial attention relationship, and thus extract features with global context information. Next, all the parallel outputs are concatenated, and then passed through one 1×1 convolutional layer to recombine the local and global context features. Finally, the attentive features element-wisely sum with r_b^2, which complements for the loss of position information of objects, caused by the sequential downsampling operation in the backbone network.

Edge detection head. In the semantic segmentation, preserving edge information benefits discriminating objects when objects are too similar to be distinguished. The problem is how to utilize the edge

information to sharp and refine the prediction.

Most edge detection methods (e.g., HED [32]) are built upon heavy networks, resulting in high computational cost. In contrast to these methods, the proposed edge detection head is a light-weight network and aims to generate edge features under the edge supervision, not the edge predictions. Herein, we integrate the edge detection head to the proposed approach to explore the utilization of edge information in semantic segmentation. As well known, edge information mainly exists in low-level layers. As a result, we propose to utilize the low-level information to extract edge representation. It can be seen that the proposed edge detection network shares the most of convolutional layers with the segmentation network.

As shown in Figure 2, the edge detection head is built-upon top of the layers of the first two residual blocks. Features r_b^1 and r_b^2 are first resized to the same resolution. The followed convolutional layers are used to calibrate the channel number of the features for concatenating. There are two streams for the concatenated features; one is passed through one 1×1 convolutional layer for edge supervision, and the other is passed through one 1×1 convolutional layer for feature transfer. With the edge supervision, the transferring features r_e contain more valuable edge information, which enables the guidance for final prediction.

Saliency detection head. In semantic segmentation, data imbalance is a fatal problem, which leads to unreasonable segmentation of objects with hard examples. Methods like Focal loss [9] and OHEM [10], try to alleviate this problem in the loss function. In this paper, we propose to alleviate this problem from a feature perspective.

Hard objects are difficult to be recognized owing to the lack of sufficient examples and would be dominated by irrelevant objects. Current methods always make the conspicuous objects to inconspicuous objects, especially, when objects are associated with small regions, so the irrelevant background pixels would severely affect the recognition of them. To remedy this problem, we identify that saliency detection enables reducing the data imbalance problem, owing to fair emphasis on all objects. With saliency detection, the network can pay attention to the hard examples, and thus we can extract their features to provide the prior knowledge for the network, guiding the features mining process.

Most saliency detection methods (e.g., [46]) integrate multiple levels outputs to recover the saliency information, and thus rely on complicated networks. In contrast to them, we capture the saliency information from the parallel pyramid outputs, and share the same network with the segmentation task, resulting in a light-weight and efficient saliency detection network.

As shown in Figure 2, we append the saliency detection head to the pyramid attentive module. The features r_p are sequentially processed by a 3×3 convolution for extracting salient features, a channel calibrated 1×1 convolution, and a 1×1 convolution for supervising saliency. The extracted features r_s are also transferred to the cross-guidance module to enhance the context information extraction of hard examples.

Cross-guidance module. Although edge and saliency information are both beneficial to the context feature extraction, it is challenging to appropriately model the correlation of them and fully utilize them to improve segmentation performance. Merely applying multi-task learning contributes little to the final segmentation. Therefore, we propose a module, called CGM, to associate segmentation, edge, and saliency information to pixel-level segmentation.

It is a natural idea to fuse edge and salient features with segmentation features and then add several extra convolutional layers. However, features may be dominated by one type features with the extremely high response, making harm on prediction. For example, edge information could make the network only focus on the edge of objects, resulting in degrading on predicting whole objects. In this situation, we leverage the edge and saliency information to conduct the intrinsic correlation, making them play a different role in guiding feature extraction.

As illustrated in Figure 3(c), (6), and (7), the CGM first element-wisely sums the edge features and the saliency features, and then performs global average pooling on the summation. Finally, the followed depth-wise 1×1 convolution is performed to generate the weights for different channels. The reason why we adopt depth-wise convolution is that, as stated in [47], features are class-specific in high-level layers, and thus the generated weight map should be based on itself, resulting in no need to build weights

upon the crossed channels. In order to model the mutual relationship between the edge and saliency information, we adopt the softmax mechanism to constrain the weights of the edge and saliency along the same channel. The actual implementation is to employ the sigmoid function (Eq. (7)) on the generated weight map to yield the edge weight map $\boldsymbol{W}_e$. As a result, $1 - \boldsymbol{W}_e$ is the saliency weight map. These two weight maps then multiply with their original features to generate edge and saliency features under the mutual influence. Finally, they separately multiply with segmentation features. The followed fusion operation on these two features achieves extracting cross-guidance-based features for final prediction. In this way, the intrinsic information among segmentation, edge, and saliency features can be implicitly modeled, making the network robust to segmenting object contours and hard examples.

$$r_{\text{segmentation}} = (\boldsymbol{W}_e \cdot r_e + \boldsymbol{W}_s \cdot r_s) \cdot r_p \in \mathbb{R}^{h \times w \times c}, \tag{6}$$

where

$$\begin{cases} \boldsymbol{W}_e = \delta(F_{DW}(f_{\text{gap}}(r_e + r_s))), \\ \boldsymbol{W}_s = 1 - \boldsymbol{W}_e. \end{cases} \tag{7}$$

Notably, the symbol δ denotes the sigmoid function.

