This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

CCNet: Criss-Cross Attention for Semantic Segmentation

Zilong Huang[1]*, Xinggang Wang[1]†, Lichao Huang[2], Chang Huang[2], Yunchao Wei[3,4], Wenyu Liu[1]
[1]School of EIC, Huazhong University of Science and Technology
[2]Horizon Robotics [3]ReLER, UTS
[4]Beckman Institute, University of Illinois at Urbana-Champaign

Abstract

Full-image dependencies provide useful contextual information to benefit visual understanding problems. In this work, we propose a Criss-Cross Network (CCNet) for obtaining such contextual information in a more effective and efficient way. Concretely, for each pixel, a novel criss-cross attention module in CCNet harvests the contextual information of all the pixels on its criss-cross path. By taking a further recurrent operation, each pixel can finally capture the full-image dependencies from all pixels. Overall, CCNet is with the following merits: 1) GPU memory friendly. Compared with the non-local block, the proposed recurrent criss-cross attention module requires 11× less GPU memory usage. 2) High computational efficiency. The recurrent criss-cross attention significantly reduces FLOPs by about 85% of the non-local block in computing full-image dependencies. 3) The state-of-the-art performance. We conduct extensive experiments on popular semantic segmentation benchmarks including Cityscapes, ADE20K, and instance segmentation benchmark COCO. In particular, our CCNet achieves the mIoU score of 81.4 and 45.22 on Cityscapes test set and ADE20K validation set, respectively, which are the new state-of-the-art results. The source code is available at https://github.com/speedinghzl/CCNet.

1. Introduction

Semantic segmentation, which is a fundamental problem in the computer vision community, aims at assigning semantic class labels to each pixel in the given image. It has been extensively and actively studied in many recent works and is also critical for various challenging and meaningful applications such as autonomous driving [14], augmented reality [1], and image editing [13]. Specifically, current state-of-the-art semantic segmentation approaches based on the fully convolutional network (FCN) [26] have made remarkable progress. However, due to the fixed geometric structures, they are inherently limited to local receptive fields and short-range contextual information. These limitations impose a great adverse effect on FCN-based methods due to insufficient contextual information.

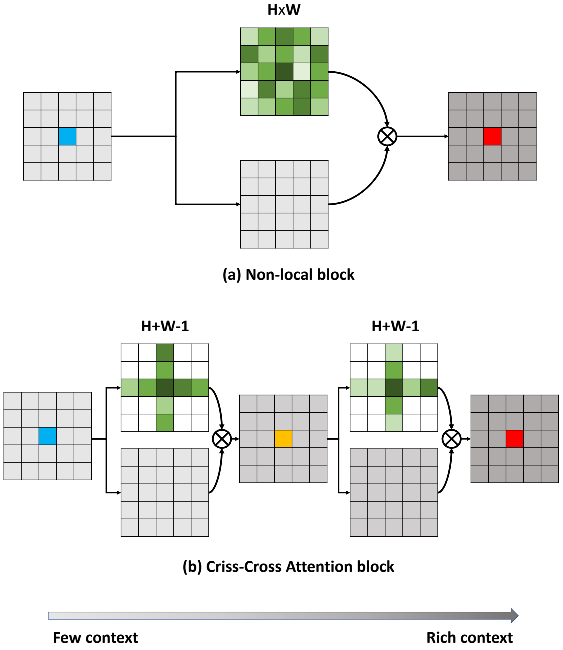

Figure 1. Diagrams of two attention-based context aggregation methods. (a) For each position (*e.g*. blue), the Non-local module [31] generates a dense attention map which has $H \times W$ weights (in green). (b) For each position (*e.g*. blue), the criss-cross attention module generates a sparse attention map which only has $H + W - 1$ weights. After the recurrent operation, each position (*e.g*. red) in the final output feature maps can collect information from all pixels. For clear display, residual connections are ignored.

To make up for the above deficiency of FCN, some works have been proposed to introduce useful contextual information to benefit the semantic segmentation task. Specifically, Chen *et al*. [5] proposed atrous spatial pyramid pool-

*The work was mainly done during an internship at Horizon Robotics
†Corresponding author.

ing module with multi-scale dilation convolutions for contextual information aggregation. Zhao *et al*. [41] further introduced PSPNet with pyramid pooling module to capture contextual information. However, the dilated convolution based methods [6, 5, 12] collect information from a few surrounding pixels and can not generate dense contextual information actually. Meanwhile, the pooling based methods [41, 39] aggregate contextual information in a non-adaptive manner and the homogeneous contextual information is adopted by all image pixels, which does not satisfy the requirement that different pixels need different contextual dependencies.

To generate dense and pixel-wise contextual information, PSANet [42] learns to aggregate contextual information for each position via a predicted attention map. Non-local Networks [31] utilizes a self-attention mechanism [9, 29], which enables a single feature from any position to perceive features of all the other positions, thus harvesting full-image contextual information, see Fig. 1 (a). However, these attention-based methods need to generate huge attention maps to measure the relationships for each pixel-pair, whose complexity in time and space are both $\mathcal{O}((H\times W)\times(H\times W))$, where $H \times W$ donates the spatial dimension of input feature maps. Since the input feature maps are always with high resolution in semantic segmentation task, self-attention based methods have high computation complexity and occupy a huge number of GPU memory. Thus, is there an alternative solution to achieve such a target in a more efficient way?

To address the above mentioned problem, our motivation is to consecutive sparse attention to replace the single layer dense attention in the non-local networks. Without loss of generality, we use two consecutive criss-cross attention modules, in which each one only has sparse connections $(H+W-1)$ for each position in the feature maps. The criss-cross attention module aggregates contextual information in horizontal and vertical directions. By serially stacking two criss-cross attention modules, it can collect contextual information from all pixels. The above decomposition strategy greatly reduces the complexity in time and space from $\mathcal{O}((H\times W)\times(H\times W))$ to $\mathcal{O}((H\times W)\times(H+W-1))$.

