Paper Reading No.19

Context-Reinforced Semantic Segmentation

Sheng Lian

November 2019

1 Brief Paper Intro

- ***Paper ref:*** CVPR 2019, http://openaccess.thecvf.com/content_CVPR_2019/html/Zhou_Context-Reinforced_Semantic_Segmentation_CVPR_2019_paper.html
- ***Authors:*** Yizhou Zhou, Xiaoyan Sun, Zheng-Jun Zha, and Wenjun Zeng from USTC and MSRA.
- ***Paper summary:*** For the semantic segmentation task, this paper proposed to explore context information in the predicted segmentation map. They formulate the context learning problem as a Markov Decision Process and optimize it using reinforcement learning. Authors integrate the baseline segmentation model with the context net iteratively to refine the segmentation result.
- ***Reading motivation:*** This paper introduced a rather new pipeline which take full utilization of the predicted segmentation map (p-map).

The p-map contains rich high-level semantic cues and can be good candidate for context. Overall, this paper tries 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.

2 Methods

As aforementioned contents, this paper tries 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. Let's first see the visualized results, including p-map, GT, baseline and recursive refinement by the proposed CiSS-Net in Fig 1.

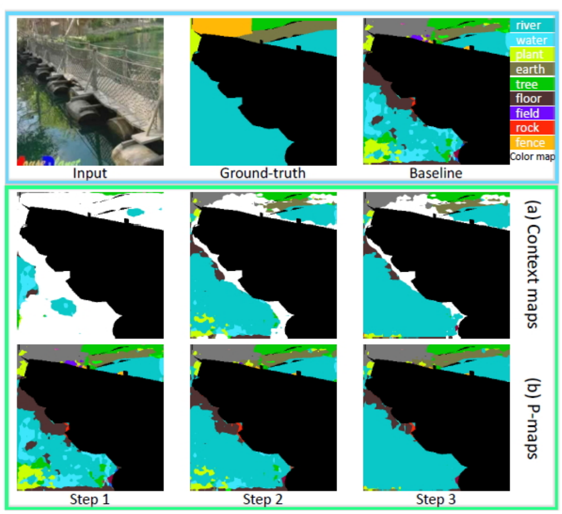

Figure 1: Visualized p-map and context maps in the proposed CiSSNet. The white areas denote uncertain regions.

The overall framework of the proposed CiSS-Net is shown in Fig 2. The

CiSS-Net has two sub-networks, Segment Net and Context Net. These two sub-networks mutually benefit each other and work iteratively. 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 the input image features to generate new context.

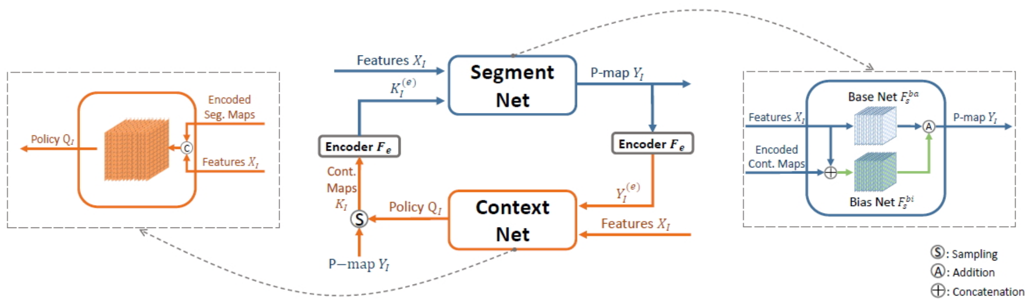

Figure 2: Overview of the proposed Context-reinforced Semantic Segmentation Network.

By properly defining the state, action and transition matrix, the context can be explicitly learned through reinforcement learning (A3C algorithm here) towards improving the segmentation performance without any extra supervision.