3.3 Loss function

During training, the loss function is composed of three terms: the commonly used cross-entropy-based segmentation loss ($\mathcal{L}_{\text{seg}}$), the cross-entropy-based saliency detection loss ($\mathcal{L}_{\text{salient}}$) with the contribution weights λ_s, and the edge detection loss ($\mathcal{L}_{\text{edge}}$) with the contribution weights λ_e.

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{seg}} + \lambda_s \cdot \mathcal{L}_{\text{saliency}} + \lambda_e \cdot \mathcal{L}_{\text{edge}}. \tag{8}$$

The edge detection loss is defined as

$$\mathcal{L}_{\text{edge}} = -\frac{1}{N} \sum_{i=1} \Big(W_{\mathcal{P}} \cdot (\mathcal{T}_i \cdot \log \mathcal{O}_i) + W_{\mathcal{N}} \cdot (1 - \mathcal{T}_i) \cdot (\log(1 - \mathcal{O}_i)) \Big), \tag{9}$$

where N denotes the number of categories, $\mathcal{T}$ denotes the label values, $\mathcal{O}$ denotes the prediction probabilities, and

$$\begin{cases} W_{\mathcal{P}} = \dfrac{\mathcal{P} + \mathcal{N}}{\mathcal{P}}, & \mathcal{T}_i = 1, \\ W_{\mathcal{N}} = \dfrac{\mathcal{P} + \mathcal{N}}{\mathcal{N}}, & \mathcal{T}_i = 0. \end{cases} \tag{10}$$

$W_{\mathcal{P}}$ and $W_{\mathcal{N}}$, which are calculated by (10), mean the weights for positive and negative instances, respectively. In (10), $\mathcal{P}$ denotes the number of positive instances in the labels, while $\mathcal{N}$ is the number of negative instances. With these two weights, the imbalance problem during supervising edge is efficiently alleviated. Notably, λ_s and λ_e are empirically set to 1.0 and 0.4 during training.

4 Experiments

To evaluate the proposed method, we conduct comprehensive experiments on three main segmentation datasets: PASCAL VOC 2012 [48], PASCAL-Person-Part [49], and Cityscapes [50], which are for fine-grained semantic segmentation. In this section, we first introduce the datasets and experiment protocol, and then compare our results with state-of-the-art methods on three datasets. Finally, we analyze the proposed approach using ablation experiments on the PASCAL-Person-Part dataset.

4.1 Datasets

PASCAL VOC 2012, which contains 20 foreground object categories and one background category for semantic segmentation, consists of 1464 images for training, 1449 for validation and 1456 for testing. The training set is extended by the semantic boundaries dataset (SBD) [51], resulting in a total of 10582 training images.

PASCAL-Person-Part, a more difficult dataset for fine-grained semantic segmentation, contains multiple humans in each image in unconstrained poses and occlusions. It provides pixel-wise annotations of seven human-body categories and consists of 1716 images for training and 1817 images for testing.

The urban scene understanding Cityscapes dataset contains 5000 finely annoted images and 19998 coarsely annoted images. The finely annotated set is divided into three parts: 2975 training images, 500 validation images and 1525 testing images, with resolution of 2048×1024.

4.2 Experiment protocol

Our approach is implemented on the PyTorch framework, using four NVIDIA Tesla V100 GPUs. During training, the weights of the backbone network are loaded from ResNet-101 pre-trained on ImageNet, and the remaining layers are randomly initialized. For data preparation, we apply data augmentation techniques for all the training data, including randomly scaling (from 0.5 to 2.0), randomly cropping (513×513 for PASCAL VOC 2012 and PASCAL-Person-Part, 769×769 for Cityscapes), and randomly horizontal-flipping. For optimization, we apply the SGD with a momentum 0.9, and weight_decay 0.0005, and the 'poly' learning rate schedule is adopt, $\mathrm{lr} = \mathrm{base\_lr} \times (1 - \frac{\mathrm{iters}}{\mathrm{total\_iters}})^{\mathrm{power}}$, in which power = 0.9 and base_lr = 0.007. The total_iters is epochs × batch_size, where batch_size = 40 and epochs = 200 for PASCAL-Person-Part, 150 for PASCAL VOC 2012. For Cityscapes, the batch_size is set to 16, and the epochs equals to 240. We use multiple GPUs for the consumption of large batch_size, and implement synchronized cross-GPU batch normalization.

4.3 Experimental results

Following the standard protocol in semantic segmentation, pixel accuracy (pixAcc), mean intersection over union (mIoU, also known as IoU), instance weighted IoU class (iIoU cla.), and iIoU category (iIoU cat.) are adopted as the evaluation metrics for all the experiments.