We compare the differences between the non-local module [31] and our criss-cross attention module in Fig. 1. Concretely, both non-local module and criss-cross attention module feed the input feature maps with spatial size $H\times W$ to generate attention maps (upper branch) and adapted feature maps (lower branch), respectively. Then, the weighted sum is adopted to collecting contextual information. Different from the dense connections adopted by the non-local module, each position (*e.g*., blue color) in the feature maps is sparsely connected with other ones which are in the same row and the same column in our criss-cross attention module, leading to the predicted attention map only has H+W–1 weights rather than $H\times W$ in non-local module. To achieve the target of capturing the full-image dependencies, we then innovatively and simply take a recurrent operation for the criss-cross attention module. In particular, the local features are firstly passed through one criss-cross attention module to collect the contextual information in horizontal and vertical directions. Then, by feeding the produced feature maps from the first criss-cross attention module to the other one, the additional contextual information obtained from the criss-cross path is finally enable the full-image dependencies to be captured by each pixel. As demonstrated in Fig. 1 (b), each position (*e.g*. red color) in the second feature maps finally collects information from all others to augment the pixel-wise representations. We share parameters of the recurrent criss-cross module to reduce extra parameters. Our criss-cross attention module can be easily plugged into any fully convolutional neural network, named CCNet, for leaning to segment in an end-to-end manner.

We have carried out extensive experiments on multiple large-scale datasets. Our proposed CCNet achieves top performance on two most competitive semantic segmentation datasets, *i.e*., Cityscapes [10] and ADE20K [44]. In addition, the proposed criss-cross attention even improves the state-of-the-art instance segmentation method, *i.e*., Mask R-CNN with ResNet-101 [17]. These results well demonstrate that our criss-cross attention module is generally beneficial to the dense prediction tasks. In summary, our main contributions are two-fold:

- We propose a novel criss-cross attention module in this work, which can be leveraged to capture contextual information from full-image dependencies in a more efficient and effective way.

- We propose CCNet by taking advantages of recurrent criss-cross attention module, achieving leading performance on segmentation-based benchmarks, including Cityscapes, ADE20K and COCO.

2. Related work

Semantic segmentation The last years have seen a renewal of interest on semantic segmentation. FCN [26] is the first approach to adopt fully convolutional network for semantic segmentation. Later, FCN-based methods have made great progress in image semantic segmentation. Chen *et al*. [4] and Yu *et al*. [37] removed the last two downsample layers to obtain dense prediction and utilized dilated convolutions to enlarge the receptive field. Unet [28], Deeplabv3+ [8], MSCI [21], SPGNet [2], RefineNet [22] and DFN [36] adopted encoder-decoder structures that fuse the information in low-level and high-level layers to predict segmentation mask. SAC [40] and Deformable Convolutional Networks [11] improved the standard convolutional operator

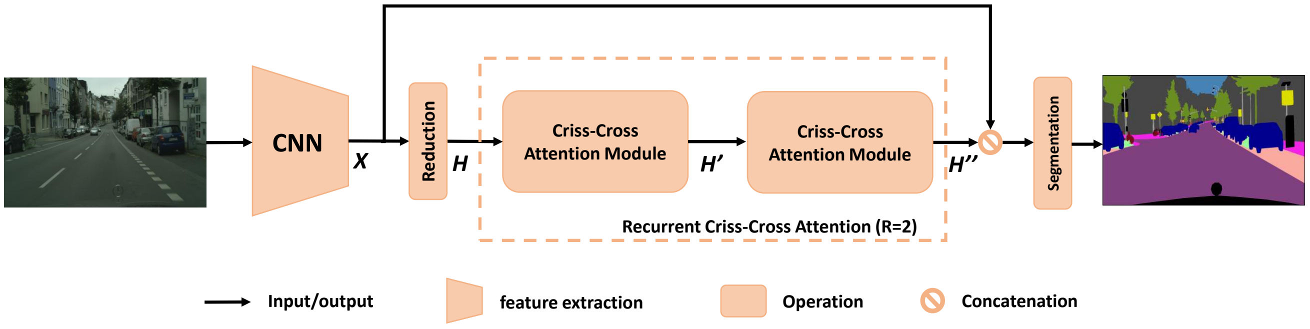
CNN
X
Reduction
H
Criss-Cross Attention Module
H'
Criss-Cross Attention Module
H''
Segmentation
Recurrent Criss-Cross Attention (R=2)
Input/output
feature extraction
Operation
Concatenation

Figure 2. Overview of the proposed CCNet for semantic segmentation.

to handle the deformation and various scale of objects. CRF-RNN [37] and DPN [25] used Graph model, *i.e*. CRF, MRF, for semantic segmentation. AAF [19] used adversarial learning to capture and match the semantic relations between neighboring pixels in the label space. BiSeNet [35] was designed for real-time semantic segmentation.

Contextual information aggregation In addition, some works aggregate the contextual information to augment the feature representation. Deeplabv2 [5] proposed ASPP module to use different dilation convolutions to capture contextual information. DenseASPP [34] brought dense connections into ASPP to generate features with various scale. DPC [3] utilized architecture search techniques to build multi-scale architectures for semantic segmentation. PSPNet [41] utilized pyramid pooling to aggregate contextual information. GCN [27] utilized global convolutional module utilized global pooling to harvest context information for global representations. Recently, Zhao *et al*. [42] proposed the point-wise spatial attention network which uses predicted attention map to guide contextual information collection. Liu *et al*. [24] utilized RNNs to capture long-range dependencies. Conditional random field (CRF) [4, 43], Markov random field (MRF) [25] are also utilized to capture long-range dependencies for semantic segmentation.

