Context-Reinforced Semantic Segmentation

Yizhou Zhou*[1] Xiaoyan Sun[2] Zheng-Jun Zha†[1] Wenjun Zeng[2]
[1]University of Science and Technology of China [2]Microsoft Research Asia
zyz0205@mail.ustc.edu.cn, zhazj@ustc.edu.cn {xysun,wezeng}@microsoft.com

Abstract

*Recent efforts have shown the importance of context on deep convolutional neural network based semantic segmentation. Among others, the predicted segmentation map (p-map) itself which encodes rich high-level semantic cues (e.g. objects and layout) can be regarded as a promising source of context. In this paper, we propose a dedicated module, Context Net, to better explore the context information in p-maps. Without introducing any new supervisions, we formulate the context learning problem as a Markov Decision Process and optimize it using reinforcement learning during which the p-map and Context Net are treated as environment and agent, respectively. Through adequate explorations, the Context Net selects the information which has long-term benefit for segmentation inference. By incorporating the Context Net with a baseline segmentation scheme, we then propose a **C**ontext-re**i**nforced **S**emantic **S**egmentation network (CiSS-Net), which is fully end-to-end trainable. Experimental results show that the learned context brings 3.9% absolute improvement on mIoU over the baseline segmentation method, and the CiSS-Net achieves the state-of-the-art segmentation performance on ADE20K, PASCAL-Context and Cityscapes.*

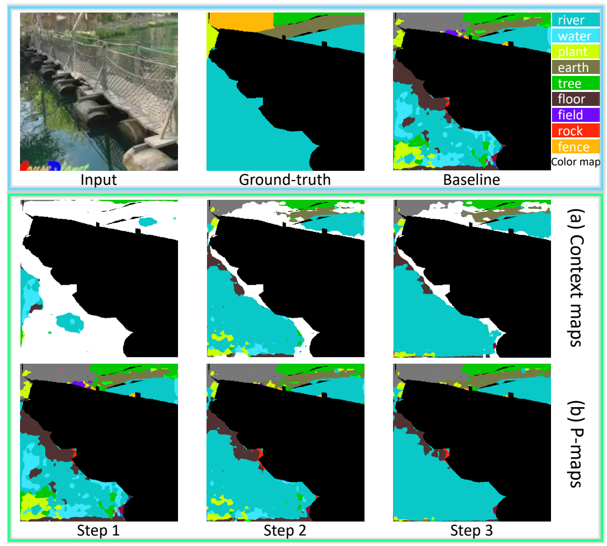

Figure 1. Visualized segmentation results (also denoted as predicted segmentation maps (p-map)) and context maps in our CiSS-Net. (a) Context maps at 3 steps. Here the white areas denote uncertain regions that are excluded from valid contextual information. The context map shown here upgrades gradually. (b) p-map at 3 steps. The prediction improves gradually due to the context-reinforced learning. It can be observed that reflection of trees in the water which easily deceives the baseline algorithm is well handled by our CiSS-Net based on the learned context.

1. Introduction

Semantic image segmentation is a fundamental and challenging task in computer vision. It interprets an image by assigning each pixel a semantic label. These semantic labels provide high-level semantic information varying from the layout of a scene to the category, location, and shape of each individual object in an image, which makes semantic image segmentation essential for many intelligent systems, such as autonomous driving and image editing.

Context is known to be essential for semantic segmentation [11]. Classifying a local region/pixel with regard to its surroundings is super helpful for reducing local ambiguities. However, the fully convolutional network (FCN), which is the most popular baseline of current successful semantic segmentation methods, lacks suitable strategies to utilize global information about an image. In order to include more information, dilated/atrous convolutions are employed in FCN to expand the receptive field of convolutions and enable dense predictions [5, 34, 4, 6]. Also, multi-scale/multi-stage features are assembled to exploit information from different scales/layers to improve the segmentation performance [21, 30, 16, 7, 13].

Recent efforts focus more on integrating global context features in FCNs. Global average pooling is adopted to pool a global context feature from one of the last few layers of FCN, which is then fused with local features to provide

* This work was performed while Yizhou Zhou was an intern with Microsoft Research Asia. † Corresponding author.

global hints for segmentation [23]. Based on the atrous convolution, the atrous spatial pyramid pooling network [5] is presented to capture multi-scale features by involving multiple parallel atrous filters at different rates. In addition to the dilated FCN, PSPNet in [37] extends the pixel-level features to the global pyramid pooled one which fuses features under four different pyramid scales. In addition to the multi-scale and multi-level features, Ding *et al.* introduce a context contrasted local feature to highlight the difference between each local and global features and boost the performance for inconspicuous objects[11]. In [35], a context encoding module is presented to capture the orderless correlations between scene context and probabilities of categories in feature maps. Lin *et al.* [22] explicitly model the patch-patch and patch-background contextual correlations via trainable pairwise potential functions and multi-scale sliding pooling. Huang *et al.* [15] introduce a extra network into semantic segmentation based on scene similarities to encode the global scene information followed by an image retrieval module which captures non-parametric prior information for the input image.

Besides aforementioned global/local context features used in feature learning, we notice that the predicted segmentation map (p-map, which refers to the intermediate segmentation map as well as the final result) generated by each semantic segmentation method already encodes rich high-level semantic cues both locally (*e.g.*, objects) and globally (*e.g.*, layouts) and can be another good candidate for context. In addition, the dimension of p-maps usually is much lower compared with that of feature maps in deep networks, which could facilitate the exploration of context. Therefore, it is beneficial to incorporate such information into feature learning. However, as illustrated in Fig.1 (b), a p-map often contains lots of noises such as misclassified regions and chaotic objects, which makes it very challenging to use p-maps as context for semantic segmentation.