It is noted that the proposed CGM aims to enhance the extracted features by the object saliency and edge information. Owing to the strong ability of CNNs, the CGM is able to learn a function that can find the best way to enhance the multi-channel features. In addition, the salient object must have edge ground truth in our settings. These two mutual features will learn the weights along the channel dimension. Because we use softmax to normalize these two learned weights (the corresponding channel weights of the edge and saliency are summed to 1), although some instances are not annotated as the salient object, the CGM does not reduce the quality of extracted features.

4.3.1 *Experiments on the PASCAL VOC* 2012 *dataset*

The proposed approach is first evaluated on the PASCAL VOC 2012 dataset. In semantic segmentation, the classes of objects can be divided into things and stuff, where things can be viewed as salient objects. In the PASCAL VOC 2012 dataset, there are not many object types in each image, and they are all things; as a result, we can directly select the objects marked in the segmentation as the salient objects. Then we adopt the pix2pixHD [36] technology, which assigns the pixels that have different values from the surrounding pixels as the boundary, to generate the edges of the selected objects as the corresponding edge ground truth. The quantitative results are compared with several state-of-the-art approaches in terms of pixAcc and mIoU. Our model is first trained on the SBD set without any other dataset, and then evaluated on the PASCAL VOC 2012 validation set. Following the standard protocol, we adopt to average the per-pixel classification scores at multiple scales with horizontal flipping, i.e., the scales range from 0.5 to 1.75 (in increments of 0.25) times the original size, as the multiple scale inference strategy. Note that the horizontal flipping is not included in single-scale inference. Statistics in Table 1 [4–7, 27] show that the proposed method outperforms prevalent segmentation results. In addition, though the evaluation with stride 8 can improve the performance, it consumes high-computation resource, resulting in difficulty to apply. The proposed approach facilitates the advantages of 16 stride in single-scale inference and can outperform DeepLab-v3 of multiple scales inference with stride 8, making it more applicable in reality.

Table 1 Segmentation results on the PASCAL VOC 2012 validation set[a)]

Method	OS (training)	OS (evaluating)	pixAcc (%)	mIoU (%)
DeepLab-v2 [5]	16	16	94.21	75.60
PSPNet [4]	16	16	94.62	76.82
PAN [27]	16	16	95.03	78.37
DeepLab-v3 [6]	16	16	–	77.21
DeepLab-v3[b)] [6]	16	**8**	–	79.77
DeepLab-v3+ [7]	16	16	–	78.85
DeepLab-v3+[b)] [7]	16	16	–	80.22
DeepLab-v3+[b)] [7]	16	**8**	–	80.57
CGNet (ours)	16	16	95.32	79.89
CGNet[b)] (ours)	16	16	**95.67**	**81.04**

a) OS denotes the output stride of features in training or evaluating process. The best score is marked in bold.
b) The method adopts the multiple scale inference strategy.

Table 2 Segmentation results on the PASCAL VOC 2012 test set w/o COCO pre-training[a)]

Method	aero (%)	bike (%)	bird (%)	boat (%)	bottle (%)	bus (%)	car (%)	cat (%)	chair (%)	cow (%)
FCN [2]	76.8	34.2	68.9	49.4	60.3	75.3	74.7	77.6	21.4	62.5
DeepLab-v2 [5]	84.4	54.5	81.5	63.6	65.9	85.1	79.1	83.4	30.7	74.1
CRF-RNN [52]	87.5	39.0	79.7	64.2	68.3	87.6	80.8	84.4	30.4	78.2
DeconvNet [18]	89.9	39.3	79.7	63.9	68.2	87.4	81.2	86.1	28.5	77.0
DPN [53]	87.7	59.4	78.4	64.9	70.3	89.3	83.5	86.1	31.7	79.9
Piecewise [54]	90.6	37.6	80.0	67.8	74.4	92.0	85.2	86.2	39.1	81.2
AAF [55]	91.3	72.9	90.7	68.2	77.7	95.6	90.7	94.7	40.9	89.5
ResNet38 [56]	94.4	72.9	94.9	68.8	78.4	90.6	90.0	92.1	40.1	90.4
PSPNet [4]	91.8	71.9	94.7	71.2	75.8	95.2	89.9	**95.9**	39.3	90.7
EncNet [21]	94.1	69.2	**96.3**	**76.7**	**86.2**	96.3	90.7	94.2	38.8	90.7
PAN [27]	**95.7**	**75.2**	94.0	73.8	79.6	**96.5**	**93.7**	94.1	40.5	**93.3**
CGNet (ours)	95.3	72.6	94.6	71.8	82.0	95.7	91.9	95.8	**41.8**	91.5