Attention model Attention model is widely used for various tasks. Squeeze-and-Excitation Networks [18] enhanced the representational power of the network by modeling channel-wise relationships in an attention mechanism. Chen *et al*. [7] made use of several attention masks to fuse feature maps or predictions from different branches. Vaswani *et al*. [29] applied a self-attention model on machine translation. Wang *et al*. [31] proposed the non-local module to generate the huge attention map by calculating the correlation matrix between each spatial point in the feature maps, then the attention guided dense contextual information aggregation. OCNet [38] and DANet [15] utilized Non-local module [31] to harvest the contextual information. PSA [42] learned an attention map to aggregate contextual information for each individual point adaptively and specifically.

CCNet vs. Non-Local vs. GCN Here, we specifically discusses the differences among GCN [27], Non-local Network [31] and CCNet. In term of contextual information aggregation, only the center point can perceive the contextual information from all pixels in GCN [27]. In contrast, Non-local Network [31] and CCNet guarantee that a pixel at any position perceives contextual information from all pixels. Although, GCN [27] alternatively decomposes the square-shape convolutional operation to horizontal and vertical linear convolutional operations which is related to CCNet, CCNet takes criss-cross way to harvest contextual information which is more effective than horizontal-vertical separate way. Moreover, CCNet is proposed to mimic Non-local Network [31] for obtaining dense contextual information through a more effective and efficient recurrent criss-cross attention module, in which dissimilar features get low attention weights and features with high attention weights are similar ones.

3. Approach

In this section, we give the details of the proposed Criss-Cross Network (CCNet) for semantic segmentation. We first present a general framework of our CCNet. Then, the criss-cross attention module which captures contextual information in horizontal and vertical directions will be introduced. Finally, to capture the dense and global contextual information, we propose to adopt a recurrent operation for the criss-cross attention module.

3.1. Network Architecture

The network architecture is given in Fig. 2. An input image is passed through a deep convolutional neural network (DCNN), which is designed in a fully convolutional fashion [5], to produce feature maps $\mathbf{X}$ with the spatial size of $H \times W$. In order to retain more details and efficiently produce dense feature maps, we remove the last two down-sampling operations and employ dilation convolutions in

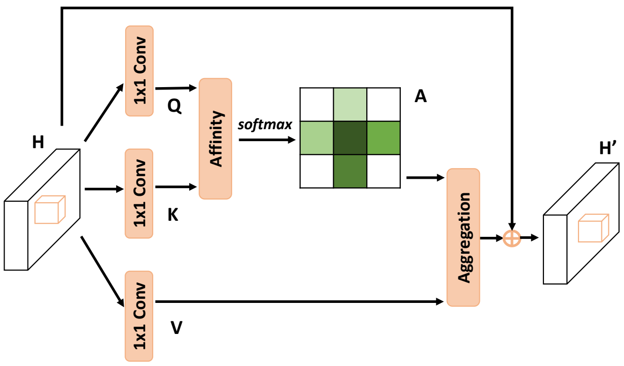

Figure 3. The details of criss-cross attention module.

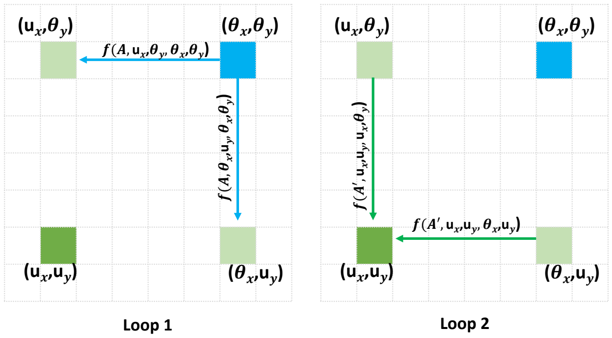
(u_x,θ_y)
(θ_x,θ_y)
$f(A, u_x,\theta_y, \theta_x,\theta_y)$
$f(A, \theta_x, u_y, \theta_x,\theta_y)$
(u_x,u_y)
(θ_x,u_y)
Loop 1
(u_x,θ_y)
(θ_x,θ_y)
$f(A', u_x, u_y, u_x,\theta_y)$
$f(A', u_x,u_y, \theta_x,u_y)$
(u_x,u_y)
(θ_x,u_y)
Loop 2

Figure 4. An example of information propagation when the loop number is 2.

the subsequent convolutional layers, leading to enlarge the width/height of the output feature maps $\mathbf{X}$ to 1/8 of the input image.

Given the $\mathbf{X}$, we first apply a convolutional layer to obtain the feature maps $\mathbf{H}$ of dimension reduction, then, the feature maps $\mathbf{H}$ are fed into the criss-cross attention module to generate new feature maps $\mathbf{H}'$ which aggregate contextual information together for each pixel in its criss-cross path. The feature maps $\mathbf{H}'$ only aggregate the contextual information in horizontal and vertical directions which are not powerful enough for semantic segmentation. To obtain richer and denser context information, we feed the feature maps $\mathbf{H}'$ into the criss-cross attention module again and output feature maps $\mathbf{H}''$. Thus, each position in feature maps $\mathbf{H}''$ actually gathers the information from all pixels. Two criss-cross attention modules before and after share the same parameters to avoid adding too many extra parameters. We name this recurrent structure as recurrent criss-cross attention (RCCA) module.