In fact, p-map has been used in previous works to refine itself either by exploiting Conditional Random Field (CRF) [19] on top of the p-map in a post-processing fashion [4, 5, 32, 3], concatenating a trainable network to facilitate end-to-end training [38, 24, 31], or incorporating a recurrent architecture to enable coarse-to-fine refinement [36, 28, 17]. Different from these algorithms that set out to refine its accuracy, we seek to fully explore the p-map to generate another source of scene context that can be effectively combined with the traditional features to further improve the segmentation performance in a recursive manner.

More specifically, we propose a Context-reinforced Semantic Segmentation Network (CiSS-Net) to explore the high-level semantic context information in p-maps to further enhance modern semantic segmentation methods. Our CiSS-Net consists of two sub-networks, Context Net (C-Net) which is a dedicated network to learn effective semantic context from p-maps, and Segment Net (S-Net) which embeds the learned context in the inference of FCN-based segmentation. Since it is hard to tell which information should be selected from p-maps as learned context, we choose not to introduce any new supervisions on context. Instead, we formulate the context learning problem as a Markov Decision Process (MDP) and propose learning the context via interactions between C-Net and S-Net. Naturally, the optimization problem can be solved through deep reinforcement learning (RL) by treating the p-maps as environment and the C-Net as agent. As shown in Fig.1, thanks to the exploration of long-term benefit during the context-reinforced learning, the p-map improves step by step based on the gradually upgraded context maps, and the challenging water area is well segmented by our CiSS-Net based on the learned context.

In summary, our main contributions are three-fold:

- We propose exploring high-level semantic context from p-maps by a dedicated network which will be embedded in the inference of the FCN-based semantic segmentation.
- Without any new supervisions, we formulate the context learning problem as a MDP and propose learning the context using reinforcement learning so that it has long-term benefits on segmentation inference, by reciprocally interacting with the segmentation network.
- We propose a fully end-to-end context-reinforced semantic segmentation network that efficiently facilitates the above learning process and achieves state-of-the-art performance on three popular segmentation datasets.

2. Context-reinforced Semantic Segmentation

We propose a new framework, CiSS-Net, which recursively extracts context from p-maps and then embeds it to improve the segmentation performance. For better understanding, we first present the overall framework of the CiSS-Net and then give detailed descriptions of the two main modules, Context Net and Segment Net, respectively.

2.1. The Framework

The overall framework of our CiSS-Net is shown in Fig.2. The CiSS-Net consists of two modules, Segment Net (S-Net) and Context Net (C-Net). The S-Net is designed to infer p-maps of input images given the additional context maps generated by C-Net through exploring both the inputs and outputs of S-Net. The S-Net and C-Net work interactively for both segmentation and context learning. To be more specific, the S-Net predicts a segmentation map based on the input image features as well as the generated context; the C-Net is then fed with both the p-map as well as

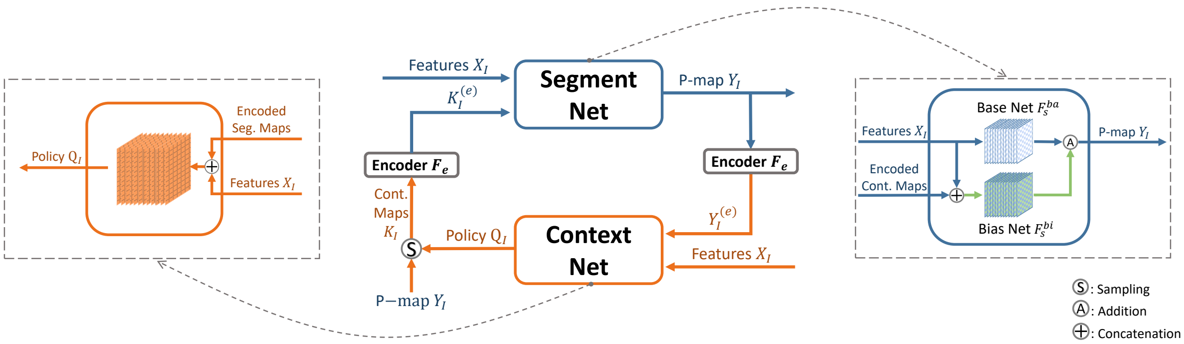

Figure 2. Overview of our proposed Context-reinforced Semantic Segmentation Network. Our CiSS-Net has two sub-networks, Segment Net and Context Net. These two sub-networks mutually benefit each other and work iteratively. The Segment Net takes the encoded context maps as additional information to generate the segmentation prediction, while the prediction is then used as conditions for Context Net to produce new context maps to further improve the segmentation prediction.

the input image features to generate new context. By properly defining the state, action and transition matrix, the context can be explicitly learned through reinforcement learning towards improving the segmentation performance without any extra supervision.

Before elaborating on each module, we first provide a short list of notations used in the following description. Given an input image I, we have

- Context map $K_I \in \{0, 1, 2, ..., N_c\}^{H' \times W'}$, where N_c, H' and W' are the number of classes, height and width of K_I, respectively.
- Domain features $X_I \in \mathbb{R}^{H' \times W' \times C}$, where C is the channel size of the feature map.
- P-map $Y_I \in \mathbb{R}^{H^o \times W^o}$, where H^o and W^o denote the spatial resolution of a segmentation prediction.
- Pyramid Pooling Module (Encoder in the figure) $F_e : \mathbb{R}^{H' \times W'} \rightarrow \mathbb{R}^{H' \times W' \times C_e}$, where C_e is the channel size of output.
- Encoded context map and p-map $K_I^{(e)}, Y_I^{(e)} \in \mathbb{R}^{H' \times W' \times C_e}$.
- Base Net $F_s^{ba} : \mathbb{R}^{H' \times W' \times C} \rightarrow \mathbb{R}^{H^o \times W^o \times N_c}$ and Bias Net $F_s^{bi} : \mathbb{R}e^{H' \times W' \times (C+C_e)} \rightarrow \mathbb{R}^{H^o \times W^o \times N_c}$.
- Context Net $F_k : \mathbb{R}^{H' \times W' \times (C+C_e)} \rightarrow \mathbb{R}^{H^o \times W^o \times 2}$.
- Policy $Q_I \in \mathbb{R}^{H^o \times W^o \times 2}$.