Method	table (%)	dog (%)	horse (%)	mbike (%)	person (%)	plant (%)	sheep (%)	sofa (%)	train (%)	tv (%)	mIoU (%)
FCN [2]	46.8	71.8	63.9	76.5	73.9	45.2	72.4	37.4	70.9	55.1	62.2
DeepLab-v2 [5]	59.8	79.0	76.1	83.2	80.8	59.7	82.2	50.4	73.1	63.7	71.6
CRF-RNN [52]	60.4	80.5	77.8	83.1	80.6	59.5	82.8	47.8	78.3	67.1	72.0
DeconvNet [18]	62.0	79.0	80.3	83.6	80.2	58.8	83.4	54.3	80.7	65.0	72.5
DPN [53]	62.6	81.9	80.0	83.5	82.3	60.5	83.2	53.4	77.9	65.0	74.1
Piecewise [54]	58.9	83.8	83.9	84.3	84.8	62.1	83.2	58.2	80.8	72.3	75.3
AAF [55]	72.6	**91.6**	94.1	88.3	88.8	67.3	92.9	62.6	85.2	74.0	82.2
ResNet38 [56]	71.7	89.9	93.7	91.0	89.1	71.3	90.7	61.3	87.7	78.1	82.5
PSPNet [4]	71.7	90.5	**94.5**	88.8	**89.6**	72.8	89.6	64.0	85.1	76.3	82.6
EncNet [21]	73.3	90.0	92.5	88.8	87.9	68.7	92.6	59.0	86.4	73.4	82.9
PAN [27]	72.4	89.1	94.1	**91.6**	89.5	**73.6**	93.2	62.8	87.3	78.6	84.0
CGNet (ours)	**74.4**	91.0	92.1	90.3	89.3	71.5	**94.1**	**67.2**	**88.6**	**81.4**	**84.2**

a) The best score is marked in bold, while the second best score is marked in underline.

Then the PASCAL VOC 2012 validation set is added to the SBD set to train our model, so as to evaluate the proposed method on PASCAL VOC 2012 test set. As shown in Table 2 [2,4,5,18,27,52–56], our proposed method outperforms all the state-of-the-art methods, which train their networks only on SBD and PASCAL VOC 2012 validation set. Notably, this also does not require any other pre-/post-processing step (i.e., over-segmentation, CRF), which is required in some methods. Furthermore, the proposed approach outperforms other methods in hard examples. For example, the chair and table are not easy to be distinguished in the PASCAL VOC 2012 dataset, but our approach enables successfully segmenting them with the benefits of utilizing edge information; the tv and sofa are not frequently

(a) (b) (c) (d) (e) (f)

Figure 4 (Color online) Examples of segmentation results on PASCAL VOC 2012 test set. We choose DeepLab-v2 as the base network. As can be seen, our proposed method can segment objects better in objects edge and small parts of objects. (a) and (d) Input images; (b) and (e) results of the network; (c) and (f) results of our method.

Table 3 Segmentation results on the PASCAL-Person-Part test set[a)]

Method	Head (%)	Torso (%)	U-Arm (%)	L-Arm (%)	U-Leg (%)	L-Leg (%)	B.G. (%)	mIoU (%)
HAZN [57]	80.79	59.11	43.05	42.76	38.99	34.46	93.59	56.11
Attention [58]	81.47	59.06	44.15	42.50	38.28	35.62	93.65	56.39
LG-LSTM [59]	82.72	60.99	45.40	47.76	42.33	37.96	88.63	57.97
Attention+SSL [60]	83.26	62.40	47.80	45.58	42.32	39.48	94.68	59.36
Attention+MMAN [61]	82.58	62.83	48.49	47.37	42.80	40.40	94.92	59.91
Graph LSTM [62]	82.69	62.68	46.88	47.71	45.66	40.93	94.59	60.16
SS-NAN [63]	86.43	67.28	51.09	48.07	44.82	42.15	97.23	62.44
Structure LSTM [64]	82.89	67.15	51.42	48.72	51.72	45.91	97.18	63.57
Joint [49]	85.50	67.87	54.72	54.30	48.25	44.76	95.32	64.39
DeepLab-v2 [5]	–	–	–	–	–	–	–	64.94
MuLA [65]	–	–	–	–	–	–	–	65.10
PCNet [66]	86.81	69.06	55.35	55.27	50.21	48.54	96.07	65.90
Holistic [67]	–	–	–	–	–	–	–	66.30
WSHP [68]	87.15	72.28	57.07	56.21	52.43	50.36	**97.72**	67.60
DeepLab-v3+ [7]	–	–	–	–	–	–	–	67.84
PGN [69]	**90.89**	**75.12**	55.83	**64.61**	**55.42**	41.57	95.33	68.40
CGNet (ours)	87.69	72.32	**63.02**	63.62	55.34	**52.99**	95.98	**70.14**

a) U-Arm and L-Arm refer to upper arm and lower arm, while U-Leg and L-Leg denote upper leg and lower leg. B.G. stands for background. The best score is marked in bold, while the second best score is marked in underline.

appeared in the dataset, however, our method can segment them better owing to the utilization of saliency information.

In addition to the quantitative results, we show the qualitative results of PASCAL VOC 2012 in Figure 4. It is observed that our method can provide segmentation results with more accurate object edges, and segment objects which are associated with small regions. By effectively exploiting the intrinsic relationship among segmentation, edge, and saliency information, our approach can output reasonable results for semantic segmentation.

4.3.2 *Experiments on PASCAL-Person-Part*

We further evaluate the proposed approach on a more challenging fine-grained semantic segmentation dataset: PASCAL-Person-Part. The salient object in an image is person, and the edge ground truth is the edge of each human body part, which is generated by the technology used in pix2pixHD [36].