Then, we concatenate the dense contextual feature $\mathbf{H}''$ with the local representation feature $\mathbf{X}$. It is followed by one or several convolutional layers with batch normalization and activation for feature fusion. Finally, the fused features are fed into the segmentation layer to predict the final segmentation result.

3.2. Criss-Cross Attention

To model full-image dependencies over local feature representations using lightweight computation and memory, we introduce a criss-cross attention module. The criss-cross attention module collects contextual information in horizontal and vertical directions to enhance pixel-wise representative capability. As shown in Fig. 3, given a local feature maps $\mathbf{H} \in \mathbb{R}^{C\times W\times H}$, the module firstly applies two convolutional layers with 1×1 filters on $\mathbf{H}$ to generate two feature maps $\mathbf{Q}$ and $\mathbf{K}$, respectively, where $\{\mathbf{Q}, \mathbf{K}\} \in \mathbb{R}^{C'\times W\times H}$. C' is the number of channel, which is less than C for dimension reduction.

After obtaining feature maps $\mathbf{Q}$ and $\mathbf{K}$, we further generate attention maps $\mathbf{A} \in \mathbb{R}^{(H+W-1)\times W\times H}$ via **Affinity** operation. At each position u in the spatial dimension of feature maps $\mathbf{Q}$, we can obtain a vector $\mathbf{Q_u} \in \mathbb{R}^{C'}$. Meanwhile, we can also obtain the set $\mathbf{\Omega_u} \in \mathbb{R}^{(H+W-1)\times C'}$ by extracting feature vectors from $\mathbf{K}$ which are in the same row or column with position u. $\mathbf{\Omega_{i,u}} \in \mathbb{R}^{C'}$ is the ith element of $\mathbf{\Omega_u}$. The **Affinity** operation is then defined as follows:

$$d_{i,u} = \mathbf{Q_u}\mathbf{\Omega_{i,u}}^{\mathsf{T}} \tag{1}$$

where $d_{i,u} \in \mathbf{D}$ is the degree of correlation between feature $\mathbf{Q_u}$ and $\mathbf{\Omega_{i,u}}$, $i = [1, ..., |\mathbf{\Omega_u}|]$, $\mathbf{D} \in \mathbb{R}^{(H+W-1)\times W\times H}$. Then, we apply a softmax layer on $\mathbf{D}$ over the channel dimension to calculate the attention map $\mathbf{A}$.

Another convolutional layer with 1×1 filters is applied on $\mathbf{H}$ to generate $\mathbf{V} \in \mathbb{R}^{C\times W\times H}$ for feature adaption. At each position u in the spatial dimension of feature maps $\mathbf{V}$, we can obtain a vector $\mathbf{V_u} \in \mathbb{R}^{C}$ and a set $\mathbf{\Phi_u} \in \mathbb{R}^{(H+W-1)\times C}$. The set $\mathbf{\Phi_u}$ is a collection of feature vectors in $\mathbf{V}$ which are in the same row or column with position u. The contextual information is collected by the **Aggregation** operation:

$$\mathbf{H}'_\mathbf{u} = \sum_{i\in|\mathbf{\Phi_u}|} \mathbf{A_{i,u}}\mathbf{\Phi_{i,u}} + \mathbf{H_u} \tag{2}$$

where $\mathbf{H}'_\mathbf{u}$ is a feature vector in output feature maps $\mathbf{H}' \in \mathbb{R}^{C\times W\times H}$ at position u. $\mathbf{A_{i,u}}$ is a scalar value at channel i and position u in $\mathbf{A}$. The contextual information is added to local feature $\mathbf{H}$ to enhance the local features and augment the pixel-wise representation. Therefore, it has a wide contextual view and selectively aggregates contexts according to the spatial attention map. These feature representations achieve mutual gains and are more robust for semantic segmentation.

3.3. Recurrent Criss-Cross Attention (RCCA)

Despite a criss-cross attention can capture contextual information in horizontal and vertical directions, the connections between one pixel and its around ones that are not in the criss-cross path are still absent. To tackle this problem, we innovatively and simply introduce a RCCA operation based on the criss-cross attention. The RCCA module can be unrolled into R loops. In the first loop, the criss-cross attention takes the feature maps $\mathbf{H}$ extracted from a CNN model as the input and output the feature maps $\mathbf{H}'$, where $\mathbf{H}$ and $\mathbf{H}'$ are with the same shape. In the second loop, the criss-cross attention takes the feature maps $\mathbf{H}'$ as the input and output the feature maps $\mathbf{H}''$. As shown in Fig. 2, the RCCA module is equipped with two loops (R=2) which is able to harvest full-image contextual information from all pixels to generate new feature maps with dense and rich contextual information.

We denote $\mathbf{A}$ and $\mathbf{A}'$ as the attention maps in loop 1 and loop 2, respectively. Since we are interested only in contextual information spreads in spatial dimension rather than in channel dimension, the convolutional layer with 1×1 filters can be view as the identical connection. In addition, the mapping function from position x', y' to weight $A_{i,x,y}$ is defined as $A_{i,x,y} = f(A, x, y, x', y')$. For any position u at the feature maps $\mathbf{H}''$ and any position θ at the feature maps $\mathbf{H}$, there is actually a connection in the case of $R = 2$. For the case that u and θ are in the same row or column:

$$\mathbf{H}''_{\mathbf{u}} \leftarrow [f(A, u, \theta) + 1] \cdot f(A', u, \theta) \cdot \mathbf{H}_\theta \qquad (3)$$

where $\leftarrow$ donates the add-to operation. For the other case that u and θ are not in the same row and column, Fig. 4 shows the propagation path of context information in spatial dimension:

$$\begin{aligned}\mathbf{H}''_{\mathbf{u}} \leftarrow [&f(A, u_x, \theta_y, \theta_x, \theta_y) \cdot f(A', u_x, u_y, u_x, \theta_y) + \\ &f(A, \theta_x, u_y, \theta_x, \theta_y) \cdot f(A', u_x, u_y, \theta_x, u_y)] \cdot \mathbf{H}_\theta \end{aligned} \qquad (4)$$

In general, our RCCA module makes up for the deficiency of criss-cross attention that cannot obtain the dense contextual information from all pixels. Compared with criss-cross attention, the RCCA module ($R = 2$) does not bring extra parameters and can achieve better performance with the cost of a minor computation increment.