2.2. Segment Net

As shown in Fig.2, the S-Net has two inputs, the context map K_I and domain features X_I of the input image I. The context map is a two-dimensional semantic map derived from p-maps by the C-Net, which will be introduced in details in subsection 2.3. As illustrated in Fig.1 (b), a context map encodes certain semantic information, e.g. the layout of a scene and category of objects, but with an extra class 'uncertain'. Regions assigned with 'uncertain' will be ignored in the S-Net during inference.

In order to efficiently take advantages of the context map during the inference, we apply some special designs for the S-Net. Since lots of low-level features, such as textures and boundaries, will be extracted by the first several convolutional layers, it is neither reasonable nor efficient to fuse the high-level semantic context with those low-level features. We thus adopt a pre-trained convolutional neural network (CNN) to extract mid-level domain features X_I instead of using the raw image. A Pyramid Pooling Module $F_e(\cdot)$ [37] as shown in Fig.3 (a) is also employed to encode context map into multiple spatial levels, which reveal more global information at each spatial position. Then the encoded context map $K_I^{(e)} = F_e(K_I)$ is concatenated with X_I and fed into the S-Net.

Our S-Net $F_s(X_I, K_I^{(e)})$ is composed of two subnetworks, Base Net F_s^{ba} and Bias Net F_s^{bi}. It can be formulated as

$$\begin{cases} F_s^{ba}(X_I), & if\ K_I\ is\ None \\ F_s^{ba}(X_I) + F_s^{bi}(X_I \oplus K_I^{(e)}), & otherwise \end{cases} \tag{1}$$

where $\oplus$ denotes concatenation operation and $K_I\ is\ None$ refers that there is not context map provided. Therefore, p-map Y_I is derived as $Y_I = argmax(F_s(X_I, K_I))$. As shown in Fig.2 and Eq. (1), the Base Net only processes X_I to infer a basic segmentation map, while the Bias Net is fed with the context embedded feature $X_I \oplus K_I^{(e)}$ to learn a conditional mask for each class that reflects the per-class bias based on the context map K_I. The mask is then added to the basic segmentation map (the tensor before Softmax activation) to further rectify the predictions. Inspired by the idea of residual learning [14], this design helps to both reduce the learning complexity and emphasize the role of context.

2.3. Context Net

The structure of C-Net is illustrated on the left side of Fig. 2. Our C-Net also has two inputs, the domain features X_I and p-map Y_I generated by the S-Net. Learning

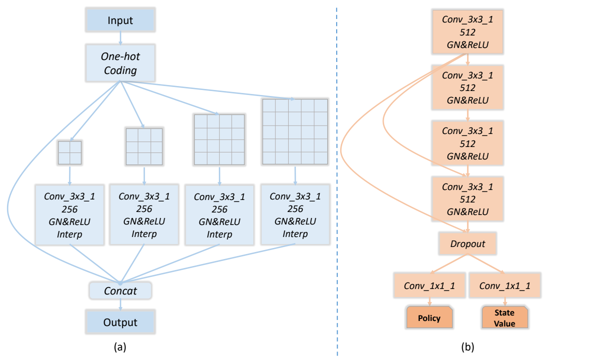

Figure 3. Illustrations of (a) the architecture of Pyramid Pooling Module (Grid rectangle blocks are spatial pooling layers with different pooling sizes and strides) and (b) the architecture of C-Net.

context K_I on top of Y_I makes K_I conditioned on Y_I, and by treating the prediction Y_I as a state that will be updated based on the content of the context K_I, the context learning process in C-Net can be reformulated as a Markov decision chain $Y_I^0 \xRightarrow{K_I^0} Y_I^1 \xRightarrow{K_I^1} \cdots \xRightarrow{K_I^{N-1}} Y_I^N$. This indicates that K_I can be learned to incrementally improve the segmentation through reinforcement learning. Our C-Net is instantiated with a five-layer CNN as illustrated in Fig. 3 (b). The input of the C-Net is the concatenation of the two signals X_I and $Y_I^{(e)}$. The output of the C-Net F_k is a policy map $Q_I = F_k(X_I \oplus Y_I^{(e)})$, where the value of $Q(i,j,k)$ indicates the probability of taking action k at position (i,j). We define $k \in \{0,1\}$, where $k = 1$ denotes the action of adopting the prediction $Y_I(i,j)$ as context whereas $k = 0$ ignores the prediction. Then a binary decision $B_I(i,j) \sim Q_I(i,j)$ is sampled at each position to generate the context map $K_I = (Y_I + 1) \circ B_I$, where $\circ$ denotes the element-wise matrix multiplication. Consequently, each value in K_I signifies the corresponding index of classes (indexed from one), except for the number '0' that represents the class 'uncertain'.