The quantitative results are given in Table 3 [5, 7, 49, 57–69]. As it shows, the proposed method

Figure 5 (Color online) Examples of segmentation results on PASCAL-Person-Part test set. (a) Input images; (b) ground truth; (c) results of PSPNet; (d) results of SS-NAN; (e) results of our method. As can be seen, our proposed method can generate better results when compared with PSPNet and SS-NAN, especially in distinguishing similar objects and segmenting small objects.

also outperforms the state-of-the-art approaches in terms of mIoU, with a large margin improvement 2.3% compared with DeepLab-v3+, and 1.74% compared with PGN, which is the previous best method in PASCAL-Person-Part. For the hard examples in this dataset, i.e., arms and legs, our method can alleviate the wrongly segmenting problem, which demonstrates the effectiveness of CGNet.

Figure 5 depicts some qualitative segmentation results. Similar to the observation on PASCAL VOC 2012, the proposed method enables to generate more semantic results, demonstrating the superiority of our proposed CGNet.

4.3.3 *Experiments on Cityscapes*

To further study the generalization ability of the proposed method, we conduct experiments on the Cityscapes dataset. Different from the other two datasets, in the Cityscapes dataset, stuff needs to be removed, and the rest (i.e., pole, traffic light, traffic sign, person, rider, car, truck, bus, train, motor, bicycle) can be viewed as salient objects. We then apply the technology used in pix2pixHD [36] on these objects to yield the edge ground truth.

Based on settings above, we evaluate the proposed method on the Cityscapes test set, results are listed in Table 4 [2,4,5,20,28,44,55,70–76]. As listed in Table 4, the proposed method outperforms the PSANet

Table 4 Segmentation results on the Cityscapes test set[a)]

Method	IoU cla. (%)	iIoU cla. (%)	IoU cat. (%)	iIoU cat. (%)
FCN [2]	65.3	41.7	85.7	70.1
DeepLab-v2 [5]	70.4	42.6	86.4	67.7
RefineNet [20]	73.6	–	–	–
DSSPN [70]	76.6	56.2	89.6	77.8
GCN [28]	76.9	–	–	–
DUC [71]	77.6	53.6	90.1	75.2
SAC [72]	78.1	55.2	90.6	78.3
PSPNet [4]	78.4	56.7	90.6	78.6
BiSeNet [73]	78.9	–	–	–
AAF [55]	79.1	56.1	90.8	78.5
DFN [74]	79.3	–	–	–
PSANet [75]	80.1	–	–	–
ANN [76]	81.3	–	–	–
DANet [44]	**81.5**	–	–	–
CGNet (ours)	81.3	**62.5**	**91.4**	**79.7**

a) The best score is marked in bold.

Table 5 Ablation study on the PASCAL-Person-Part test set[a)]

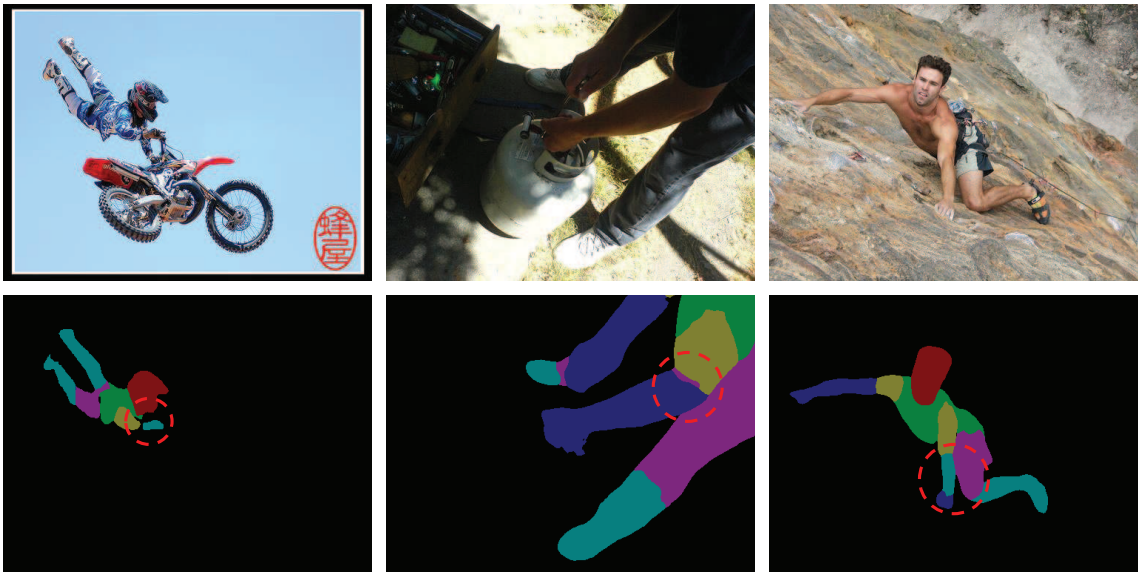
Method	pixAcc (%)	mIoU (%)
DeepLab-v2 [5]	93.55	64.94
DeepLab-v3+ [7]	94.23	67.84
Base	93.02	62.62
Base + Pyramid Attention	94.02	66.95
Base + Pyramid Attention + Edge	94.21	67.78
Base + Pyramid Attention + Salient	94.17	67.63
Base + Pyramid Attention + Edge + Salient	94.33	68.17
Base + Pyramid Attention + Concat (edge & salient)	94.44	68.46
Base + Pyramid + CGM	**94.78**	**70.14**

a) Base denotes segmentation only using the backbone network. The best score is marked in bold.