4. Experiments

To evaluate the effectiveness of the CCNet, we carry out comprehensive experiments on the Cityscapes dataset [10], the ADE20K dataset [44], and the COCO dataset [23]. Experimental results demonstrate that CCNet achieves state-of-the-art performance on Cityscapes and ADE20K. Meanwhile, CCNet can bring constant performance gain on COCO for instance segmentation. In the following subsections, we first introduce the datasets and implementation details, then we perform a series of ablation experiments on Cityscapes dataset. Finally, we report our results on ADE20K and COCO dataset.

4.1. Datasets and Evaluation Metrics

We adopt Mean IoU (mIOU, mean of class-wise intersection over union) for Cityscapes and ADE20K and the standard COCO metrics Average Precision (AP) for COCO.

- **Cityscapes** is tasked for urban segmentation, Only the 5,000 finely annotated images are used in our experiments and are divided into 2,975/500/1,525 images for training, validation, and testing.
- **ADE20K** is a recent scene parsing benchmark containing dense labels of 150 stuff/object categories. The dataset includes 20K/2K/3K images for training, validation and test.
- **COCO** is a very challenging dataset for instance segmentation that contains 115K images over 80 categories for training, 5K images for validation and 20k images for testing.

4.2. Implementation Details

Network Structure For semantic segmentation, we choose the ImageNet pre-trained ResNet-101 [17] as our backbone and remove the last two down-sampling operations and employ dilated convolutions in the subsequent convolutional layers following the previous work [4], resulting in the output stride as 8. For instance segmentation, we choose Mask-RCNN [16] as our baseline.

Training settings SGD with mini-batch is used for training. For semantic segmentation, the initial learning rate is 1e-2 for Cityscapes and ADE20K. Following the prior works [5, 39], we employ a poly learning rate policy where the initial learning rate is multiplied by $1 - (\frac{iter}{max\_iter})^{power}$ with $power$ = 0.9. We use the momentum of 0.9 and a weight decay of 0.0001. For Cityscapes, the training images are augmented by randomly scaling (from 0.75 to 2.0), then randomly cropping out the high-resolution patches (769×769) from the resulting images. Since the images from ADE20K are with various sizes, we adopt an augmentation strategy of resizing the short side of input image to the length randomly chosen from the set {300, 375, 450, 525, 600}. For instance segmentation, we take the same training settings as that of Mask-RCNN [16].

4.3. Experiments on Cityscapes

4.3.1 Comparisons with state-of-the-arts

Results of other state-of-the-art semantic segmentation solutions on Cityscapes validation set are summarized in Tab. 1. We provide these results for reference and emphasize that these results should not be simply compared with our method, since these methods are trained on different (even larger) training sets or different basic network. Among these approaches, Deeplabv3 [6] and CCNet

Table 1. Comparison with state-of-the-arts on Cityscapes (val).

Method	Backbone	multi-scale	mIOU(%)
DeepLabv3 [6]	ResNet-101	Yes	79.3
DeepLabv3+ [8]	Xception-65	No	79.1
DPC [3] †	Xception-71	No	80.8
CCNet	ResNet-101	No	80.2
CCNet	ResNet-101	Yes	**81.3**

† use extra COCO dataset for training.

Table 2. Comparison with state-of-the-arts on Cityscapes (test).

Method	Backbone	mIOU(%)
DeepLab-v2 [5]	ResNet-101	70.4
RefineNet [22] ‡	ResNet-101	73.6
SAC [40] ‡	ResNet-101	78.1
GCN [27] ‡	ResNet-101	76.9
DUC [30] ‡	ResNet-101	77.6
ResNet-38 [32]	WiderResnet-38	78.4
PSPNet [41]	ResNet-101	78.4
BiSeNet [35] ‡	ResNet-101	78.9
AAF [19]	ResNet-101	79.1
PSANet [42] ‡	ResNet-101	80.1
DFN [36] ‡	ResNet-101	79.3
DenseASPP [34] ‡	DenseNet-161	80.6
CCNet ‡	ResNet-101	**81.4**

‡ train with both the train-fine and val-fine datasets.

adopt the same backbone and multi-scale testing strategy. Deeplabv3+ [8] and DPC [3] both use a more stronger backbone (*i.e.*, Xception-65 & 71 *vs.* ResNet-101). In addition, DPC [3] makes use of additional dataset, *i.e.*, COCO, for pre-training beyond the training set of Cityscapes. The results show that the proposed CCNet with multi-scale testing still outperforms all these strong baselines.

Additionally, we also train the best learned CCNet with ResNet-101 as the backbone using both training and validation sets and make the evaluation on the test set by submitting our test results to the official evaluation server. Most of methods [5, 22, 40, 27, 30, 41, 35, 19, 42, 36] adopt the same backbone as ours and the others [32, 34] utilize stronger backbones. From Tab. 2, it can be observed that our CCNet substantially outperforms all the previous state-of-the-arts. Among the approaches, PSANet [42] is most related to our method which generates sub attention map for each pixel. One of the differences is that the sub attention map has $2 \times H \times W$ weights in PSANet and $H + W - 1$ weights in CCNet. Even with lower computation cost and memory usage, Our method still achieves better performance.