Having both Eq. 1 and $K_I = (Y_I + 1) \circ B_I$, we can observe a mutual dependency between the predicted segmentation map Y_I and context map K_I. One can decouple the dependency along the time domain as

$$\begin{aligned} Y_I^{t+1} &= F_s(X_I, K_I^t) \\ K_I^t &= (Y_I^t + 1) \circ B^t \\ where\ B^t &\sim F_k(X_I \oplus F_e(Y_I^t)), \end{aligned} \tag{2}$$

where t is the iteration index. The decoupled dependence reveals that the Y_I and K_I can be regarded as a state-action pair. Therefore, an infinite-horizon discounted Markov decision process (MDP) can be naturally defined with the tuple $(\mathbb{S}, \mathbb{A}, P, r, \rho_0, \gamma)$, where $\mathbb{S}$ is a finite set of states defined as $\mathbb{S} = \{Y_I\}$, $\mathbb{A}$ is a finite set of actions defined as $\mathbb{A} = \{B_I\}$, $P : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$ is the transition probability distribution defined as $P(s_{t+1} = F_s(X_I, (s_t + 1) \circ a_t)|s_t) = 1$ ($s_t \in \mathbb{S}$ and $a_t \in \mathbb{A}$), $\rho_0 : \mathbb{S} \to \mathbb{R}$ is the probability distribution for the initial state, $\gamma \in [0,1]$ is the discount factor and $r : \mathbb{S} \times \mathbb{A} \to \mathbb{R}^{H' \times W'}$ is the reward function. Letting π_{F_k} denote the probability distribution over the outputs of the C-Net and $\eta(F_k)$ denote the expected discount reward under C-Net F_k, we have

$$\begin{aligned} &\eta(F_k) = \mathbb{E}_{s_0, a_0, \ldots}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right],\ where \\ &s_0 \sim \rho_0, a_t \sim \pi_{F_k}(a_t|s_t), s_{t+1} \sim P(s_{t+1}|s_t). \end{aligned} \tag{3}$$

The behavior of the C-Net can be aligned to the reward function r by maximizing the expected future discount reward $\eta(F_k)$. In order to enable the context to incrementally improve the segmentation, we compute $r_t(i,j)$, *i.e.*, the reward on spatial location (i,j) at time step t as

$$\begin{aligned} &\frac{1}{C_h C_w} \sum_{i',j'} M(Y_I^t(i',j'), Y_I^{t+1}(i',j'), L_I(i',j')) \\ &+ \beta_1 \mathbb{1}_{L_I(i,j)}(K_I^t(i,j)) + \beta_2 \mathbb{1}_0(K_I^t(i,j)). \end{aligned} \tag{4}$$

The reward function has three terms. The first term $M()$ is a measurement function that calculates how much improvement is made from Y_I^t to Y_I^{t+1} for a given location, where $i' \in [i - C_h/2, i + C_h/2]$, $j' \in [j - C_w/2, j + C_w/2]$ and C_h/C_w denotes the height/width of the region considered in the reward computing for the action at position (i, j). This is used to encourage the generated context map to improve the segmentation performance. More specifically, given the $L_I(i',j')$, *i.e.* the ground-truth at location i', j', $M()$ first computes the correctness of the prediction at time t and $t+1$ respectively. There are four different cases, *i.e.* $Y_I^t(i',j')$ is correct/incorrect $\to Y_I^{t+1}(i',j')$ is correct/incorrect. We assign a reward 1 to the case 'incorrect $\to$ correct', -1 to the case 'correct $\to$ incorrect', 0 to the case 'incorrect $\to$ incorrect' and 0.5 to the case 'correct $\to$ correct'. Because the value of K_I^t at position (i,j) actually takes effect on a $C_h \times C_w$ rectangle region of Y_I^{t+1} (depends on the receptive field of the C-Net), the scores computed by $M()$ are averaged in the target region. We simply ignore the associated regions that go beyond the boundary of the image. The second and third terms $\mathbb{1}_{L_I(i,j)}(K_I^t(i,j))$ and $\mathbb{1}_0(K_I^t(i,j))$ are indicator functions that assign a smaller positive rewards β_1 and β_2 for the correct context (the semantic information that is consistent with the ground-truth) and 'uncertain' respectively, since effective context is always supposed to be correct information.

2.4. Context-reinforced Segmentation

We employ the asynchronous advantage actor-critic algorithm [26] to optimize the MDP presented in the previous section, and the following standard definitions are used for the value function V_{F_k}, the state-action value function Q_{F_k}

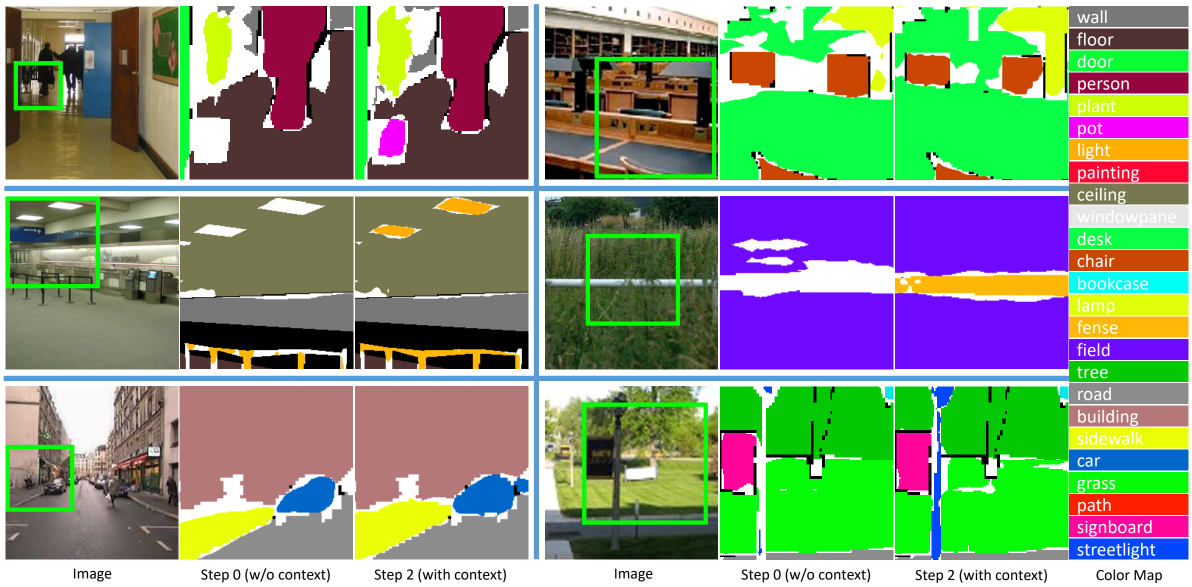

Figure 4. Visualized segmentation results of our CiSS-Net. For each input image, we show two segmentation results generated at iteration step 0 (Baseline) and 2 (with learned context). Segmentation results shown here are enlarged portions denoted by the green boxes in the input images. White areas in the segmentation results show the misclassified regions. We can observe that objects/stuffs (such as the pot, plant, lamp and desk in the first row, the fence and the light in the second row, and the car, sidewalk, streetlight in the third row) which are misclassified in the initial stage can be segmented much more accurately at stage 2 by involving the learned context.