with a large margin, from 80.1% to 81.3%. When compared with DANet, the proposed method achieves 81.3% while DANet achieves 81.5%. The main reason the DANet achieves higher performance than ours is that the DANet adopts stride 8 for training and testing, while the proposed method adopts stride 16 for training and testing owing to the limitation of computational resources. In general, the larger stride is better for real applications, as a result, with a small gap, the performance of the proposed method is acceptable. The proposed method achieves the same performance with ANN, which proposes a novel way based on the self-attention mechanism to enhance the performance, indicating the effectiveness of the proposed method.

4.4 Ablation study

To evaluate the contributions of each component of the proposed method, we conduct ablation experiments with different settings on the PASCAL-Person-Part dataset. As shown in Table 5 [5, 7], we adopt the ResNet-101 as the base network, which achieves 62.62% in terms of mIoU. When the pyramid attentive module, there is a 4.33% mIoU improvements. In addition, when the edge or saliency detection head is integrated into the unified network, it dramatically outperforms the previous network, with only a small increasing on the computational cost, proving that edge and saliency information is of vital importance for semantic segmentation. However, when we append both of them, there is only a slight improvement when compared with using the proposed CGM, which is about 2% higher than the multi-task method, demonstrating the importance of modeling the intrinsic information among them. In addition, if we di-



(a) (b) (c)

Figure 6 (Color online) Illustration of the failure cases. The first row shows the input images, the second row shows the predictions of the proposed method. (a) and (c) show that the proposed method fails to predict the lower-arm into lower-leg. (b) shows that the proposed fails to segment the lower-arm into upper-leg.

rectly concatenate the edge and salient features with the output of the PAM, it yields 68.46% mIoU, which is lower than the 70.14% mIoU of the proposed method. The 1.68% higher performance demonstrates the superiority of the proposed CGM.

4.5 Failure cases

Despite the excellent results, there are still a small number of failure cases produced by the proposed method. Figures 6(a) and (c) show that the proposed method mistakenly predicts the lower-arm as lower-leg, while Figure 6(b) shows that the method mistakenly segments the lower-arm as upper-leg. These bad cases are often caused by the complex human topology, and these complex instances are very rare in the training set, making the model difficult to recognize them. For example, in Figure 6(b), the upper-arm is close to the upper-leg. This phenomenon is very similar to the closed upper-legs in normal human topology, as a result, the proposed method has difficulty in predicting the upper-arm. To solve this problem, more training images that contain more complex human topology instances are needed, or extra human body configuration such as human key points should be considered.

5 Conclusion

To model the intrinsic correlation among segmentation, edge, and saliency information, so as to guide the extraction of discriminative context features, we proposed the CGNet, which can yield more semantic results, without any pre-/post- processing. As a consequence, the proposed method achieves significant performance gain both in accuracy and efficiency, making it easy to be widely applied. Our future work will consider transferring this method to more sensitive tasks, such as 3D point clouds segmentation, object detection, and key points detection.

Acknowledgements This work was supported in part by the Science and Technology Innovation 2030-Major Project of Artificial Intelligence of the Ministry of Science and Technology of China (Grant No. 2018AAA01028) and in part by National Natural Science Foundation of China (Grant No. 61632018).