Table 3. Performance on Cityscapes (val) for different number of loop in RCCA. FLOPs and Memory increment are estimated for an input of $1 \times 3 \times 769 \times 769$.

Loops	GFLOPs(▲)	Memory(M▲)	mIOU(%)
baseline	0	0	75.1
R=1	8.3	53	78.0
R=2	16.5	127	79.8
R=3	24.7	208	80.2

4.3.2 Ablation studies

To verify the rationality of the CCNet, we conduct extensive ablation experiments on the validation set of Cityscapes with different settings for CCNet.

The effect of the RCCA module Tab. 3 shows the performance on Cityscapes validation set by adopting different number of loop in RCCA. All experiments are conducted using ResNet-101 as the backbone. Beside, the input size of one image is 769×769, resulting in the size of input feature maps H of RCCA is 97×97. Our baseline network is the ResNet-based FCN with dilated convolutional module incorporated at stage 4 and 5, *i.e.*, dilations are set to 2 and 4 for these two stages respectively. The increment of FLOPs and Memory usage are estimated when $R = 1, 2, 3$, respectively.

We observe that adding a criss-cross attention into the baseline, donated as $R = 1$, improves the performance by 2.9% compared with the baseline, which can effectively demonstrate the significance of criss-cross attention. Furthermore, increasing loops from 1 to 2 can improve the performance by 1.8%, demonstrating the effectiveness of dense contextual information. Finally, increasing loops from 2 to 3 slightly improves the performance by 0.4%. Meanwhile, with the increasing of loops, the usage of FLOPs and GPU memory will still be increased. These results prove that the proposed criss-cross attention can significantly improve the performance by capturing contextual information in horizontal and vertical direction. In addition, the proposed criss-cross attention is effective in capturing the dense and global contextual information, which can finally benefit the performance of semantic segmentation. To balance the performance and resource usage, we choose $R = 2$ as default settings in all the following experiments.

To further validate the effectiveness of the criss-cross module, We provide the qualitative comparisons in Fig. 5. We leverage the *white circles* to indicate those challenging regions that are easily to be misclassified. It can be seen that these challenging regions are progressively corrected with the increasing of loops, which can well prove the effectiveness of dense contextual information aggregation for semantic segmentation.

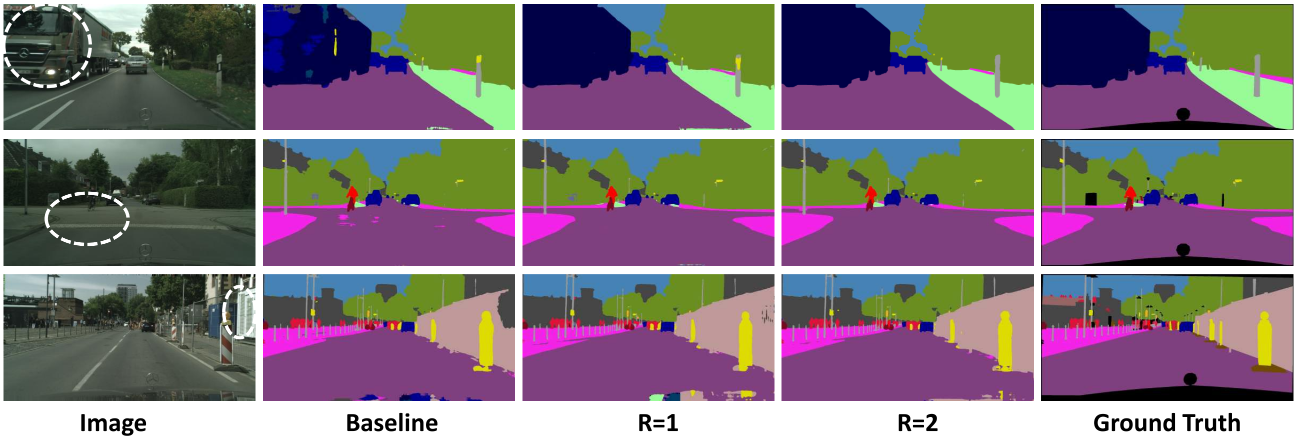

Figure 5. Visualization results of RCCA with different loops on Cityscapes validation set.

Table 4. Comparison of context aggregation approaches on Cityscapes (val).

Method	mIOU(%)
ResNet101-Baseline	75.1
ResNet101+GCN	78.1
ResNet101+PSP	78.5
ResNet101+ASPP	78.9
ResNet101+NL	79.1
ResNet101+RCCA(R=2)	79.8
ResNet50-Baseline	73.3
ResNet50+GCN	76.2
ResNet50+PSP	76.4
ResNet50+ASPP	77.1
ResNet50+NL	77.3
ResNet50+HV	77.3
ResNet50+HV&VH	77.8
ResNet50+RCCA(R=2)	78.5

Comparison of context aggregation approaches We compare the performance of several different context aggregation approaches on the Cityscapes validation set with ResNet-50 and ResNet-101 as backbones.

Specifically, the baselines of context aggregation mainly include: 1) Peng *et al.* [27] utilized global convolution network for contextual information aggregation, donated as "+GCN". 2) Zhao *et al.* [41] proposed Pyramid pooling which is the simple and effective way to capture global contextual information, donated as "+PP"; 3) Chen *et al.* [6] used different dilation convolutions to harvest pixel-wise contextual information at the different range, donated as "+ASPP"; 4) Wang *et al.* [31] introduced non-local network for context aggregation, donated as "+NL".