and the advantage function A_{F_K}

$$\begin{aligned} V_{F_k}(s_t) &= \mathbb{E}_{a_t, s_{t+1}\cdots}\left[\sum_{l=0}^{\infty}\gamma^l r_{t+l}\right] \\ Q_{F_k}(s_t, a_t) &= \mathbb{E}_{s_{t+1}, a_{t+1}\cdots}\left[\sum_{l=0}^{\infty}\gamma^l r_{t+l}\right] \\ A_{F_k}(s, a) &= Q_{F_k}(s, a) - V_{F_k}(s), where \\ a_t &\sim \pi_{F_k}(a_t|s_t), s_{t+1} \sim P(s_{t+1}|s_t). \end{aligned} \tag{5}$$

where the value function V_{F_k} is estimated by a CNN $F_k^v(s_t)$ that shares the same weights as C-Net F_k except for the last layer as shown in Fig.3 (b), and the advantage function $A_{F_k}(s_t, a_t)$ is estimated by $\sum_{l=0}^{k-1}\gamma^l r_{t+l} + \gamma^k V_{F_k}(s_{t+k}) - V_{F_k}(s_t)$. The parameters of the C-Net F_k and the value function F_k^v are updated as

$$\begin{aligned} \theta_k &= \theta_k + \nabla_\theta \log\left[\pi_{F_k}(a_t|s_t;\theta_k)\right]A(a_t, s_t) \\ \theta_v &= \theta_v + \frac{\partial(R - F_k^v(s_t;\theta_v))^2}{\partial\theta_v}, where \\ R &= \sum_{l=0}^{T-1}\gamma^l r_{t+l} + \gamma^k V_{F_k}(s_{t+k}). \end{aligned} \tag{6}$$

Note that Eq.(5) and Eq.(6) together only enforce context to be beneficial to a fixed S-Net F_s (*i.e.* fixed the transition probability distribution in Eq.(3) and Eq.(5)) by maximizing $\eta(F_k)$ alone, and could lead to only numerical improvements to p-maps rather than selecting genuinely effective context. To tackle this problem, we update the two networks simultaneously and encourage more exploration of the C-Net. By doing so, the S-Net is promised to experience adequately different context configurations during training and thus prevents the local optimum. The final loss of the scheme is formulated as

$$Loss = Loss_p + Loss_v + \lambda_1 Loss_s + \lambda_2 Loss_e, \tag{7}$$

where $Loss_P = \log\left[\pi_{F_k}(a_t|s_t;\theta_k)\right]A(a_t, s_t)$ and $Loss_v = (R - F_k^v(s_t;\theta_v))^2$ are the policy loss and value loss and the update rules are defined in Eq.(6). $Loss_s$ is the cross-entropy loss of the segment prediction and $Loss_e = \pi_{F_k}\log\pi_{F_k}$ is the entropy regularization term of F_k to encourage adequate exploration.

3. Experiments

3.1. Datasets and Experimental Settings

ADE20K [39] provides more than 20K scene-centric images fully annotated with objects and object parts. It is divided into three subsets containing 20,210, 2,000, and 3,000 images for training, validation and testing, respectively. It has up to 150 classes with 1,038 different image-level labels including both objects and stuffs. Evaluations on ADE20K are made on both pixel-wise accuracy (pixAcc) and mean of the class-wise intersection over union (mIoU).

Cityscapes [8] contains 5,000 frames with pixel-level annotation and 20,000 weakly annotated images recorded in street scenes from 50 cities. It involves 19 categories of both objects and stuffs. The data split follows 2,975 for training, 500 for validation and 1,525 for testing. We

Step	Context	Cityscapes(%)	Ade20K(%)
0	no	77.59	40.97
2	yes	79.21	42.56
4	yes	79.29	42.63
6	yes	79.29	42.66

Table 1. mIoU of CiSS-Net on test set of Cityscapes and validation set of Ade20K at iteration steps $0, 2, 4$, and 6.

γ	Cityscapes		Ade20K	
	IoU_{cls}	$iIoU_{cls}$	$mIoU$	$pixAcc$
0.1	78.31%	54.39%	41.53%	78.76%
0.3	78.45%	54.61%	41.80%	79.47%
0.9	78.94%	54.97%	42.42%	80.51%

Table 2. Segmentation results (single scale) of our CiSS-Net on the validation sets of Cityscapes and Ade20K with different γ.

only use the fully annotated data for training to stimulate the context learning in our CiSS-Net. We use the Cityscapes official server to evaluate the performance on both class-wise/category-wise intersection over union (IoU class/category) and instance-level intersection-over-union (iIoU class/category).

Pascal Context [27] provides 4998 fully annotated images for training and 5105 images for testing, which are re-annotated from Pascal VOC. We use the most commonly used 60 classes (59 classes plus the "background" class) in our evaluation. The performance is evaluated on both global pixel accuracy (GPA) and mIoU.