References

1 Geng Q C, Zhou Z, Cao X C. Survey of recent progress in semantic image segmentation with CNNs. Sci China Inf Sci, 2018, 61: 051101
2 Shelhamer E, Long J, Darrell T. Fully convolutional networks for semantic segmentation. IEEE Trans Pattern Anal Mach Intell, 2017, 39: 640–651
3 He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell, 2015, 37: 1904–1916
4 Zhao H, Shi J, Qi X, et al. Pyramid scene parsing network. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 6230–6239
5 Chen L-C, Papandreou G, Kokkinos I, et al. DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Trans Pattern Anal Mach Intell, 2018, 40: 834–848
6 Chen L-C, Papandreou G, Schroff F, et al. Rethinking atrous convolution for semantic image segmentation. 2017. ArXiv: 1706.05587
7 Chen L-C, Zhu Y, Papandreou G, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of European Conference on Computer Vision, Munich, 2018. 833–851
8 Joachims T, Finley T, Yu C-N J. Cutting-plane training of structural SVMs. Mach Learn, 2009, 77: 27–59
9 Lin T-Y, Goyal P, Girshick R, et al. Focal loss for dense object detection. In: Proceedings of IEEE International Conference on Computer Vision, Venice, 2017. 2999–3007
10 Wu Z, Shen C, Hengel A. High-performance semantic segmentation using very deep fully convolutional networks. 2016. ArXiv: 1604.04339
11 Kokkinos I. UberNet: training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 5454–5463
12 Sun H Q, Pang Y W. GlanceNets efficient convolutional neural networks with adaptive hard example mining. Sci China Inf Sci, 2018, 61: 109101
13 Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: Proceedings of International Conference on Learning Representations, San Diego, 2015
14 He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 2016. 770–778
15 Huang G, Liu Z, Maaten L, et al. Densely connected convolutional networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 2261–2269
16 Chollet F. Xception: deep learning with depthwise separable convolutions. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 1800–1807
17 Badrinarayanan V, Kendall A, Cipolla R. SegNet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell, 2017, 39: 2481–2495
18 Noh H, Hong S, Han B. Learning deconvolution network for semantic segmentation. In: Proceedings of IEEE International Conference on Computer Vision, Santiago, 2015. 1520–1528
19 Yu F, Koltun V, Funkhouser T A. Dilated residual networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 636–644
20 Lin G, Milan A, Shen C, et al. RefineNet: multi-path refinement networks for high-resolution semantic segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 5168–5177
21 Zhang H, Dana K, Shi J, et al. Context encoding for semantic segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 7151–7160
22 Huang Z, Wang X, Huang L, et al. CCNet: criss-cross attention for semantic segmentation. In: Proceedings of IEEE International Conference on Computer Vision, Seoul, 2019
23 Jégou S, Drozdzal M, Vázquez D, et al. The one hundred layers tiramisu: fully convolutional densenets for semantic segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, 2017. 1175–1183
24 Yang M, Yu K, Zhang C, et al. DenseASPP for semantic segmentation in street scenes. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 3684–3692
25 Zhang Z, Zhang X, Peng C, et al. ExFuse: enhancing feature fusion for semantic segmentation. In: Proceedings of European Conference on Computer Vision, Munich, 2018. 273–288
26 Zhao H, Qi X, Shen X, et al. ICNet for real-time semantic segmentation on high-resolution images. In: Proceedings of European Conference on Computer Vision, Munich, 2018. 418–434
27 Li H, Xiong P, An J, et al. Pyramid attention network for semantic segmentation. In: Proceedings of British Machine Vision Conference, Newcastle, 2018. 285
28 Peng C, Zhang X, Yu G, et al. Large kernel matters—improve semantic segmentation by global convolutional network. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 1743–1751
29 Wei Z, Sun Y, Wang J. Learning adaptive receptive fields for deep image parsing network. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 3947–3955
30 Pang Y, Wang T, Anwer R M, et al. Efficient featurized image pyramid network for single shot detector. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, 2019. 7336–7344
31 Deng R, Shen C, Liu S, et al. Learning to predict crisp boundaries. In: Proceedings of European Conference on Computer Vision, Munich, 2018. 570–586