In Tab. 4, both "+NL" and "+RCCA" achieve better performance compared with other the context aggregation approaches, which demonstrates the importance of capturing full-image contextual information. More interestingly, our method achieves better performance than "+NL". This reason may be attributed to the sequentially recurrent operation of criss-cross attention. Concretely, "+NL" generates an attention map directly from the feature which has limit receptive field and short-range dependencies. In contrast, our "+RCCA" takes two steps to form dense contextual information, leading to that the latter step can learn a better attention map benefiting from the feature maps produced by the first step in which some long-range dependencies has already been embedded.

To prove the effectiveness of attention with criss-cross shape, we compare criss-cross shape with other shapes in Tab. 4. "+HV" means stacking horizontal attention and vertical attention. "+HV&VH" means summing up features of two parallel branches: "HV" and "VH". These results prove that criss-cross attention can achieve better performance than other shapes.

We further explore the amount of computation and memory footprint of RCCA. As shown in Tab. 5, compared with "+NL" method, the proposed "+RCCA" requires 11× less GPU memory usage and significantly reduce FLOPs by about 85% of non-local block in computing full-image dependencies, which shows that the CCNet is an efficient way to capture full-image contextual information in the least amount of computation and memory footprint.

Visualization of Attention Map To get a deeper understanding of our RCCA, we visualize the learned attention masks as shown in Fig. 6. For each input image, we select one point (cross in green color) and show its corresponding attention maps when $R = 1$ and $R = 2$ in columns 2 and 3, respectively. It can be observed that only contextual information from the criss-cross path of the target point is capture when $R = 1$. By adopting one more criss-cross modules, *i.e.*, $R = 2$, RCCA can finally aggregate denser and

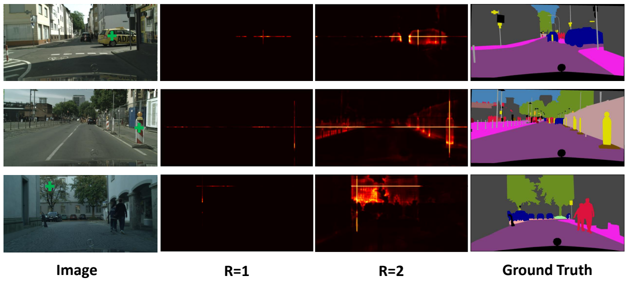

Figure 6. Visualization of attention module on Cityscapes validation set. The left column is the input images, the 2 and 3 columns are pixel-wise attention maps when $R = 1$ and $R = 2$ in RCCA.

Table 5. Comparison of Non-local module and RCCA. FLOPs and Memory increment are estimated for an input of $1 \times 3 \times 769 \times 769$.

Method	GFLOPs(▲)	Memory(M▲)	mIOU(%)
baseline	0	0	73.3
+NL	108	1411	77.3
+RCCA(R=2)	16.5	127	78.5

richer contextual information compared with that of $R = 1$. Besides, we observe that the attention module could capture semantic similarity and full-image dependencies.

4.4. Experiments on ADE20K

In this subsection, we conduct experiments on the AED20K dataset, which is a very challenging scene parsing dataset. As shown in Tab. 6, CCNet achieves the state-of-the-art performance of 45.22%, outperforms the previous state-of-the-art methods by more than 0.6%. Among the approaches, most of methods [40, 41, 42, 20, 33, 39] adopt the ResNet-101 as backbone and RefineNet [22] adopts a more powerful network, *i.e.*, ResNet-152, as the backbone. EncNet [39] achieves previous best performance among the methods and utilizes global pooling with image-level supervision to collect image-level context information. In contrast, our CCNet adopts an alternative way to integrate contextual information by capture full-image dependencies and achieve better performance.

4.5. Experiments on COCO

To further demonstrate the generality of CCNet, we conduct the instance segmentation task on COCO [23] using the competitive Mask R-CNN model [16] as the baseline. Following [31], we modify the Mask R-CNN backbone by adding the RCCA module right before the last convolutional residual block of res4. We evaluate a standard baseline of ResNet-50/101. All models are fine-tuned from ImageNet pre-training. We use the official implementation[1] with end-to-end joint training whose performance is almost the same as the baseline reported in [31]. We report the results in terms of box AP and mask AP in Tab. 7 on COCO. The results demonstrate that our method substantially outperforms the baseline in all metrics. Meanwhile, the network with "+RCCA" also achieve the better performance than the network with one non-local block "+NL".

Table 6. Comparison with state-of-the-arts on ADE20K (val).

Method	Backbone	mIOU(%)
RefineNet [22]	ResNet-152	40.70
SAC [40]	ResNet-101	44.30
PSPNet [41]	ResNet-101	43.29
PSANet [42]	ResNet-101	43.77
DSSPN [20]	ResNet-101	43.68
UperNet [33]	ResNet-101	42.66
EncNet [39]	ResNet-101	44.65
CCNet	ResNet-101	**45.22**

Table 7. Comparisons on COCO (val).

Method		AP^{box}	AP^{mask}
R50	baseline	38.2	34.8
	+NL	39.0	35.5
	+RCCA	**39.3**	**36.1**
R101	baseline	40.1	36.2
	+NL	40.8	37.1
	+RCCA	**41.0**	**37.3**

5. Conclusion and future work

In this paper, we have presented a Criss-Cross Network (CCNet) for deep learning based dense prediction tasks, which adaptively captures contextual information on the criss-cross path. To obtain dense contextual information, we introduce RCCA which aggregates contextual information from all pixels. The experiments demonstrate that RCCA captures full-image contextual information in less computation cost and less memory cost. Our CCNet achieves outstanding performance consistently on two semantic segmentation datasets, *i.e.* Cityscapes, ADE20K and instance segmentation dataset, *i.e.* COCO.