3.2. Implementation Details

Fig. 3 (b) shows the architecture of the C-Net. We generate domain features X_I using the first four blocks of PSPNet [37] pre-trained on the three datasets (PSPNet-101 for Cityscapes and PSPNet-50 for both Ade20K and Pascal Context). In our S-Net, the Base Net has two convolutional layers of which the channel, stride and kernel sizes are 512, 1 and 3, respectively; the Bias Net uses the same architecture as the C-Net except that the last convolutional layer is modified to fit the input/output dimension. Group Normalization [33] is used as the normalization layer. The channel size in each group is 16.

The CiSS-Net is implemented with Tensorflow and sixteen Nvidia M40 GPUs. The batch size on each gpu is 2. The Dropout ratio is 0.1 and random mirroring as well as random resizing by a factor between 0.5 and 2.0 are adopted. On ADE20K, we randomly crop a 473×473 region in an image and employ Stochastic Gradient Descent (SGD) to train the network with initial learning rates of 2×10^{-3} for Base Net and 2×10^{-4} for both Bias Net and C-Net. We randomly crop a 713×713 region on CityScapes, and we employ SGD to train the network with initial learning rates of 5×10^{-4} for Base Net and 5×10^{-5} for Bias Net and C-Net on both Cityscapes and Pascal Context.

λ_2	Cityscapes		Ade20K	
	IoU_{cls}	$iIoU_{cls}$	$mIoU$	$pixAcc$
0.001	77.23%	53.62%	41.31%	78.68%
0.005	78.59%	54.47%	41.90%	79.52%
0.01	78.44%	54.38%	42.19%	80.07%
0.02	78.94%	54.97%	42.42%	80.51%
0.05	79.10%	55.04%	42.56%	80.77%

Table 3. Segmentation results (single scale) with different values of λ_2 on validation sets of Cityscapes and Ade20K.

3.3. Hyper-parameters

We analyze two important hyper-parameters, γ in Eq. (3) and λ_2 in Eq. (7), in our CiSS-Net. The parameter γ rewards long-term benefits and λ_2 effects the degree of exploration in the context-reinforced learning in the CiSS-Net. In all the following tests, parameters β_1 and β_2 in Eq. (4) are set to 0.4 and 0.2, respectively; $\lambda_1 = 1.0$ in Eq. (7) .

Table 2 exhibits the results with different values of γ. The result improves consistently with the increase of γ. It demonstrates that the long-term benefit in context learning plays an important role in our CiSS-Net. We thus set $\gamma = 0.9$ to encourage long-term benefits.

Table 3 gives the sensitivity analysis on λ_2. It shows that the performance of our CiSS-Net improves when λ_2 runs from 0.001 to 0.05 while the training does not converge well when $\lambda_2 > 0.15$. This suggests that a suitable amount of exploration is crucial for the learning of effective context. In our CiSS-Net, we set $\lambda_2 = 0.05$ to make a trade-off between convergence and exploration.

3.4. Ablation Study

In the ablation study, we first investigate the necessity of the RL-based learnable module for context generation and then discuss the effect of the number of iterations in RL on the performance of our CiSS-Net.

RL strategy. There are alternative ways to make use of the context information in p-maps under the framework of our CiSS-Net. As listed in Table 4, 'Baseline' denotes the approach without the learned context, *i.e.*, no context map is fed to the S-Net in our CiSS-Net; 'Baseline+p-map' shows the performance when the p-map is directly used as input to the S-Net as context, which is even lower than that of 'Baseline'; 'Baseline+gated(p-map)*' approximates the performance of using a gate function to generate context, where the approximation is made by assigning a very small number 0.001 to λ_2 in the RL to greatly suppress the exploration. Among all the methods, our CiSS-Net with RL-based context learning achieves the best performance.

Furthermore, different from the RNN-based attempts which take complete p-maps as inputs [24, 17, 38, 36], we seek to explore the p-map to generate another source of scene context that can be effectively combined with traditional features. Table 5 further signifies the benefits of us-

Method	Cityscapes(%)	Ade20K(%)
Baseline	76.36	40.97
Baseline+p-map	75.19	39.44
Baseline+gated(p-map) *	77.23	41.31
CiSS-Net	79.10	42.56

Table 4. Evaluation on the RL strategy in our CiSS-Net on the validation sets of Cityscapes and Ade20K. Four alternative approaches are tested, i.e. the baseline method, the baseline method fed directly with p-map, the baseline method with a gated p-map and the CiSS-Net. * indicates that we use the RL approach with $\lambda_2 = 0.001$ as a approximation to the gate function.

	[24]	[17]	[38]	[36]	w/o RL	Ours
City.	66.8	-	62.5	76.2^{*}	75.2^{*}	79.2^{*}
Ade.	-	34.6^{*}	-	42.6^{*}	$39.4^{\dagger}$	$42.6^{\dagger}$

Table 5. mIoU performance of CiSS-Net and other RNN-based methods. w/o RL is our method using complete p-maps. $*/\dagger$ denote results achieved based on ResNet101/50

ing our design of context learning to facilitate the p-maps as context. It demonstrates that our proposed RL-based context learning can effectively utilize the context information in p-maps to boost the performance of our CiSS-Net.

Iteration steps. Table.1 shows the performance of our CiSS-Net with regard to the iteration index t as denoted in Eq. (2). Note that the context map K_I^0 is set to all-zero and no context information from p-map is involved in the Segment-Net when $t = 0$. It can be observed that the performance improves noticeably by utilizing the learned context. Similar to many iterative optimization processes, the improvement becomes marginal as the iteration continues. Accordingly, we choose $t = 2$ in the following tests to balance the performance and time complexity of inference.

We also visualize the predicted segmentation maps at $t = 0$ and $t = 2$ in Fig. 4, respectively. It can be observed that the CiSS-Net is able to correct mis-segmented objects/regions as the learned context gets involved into the segmentation inference. For example, the CiSS-Net achieves much better segmentation results on the pot, plant, lamp and desk in the first row; the fence, light in the second row; and the car, sidewalk, streetlight in the third row, rather than only refining object boundaries.