32 Xie S, Tu Z. Holistically-nested edge detection. Int J Comput Vis, 2017, 125: 3–18
33 Liu Y, Cheng M-M, Hu X, et al. Richer convolutional features for edge detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 5872–5881
34 Liu Y, Lew M S. Learning relaxed deep supervision for better edge detection. In: Proceedings of IEEE Conference on Computer Vision, Las Vegas, 2016. 231–240
35 Shen W, Wang X, Wang Y, et al. DeepContour: a deep convolutional feature learned by positive-sharing loss for contour detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Boston, 2015. 3982–3991
36 Wang T-C, Liu M-Y, Zhu J-Y, et al. High-resolution image synthesis and semantic manipulation with conditional GANs. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 8798–8807
37 Wang W, Lai Q, Fu H, et al. Salient object detection in the deep learning era: an in-depth survey. 2019. ArXiv: 1904.09146
38 Liu N, Han J. DHSNet: deep hierarchical saliency network for salient object detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 2016. 678–686
39 Wang W, Shen J, Dong X, et al. Salient object detection driven by fixation prediction. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 1711–1720
40 Wang W, Shen J, Yang R, et al. Saliency-aware video object segmentation. IEEE Trans Pattern Anal Mach Intell, 2018, 40: 20–33
41 Wang W, Shen J, Dong X, et al. Inferring salient objects from human fixations. IEEE Trans Pattern Anal Mach Intell, 2019. doi: 10.1109/TPAMI.2019.2905607
42 Liu N, Han J, Yang M-H. PiCANet: learning pixel-wise contextual attention for saliency detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 3089–3098
43 Hu J, Shen L, Sun G. Squeeze-and-excitation networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 7132–7141
44 Fu J, Liu J, Tian H, et al. Dual attention network for scene segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, 2019. 3146–3154
45 Wang X, Girshick R, Gupta A, et al. Non-local neural networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 7794–7803
46 Zhang X, Wang T, Qi J, et al. Progressive attention guided recurrent network for salient object detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 714–722
47 Zhang X, Xiong H, Zhou W, et al. Picking deep filter responses for fine-grained image recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 2016. 1134–1142
48 Everingham M, van Gool L, Williams C K I, et al. The pascal visual object classes (VOC) challenge. Int J Comput Vis, 2010, 88: 303–338
49 Xia F, Wang P, Chen X, et al. Joint multi-person pose estimation and semantic part segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 6080–6089
50 Cordts M, Omran M, Ramos S, et al. The cityscapes dataset for semantic urban scene understanding. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 2016. 3213–3223
51 Hariharan B, Arbelaez P, Bourdev L D, et al. Semantic contours from inverse detectors. In: Proceedings of the IEEE International Conference on Computer Vision, Barcelona, 2017. 991–998
52 Zheng S, Jayasumana S, Romera-Paredes B. Conditional random fields as recurrent neural networks. In: Proceedings of International Conference on Computer Vision, Santiago, 2015. 1529–1537
53 Liu Z, Li X, Luo P, et al. Semantic image segmentation via deep parsing network. In: Proceedings of International Conference on Computer Vision, Santiago, 2015. 1377–1385
54 Lin G, Shen C, Hengel A, et al. Efficient piecewise training of deep structured models for semantic segmentation. In: Proceedings of Conference on Computer Vision and Pattern Recognition, Las Vegas, 2016. 3194–3203
55 Ke T-W, Hwang J-J, Liu Z, et al. Adaptive affinity fields for semantic segmentation. In: Proceedings of European Conference on Computer Vision, Munich, 2018. 605–621
56 Wu Z, Shen C, van den Hengel A. Wider or deeper: revisiting the ResNet model for visual recognition. Pattern Recogn, 2019, 90: 119–133
57 Xia F, Wang P, Chen L-C, et al. Zoom better to see clearer: human and object parsing with hierarchical auto-zoom net. In: Proceedings of European Conference on Computer Vision, Amsterdam, 2016. 648–663
58 Chen L-C, Yang Y, Wang J, et al. Attention to scale: scale-aware semantic image segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 2016. 3640–3649
59 Liang X, Shen X, Xiang D, et al. Semantic object parsing with local-global long short-term memory. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, 2016. 3185–3193
60 Gong K, Liang X, Zhang D, et al. Look into person: self-supervised structure-sensitive learning and a new benchmark for human parsing. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 6757–6765
61 Luo Y, Zheng Z, Zheng L, et al. Macro-micro adversarial network for human parsing. In: Proceedings of European Conference on Computer Vision, Munich, 2018. 424–440
62 Liang X, Shen X, Feng J, et al. Semantic object parsing with graph LSTM. In: Proceedings of European Conference on Computer Vision, Amsterdam, 2016. 125–143

63 Zhao J, Li J, Nie X, et al. Self-supervised neural aggregation networks for human parsing. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, 2017. 1595–1603
64 Liang X, Lin L, Shen X, et al. Interpretable structure-evolving LSTM. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 2017. 2175–2184
65 Nie X, Feng J, Yan S. Mutual learning to adapt for joint human parsing and pose estimation. In: Proceedings of European Conference on Computer Vision, Munich, 2018. 519–534
66 Zhu B, Chen Y, Tang M, et al. Progressive cognitive human parsing. In: Proceedings of AAAI Conference on Artificial Intelligence, New Orleans, 2018. 7607–7614
67 Li Q Z, Arnab A, Torr P H S. Holistic, instance-level human parsing. In: Proceedings of British Machine Vision Conference, London, 2017
68 Fang H, Lu G, Fang X, et al. Weakly and semi supervised human body part parsing via pose-guided knowledge transfer. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 70–78
69 Gong K, Liang X, Li Y, et al. Instance-level human parsing via part grouping network. In: Proceedings of European Conference on Computer Vision, Munich, 2018. 805–822
70 Liang X, Zhou H, Xing E. Dynamic-structure semantic propagation network. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 752–761
71 Wang P, Chen P, Yuan Y, et al. Understanding convolution for semantic segmentation. In: Proceedings of IEEE Winter Conference on Applications of Computer Vision, Lake Tahoe, 2018. 1451–1460
72 Zhang R, Tang S, Zhang Y, et al. Scale-adaptive convolutions for scene parsing. In: Proceedings of IEEE International Conference on Computer Vision, Venice, 2017. 2050–2058
73 Yu C, Wang J, Peng C, et al. BiSeNet: bilateral segmentation network for real-time semantic segmentation. In: Proceedings of European Conference on Computer Vision, Munich, 2018. 334–349
74 Yu C, Wang J, Peng C, et al. Learning a discriminative feature network for semantic segmentation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 2018. 1857–1866
75 Zhao H, Zhang Y, Liu S, et al. PSANet: point-wise spatial attention network for scene parsing. In: Proceedings of European Conference on Computer Vision, Munich, 2018. 270–286
76 Zhu Z, Xu M, Bai S, et al. Asymmetric non-local neural networks for semantic segmentation. In: Proceedings of IEEE International Conference on Computer Vision, Seoul, 2019. 593–602