Acknowledgements

This work was supported by NSFC (No. 61876212, No. 61733007 and No. 61572207), the fund of HUST-Horizon Computer Vision Research Center, China Scholarship Council, Hubei Scientific and Technical Innovation Key Project, IBM-Illinois Center for Cognitive Computing Systems Research (C3SR) and ARC DECRA DE190101315.

[1] https://github.com/facebookresearch/maskrcnn-benchmark

References

[1] Ronald T Azuma. A survey of augmented reality. *Presence: Teleoperators & Virtual Environments*, 6(4):355–385, 1997. 1

[2] Bowen Chen, Liang-Chieh Chen, Yunchao Wei, Yukun Zhu, Zilong Huang, Jinjun Xiong, Thomas Huang, Wen-Mei Hwu, and Honghui Shi. Spgnet: Semantic prediction guidance for scene parsing. In *iccv*, 2019. 2

[3] Liang-Chieh Chen, Maxwell D Collins, Yukun Zhu, George Papandreou, Barret Zoph, Florian Schroff, Hartwig Adam, and Jonathon Shlens. Searching for efficient multi-scale architectures for dense image prediction. *arXiv preprint arXiv:1809.04184*, 2018. 3, 6

[4] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014. 2, 3, 5

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018. 1, 2, 3, 5, 6

[6] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2, 5, 6, 7

[7] Liang-Chieh Chen, Yi Yang, Jiang Wang, Wei Xu, and Alan L Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3640–3649, 2016. 3

[8] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. *arXiv preprint arXiv:1802.02611*, 2018. 2, 6

[9] Jianpeng Cheng, Li Dong, and Mirella Lapata. Long short-term memory-networks for machine reading. *arXiv preprint arXiv:1601.06733*, 2016. 2

[10] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 2, 5

[11] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. *CoRR, abs/1703.06211*, 1(2):3, 2017. 2

[12] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[13] Martin Evening. *Adobe Photoshop CS3 for photographers: a professional image editor's guide to the creative use of Photoshop for the Macintosh and PC*. Focal press, 2012. 1

[14] Jannik Fritsch, Tobias Kuehnl, and Andreas Geiger. A new performance measure and evaluation benchmark for road detection algorithms. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 1693–1700. IEEE, 2013. 1

[15] Jun Fu, Jing Liu, Haijie Tian, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. *arXiv preprint arXiv:1809.02983*, 2018. 3

[16] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017. 5, 8

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 5

[18] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. *arXiv preprint arXiv:1709.01507*, 7, 2017. 3

[19] Tsung-Wei Ke, Jyh-Jing Hwang, Ziwei Liu, and Stella X Yu. Adaptive affinity field for semantic segmentation. *arXiv preprint arXiv:1803.10335*, 2018. 3, 6

[20] Xiaodan Liang, Hongfei Zhou, and Eric Xing. Dynamic-structured semantic propagation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 752–761, 2018. 8

[21] Di Lin, Yuanfeng Ji, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Multi-scale context intertwining for semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 603–619, 2018. 2

[22] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian D Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Cvpr*, volume 1, page 5, 2017. 2, 6, 8

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 5, 8

[24] Sifei Liu, Shalini De Mello, Jinwei Gu, Guangyu Zhong, Ming-Hsuan Yang, and Jan Kautz. Learning affinity via spatial propagation networks. In *Advances in Neural Information Processing Systems*, pages 1520–1530, 2017. 3

[25] Ziwei Liu, Xiaoxiao Li, Ping Luo, Chen-Change Loy, and Xiaoou Tang. Semantic image segmentation via deep parsing network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1377–1385, 2015. 3

[26] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1, 2

[27] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. Large kernel mattersimprove semantic segmentation by global convolutional network. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1743–1751. IEEE, 2017. 3, 6, 7

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmen-

tation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 2

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. 2, 3

[30] Panqu Wang, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou, and Garrison Cottrell. Understanding convolution for semantic segmentation. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1451–1460. IEEE, 2018. 6

[31] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. *arXiv preprint arXiv:1711.07971*, 10, 2017. 1, 2, 3, 7, 8

[32] Zifeng Wu, Chunhua Shen, and Anton van den Hengel. Wider or deeper: Revisiting the resnet model for visual recognition. *arXiv preprint arXiv:1611.10080*, 2016. 6

[33] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. *arXiv preprint arXiv:1807.10221*, 2018. 8

[34] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3684–3692, 2018. 3, 6

[35] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. *arXiv preprint arXiv:1808.00897*, 2018. 3, 6

[36] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Learning a discriminative feature network for semantic segmentation. *arXiv preprint arXiv:1804.09337*, 2018. 2, 6

[37] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015. 2, 3

[38] Yuhui Yuan and Jingdong Wang. Ocnet: Object context network for scene parsing. *arXiv preprint arXiv:1809.00916*, 2018. 3

[39] Hang Zhang, Kristin Dana, Jianping Shi, Zhongyue Zhang, Xiaogang Wang, Ambrish Tyagi, and Amit Agrawal. Context encoding for semantic segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 5, 8

[40] Rui Zhang, Sheng Tang, Yongdong Zhang, Jintao Li, and Shuicheng Yan. Scale-adaptive convolutions for scene parsing. In *Proc. 26th Int. Conf. Comput. Vis.*, pages 2031–2039, 2017. 2, 6, 8

[41] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890, 2017. 2, 3, 6, 7, 8

[42] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. Psanet: Point-wise spatial attention network for scene parsing. In *European Conference on Computer Vision*, pages 270–286. Springer, 2018. 2, 3, 6, 8

[43] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1529–1537, 2015. 3

[44] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, page 4. IEEE, 2017. 2, 5