3.5. Comparison with the state-of-the-art

We further evaluate the performance of our CiSS-Net by comparisons with the state-of-the-art semantic segmentation methods. In the following, the results of the Ciss-Net are given at $t = 2$ so that the inference time of our CiSS-Net is competitive to the comparison methods.

ADE20K Table 6 shows the comparison results on the validation set of ADE20K. ADE20K is a challenging dataset with complicated scenes and diverse objects. Results show that our CiSS-Net is able to benefit from the learned context and achieves the highest performance

Method	Backbone	mIoU	pixAcc
SegNet [1]		21.64	71.00
FCN [25]		29.39	71.32
DilatedNet [34]		32.31	73.55
Cascaded-SegNet [39]		27.51	71.83
Cascaded-DilatedNet [39]		34.90	74.52
RefineNet [21]	Res101	40.2	-
PSPNet [37]	Res101	41.96	80.64
GRN+LRN(single model) [36]	Res101	42.60	-
DSSPN-Softmax [20]	Res101	42.03	80.81
Global-Context [15]	Res101	38.37	77.76
PSPNet [37]	Res50	41.68	80.04
EncNet [35]	Res50	41.11	79.73
CiSS-Net (Ours)	Res50	**42.56**	**80.77**

Table 6. Segmentation results on the validation set of ADE20K. Our CiSS-Net achieves the best performance among algorithms with the same backbone ResNet-50. It also obtains competitive or even better performance in comparison to algorithms with the much complicated backbone ResNet-101.

Method	Backbone	GPA	mIoU
O2P [2]	-	-	18.1
CFM [10]	-	-	34.4
BoxSup [9]	VGG16	-	40.5
Context-CRF [22]	VGG16	71.5	43.3
FCN [25]	FCN-8s	67.5	39.1
CRF-RNN [38]	FCN-8s	-	39.3
RefineNet [21]	Res101	-	47.1
Context-Contrasted [11]	Res101	**78.4**	**51.6**
Context-Contrasted (CCL) [11]	Res101	76.6	48.3
Global-Context [15]	Res101	73.8	46.5
PSPNet† [37]	Res101	76.0	47.8
Context-Contrasted [11]	Res50	-	48.1
Context-Contrasted (CCL) [11]	Res50	-	46.3
CiSS-Net (Ours)	Res50	76.5	48.7

Table 7. Segmentation results on Pascal Context. Our CiSS-Net achieves the state-of-the-art performance.† indicates the performance is reported in [11].

(42.56%/80.77% mIoU/pixACC) among the algorithms with the same backbone, ResNet-50. Moreover, our CiSS-Net with ResNet-50 also outperforms algorithms with much complicated backbones, e.g. ResNet-100.

Cityscapes Table 8 shows comparison results on the test set of Cityscapes. In this test, only the 5,000 finely annotated images in Cityscapes are involved in the training of CiSS-Net for fair comparison. Among all the compared algorithms, our CiSS-Net performs the best.

Pascal Context In Table 7, we evaluate the performance of our CiSS-Net on Pascal Context without utilizing additional data. This table shows that the CiSS-Net outperforms state-of-the-art methods with the same backbone and is comparable with those works with deeper backbone.

Method	$IoU_{cla.}$	$IoU_{cat.}$
SegNet [1]	57.0	79.1
CRF-RNN [38]	62.5	82.7
FCN [25]	65.3	85.7
DPN [24]	66.8	86.0
DilatedNet [34]	67.1	86.5
LRR [12]	69.7	88.2
DeepLab [5]	70.4	86.4
Context-CRF [22]	71.6	87.3
RefineNet [21]	73.6	87.9
FRRN [29]	71.8	88.9
GRN+LRN(single model) [36]	76.2	-
DSSPN(Universal) [20]	76.6	89.6
DepthAware [18]	78.2	89.7
PSPNet [37]	78.4	90.6
CiSS-Net (Ours)	**79.2**	**90.7**

Table 8. Segmentation results on Cityscapes. Our CiSS-Net achieves the best performance by using only the fully annotated images in training.

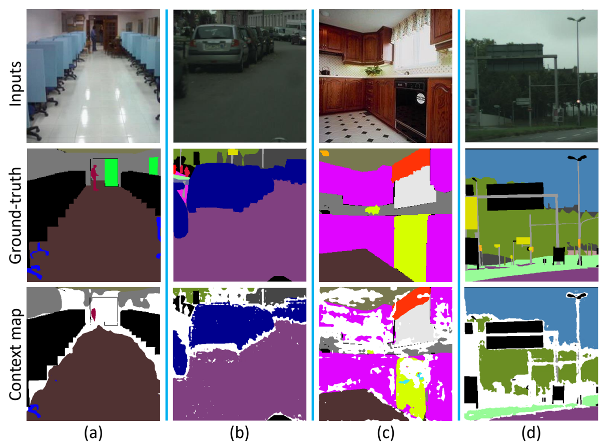

Figure 5. Visualization of the context maps generated at step 2. The white color indicates the uncertain regions generated by the C-Net. It can be observed that the context tends to progressively perceive regions/objects with high reliability while ignoring those with large ambiguity, e.g. the details and the far end of the street in the second and last columns, as context information.

3.6. Discussion on the Learned Context

First of all, the learned context map is not composed of high-confidence pixels of p-maps. The information in the learned context map is selected to have long-term benefits to the segmentation. Taking Fig. 1 as an example, the region containing the reflection of trees with a higher probability (0.993) is ignored while the water region along the left edge with a lower probability (0.870) is selected in step 1 by our C-Net. More examples can be found in Fig. 5, e.g. the right-most grasses (probability=0.929) rather than the left most board (0.976) in the 4^{th} image is selected as context.