For better elaborating the following sub-models, we have some description:

- Context map: K_I
- Domain features: X_I
- P-map: Y_I
- Pyramid pooling Module F_e
- Base Net: F_s^{ba}, Bias Net: F_s^{bi}

- Context Net: F_k
- Policy Q_I

Segment Net The S-Net adopt a pre-trained convolutional neural network (CNN) to extract mid-level domain features X_i instead of using the raw image. The Pyramid Pooling Module is also employed to encode context map into multiple spatial levels, which reveal more global and local information at each spatial position. As is shown in the right side of Fig 2, and Fig 3, the S-Net contains two sub networks, and it can be formulated as

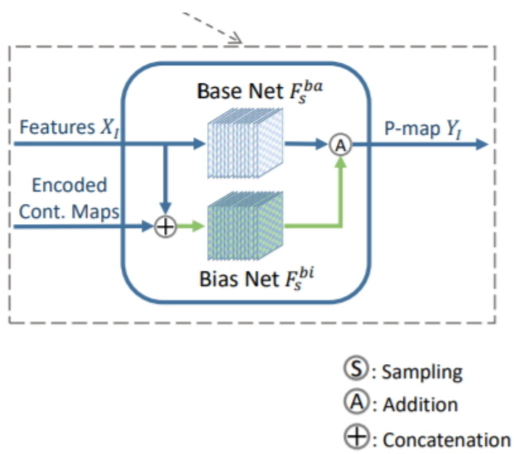

Figure 3: The brief architecture of segment net.

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

Context Net The architecture of C-Net is shown in Fig 4. 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 $Q_I = F_k\left(X_I \oplus Y_I^{(e)}\right)$, where the value of

Q(i, j, k) indicates the probability of taking action k at position (i, j). The action space is defined as 0，1.

- K=1: action of adopting the prediction,
- K=0: action of ignoring the prediction.

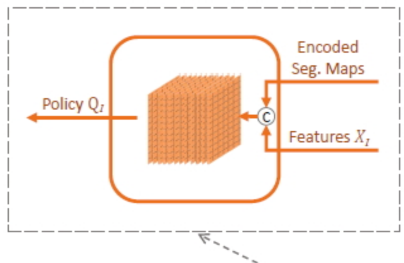

Figure 4: The brief architecture of context net.

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.

So, the overall formulation goes as

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

The author defines the context learning problem as the Markov decision process and proposes to learn the context through the interaction between CNet and SNet. This optimization process can be solved by deep reinforcement learning, considering p-map as the environment and CNet as the agent. In the intensive learning process, p-map can improve performance step by step.

Here, we can regard Y_I and K_I as a state-action pair and model it as a Markov decision process (MDP) with the tuple $(\mathbb{S}, \mathbb{A}, P, r, \rho_0, \gamma)$, the reward on spatial location (i, j) at time step t as

$$
\begin{array}{l}
\frac{1}{C_h C_w} \sum_{i',j'} M\left(Y_I^t\left(i', j'\right), Y_I^{t+1}\left(i', j'\right), L_I\left(i', j'\right)\right) \\
+\beta_1 \nVdash_{L_I(i,j)}\left(K_I^t(i,j)\right) + \beta_2 \nVdash_0\left(K_I^t(i,j)\right)
\end{array}
\tag{3}
$$

So the paper treat the extraction context as an action, consider the image and the segmentation p-map of the previous iteration as the environment, construct a Markov decision process. By maximizing the accuracy of future segmentation, the model guides the network selects contextual information that has the largest long benefit.

The reward's first part is defined as follows:

> respectively. There are four different cases, *i.e.* $Y_I^t(i', j')$ is correct/incorrect $\rightarrow Y_I^{t+1}(i', j')$ is correct/incorrect. We assign a reward 1 to the case 'incorrect $\rightarrow$ correct', -1 to the case 'correct $\rightarrow$ incorrect', 0 to the case 'incorrect $\rightarrow$ incorrect' and 0.5 to the case 'correct $\rightarrow$ correct'. Because

Figure 5: reward definition.

Here, authors proposed to use asynchronous advantage actor-critic algorithm (A3C) [1] to optimize the MDP problem.

The overall loss can be formulated as

$$
\text{Loss } = \text{Loss}_p + \text{Loss}_v + \lambda_1 \text{Los}\, s_s + \lambda_2 Loss_e \tag{4}
$$

> 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.

References

[1] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937, 2016.