Second, we observe that most of the contextual information provided by the context map is background objects and stuff, such as the floor, wall and cabinet in the (a) and (c) in Fig. 5, instead of regions/objects with large ambiguity, such as the details and end of the street in the (b) and (d) in Fig. 5. We believe that this kind of contextual information contains the overall layout of the scene that has rich semantic cues, constraints and even location priors of the current image. Thus it is very beneficial for the predictions of other items in the scene. It also echoes the first term of the reward function in Eq. (4) which encourages the pursuit of context that improves segmentation predictions.

Third, we notice that our RL-based context learning automatically provides a unique information, the 'uncertain class'. This information can be very helpful in identifying the 'hard' examples or high ambiguity regions in semantic segmentation, as illustrated by the white regions in Fig. 5. One the other hand, we also find that the uncertain regions contain lots of boundaries and small objects. It indicates that the learned context may lack enough support for these regions. The iIoU performance on Cityscapes that puts more weight on small objects also supports our observations, where the class-level and category-level iIoU of our methods are 55.6% and 78.0%, respectively, which are a little bit lower than the best ones 56.7%/78.6% (full table is provided in supplementary materials). Therefore, we will focus more on enhancing the performance of our CiSS-Net on boundaries and small objects, e.g. by introducing boundary refinement ideas and mining hard examples from the 'uncertain class' to the context learning, in future work.

4. Conclusion

In this paper, we propose using the p-maps as another source of the scene context in addition to the traditional contextual features. The context that has long-term benefits for the segmentation inference is selectively and adaptively extracted from p-maps via a dedicated module, Context Net, by reciprocally interacting with the segmentation network. By formulating the above process as MDP, we optimize the Context Net through reinforcement learning without introducing any extra supervision, and we further propose a fully end-to-end context-reinforced semantic segmentation network to efficiently facilitate such learning process. Numerical and visualization results demonstrate the benefits brought by the proposed context-reinforced scheme. In the future, we will make effort on enhancing the performance of our CiSS-Net for small object and explore the potential of the context-reinforced concept for other cognition tasks.

Acknowledgement

This work was supported by the National Key R&D Program of China under Grant 2017YFB1300201, the National Natural Science Foundation of China (NSFC) under Grants 61622211 and 61620106009 as well as the Fundamental Research Funds for the Central Universities under Grant WK2100100030.

References

[1] V. Badrinarayanan, A. Kendall, , and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[2] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. Semantic segmentation with second-order pooling. In *European Conference on Computer Vision*, pages 430–443. Springer, 2012.

[3] S. Chandra and I. Kokkinos. Fast, exact and multi-scale inference for semantic image segmentation with deep gaussian crfs. In *European Conference on Computer Vision*, pages 402–418. Springer, 2016.

[4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.

[5] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. 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.

[6] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[7] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. 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.

[8] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. 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.

[9] J. Dai, K. He, and J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1635–1643, 2015.

[10] J. Dai, K. He, and J. Sun. Convolutional feature masking for joint object and stuff segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3992–4000, 2015.

[11] H. Ding, X. Jiang, B. Shuai, A. Q. Liu, and G. Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2393–2402, 2018.

[12] G. Ghiasi and C. C. Fowlkes. Laplacian pyramid reconstruction and refinement for semantic segmentation. In *European Conference on Computer Vision*, pages 519–534. Springer, 2016.

[13] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 447–456, 2015.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] W.-C. Hung, Y.-H. Tsai, X. Shen, Z. L. Lin, K. Sunkavalli, X. Lu, and M.-H. Yang. Scene parsing with global context embedding. In *ICCV*, pages 2650–2658, 2017.

[16] M. A. Islam, M. Rochan, N. D. Bruce, and Y. Wang. Gated feedback refinement network for dense image labeling. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4877–4885. IEEE, 2017.

[17] X. Jin, Y. Chen, Z. Jie, J. Feng, and S. Yan. Multi-path feedback recurrent neural networks for scene parsing. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[18] S. Kong and C. Fowlkes. Recurrent scene parsing with perspective understanding in the loop. *arXiv preprint arXiv:1705.07238*, 2017.

[19] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.

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

[21] G. Lin, A. Milan, C. Shen, and I. D. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Cvpr*, volume 1, page 5, 2017.

[22] G. Lin, C. Shen, A. Van Den Hengel, and I. Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3194–3203, 2016.

[23] W. Liu, A. Rabinovich, and A. C. Berg. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015.

[24] Z. Liu, X. Li, P. Luo, C.-C. Loy, and X. Tang. Semantic image segmentation via deep parsing network. In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pages 1377–1385. IEEE, 2015.

[25] J. Long, E. Shelhamer, , and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[26] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937, 2016.

[27] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 891–898, 2014.

[28] P. H. Pinheiro and R. Collobert. Recurrent convolutional neural networks for scene labeling. Technical report, 2014.

[29] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe. Fullresolution residual networks for semantic segmentation in street scenes. *arXiv preprint*, 2017.

[30] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In

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

[31] A. G. Schwing and R. Urtasun. Fully connected deep structured networks. *arXiv preprint arXiv:1503.02351*, 2015.

[32] R. Vemulapalli, O. Tuzel, M.-Y. Liu, and R. Chellapa. Gaussian conditional random field network for semantic segmentation. In *CVPR*, pages 3224–3233, 2016.

[33] Y. Wu and K. He. Group normalization. *arXiv preprint arXiv:1803.08494*, 2018.

[34] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[35] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal. Context encoding for semantic segmentation. *arXiv preprint arXiv:1803.08904*, 2018.

[36] R. Zhang, S. Tang, M. Lin, J. Li, and S. Yan. Global-residual and local-boundary refinement networks for rectifying scene parsing predictions. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 3427–3433. AAAI Press, 2017.

[37] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890, 2017.

[38] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1529–1537, 2015.

[39] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ade20k dataset. *arXiv preprint arXiv:1608.05442*, 2016.