Refined Pixel-labeling Process for Weakly Supervised Semantic Segmentation

Zhengyang LYU Pierre BEAUSEROY Alexandre BAUSSARD

LIST3N, Université de Technologie de Troyes, 10004 Troyes, France

Résumé – La segmentation sémantique consiste à associer une étiquette ou une catégorie à chaque pixel d'une image. Afin d'éviter l'annotation de l'ensemble des pixels des images des bases d'apprentissage, qui est très coûteuse, une alternative consiste à utiliser des annotations dites faibles comme les catégories présentes dans les images. On parle alors de segmentation sémantique faiblement supervisée. Ces approches font généralement appel à des cartes d'activation par classe (CAM). Cependant, ces CAM ne se concentrent que sur les régions discriminantes de l'image, ce qui limite leur qualité. Des méthodes de post-traitement sont généralement proposées pour améliorer les masques générés, utilisés dans l'algorithme de segmentation. En effet, ceux-ci conditionnent fortement le résultat final de segmentation. Ces traitements sont essentiellement basés sur des informations de faible résolution, obtenues en sortie du réseau de neurones convolutionnels. Nous proposons ici d'exploiter des sorties de couches en début du réseau afin d'intégrer des informations de plus haute résolution via un algorithme de type machine à vecteurs de support (SVM). Les expérimentations montrent une amélioration par rapport aux approches de la littérature.

Abstract – Semantic segmentation aims to assign each pixel in an image to a semantic category. To avoid expensive annotating of all pixels in training images, weakly supervised semantic segmentation with weak annotations such as the image-level labels is used as an alternative approach. It is a common practice to utilize Class Activation Map (CAM). However, CAM only focuses on the discriminative regions which limits its quality. Post-processing methods are generally proposed to improve the generated masks, which greatly improve the final segmentation results. However, most post-processing methods rely on low resolution features from the outputs of the convolutional neural networks. In this work, we propose a method that leverages early layer outputs to integrate high-resolution features via a support vector machine (SVM) algorithm. Experiments demonstrate improvement compared with state-of-the-art methods.

1 Introduction

Semantic segmentation is a popular task in computer vision which requires labor-intensive pixel-level manual annotation. Compared to classification labels, giving segmentation annotation is considerably more time-consuming [3]. To reduce the annotation burden, weakly supervised semantic segmentation (WSSS) approaches have been proposed using weaker supervision [1, 7, 8, 10]. In this paper, we focus on weakly supervised semantic segmentation with only image-level labels due to it is the easiest and cheapest available annotation.

Almost all the latest WSSS algorithms with image-level labels require a two-stage pipeline. To start with, class activation map (CAM) [11] is generated from a trained classification network. Based on CAM, pseudo mask is obtained by a refinement process. Secondly, pseudo mask is used to train a fully supervised semantic segmentation model. In order to get the high quality pseudo mask, recent methods focus on improving the performance of CAM, or finding efficient refinement process to fully utilized the given CAM.

However, there are two limitations for recent refinement methods. Firstly, most of them focus on the prediction from CAM and high-semantic deep features with low resolution. Generally, the resolution of CAM and deep feature is 1/8 or 1/16 compared with the initial size of the image. A simple interpolation resize process can not recover the details especially among the boundary of objects. Secondly, the refinements show no partiality and miss the chance to analyze disparity in prediction of CAM for different images in the dataset.

Therefore, we introduce a refined pixel-labeling process that uses not only deep semantic feature to generate CAM and initial seed, but also shallow features with high resolution to obtain better prediction with details. Different with methods which are handled without consideration for differences among images, our method trains an image-specific classifier to make pixel-wise segmentation.

The proposed method have three steps. Firstly, we generate CAM and initial seed from a trained classification network. Then, high precision reliable data is obtained. Specifically, we select train sample from the confident regions and exclude noisy false-positive samples by using an image-specific background prototype. Finally, a self-enhanced SVM is trained to make the final segmentation. The train label for the self-enhanced SVM is allowed to be updated. In this case, uncertainty is decreased and mistakes made by the initial errors are corrected as the training goes on. As a common practice, dCRF [6] is used as post-processing method to improve the prediction. The pipeline of our method is illustrated in Figure 1.

The rest of the paper is organized as follows: After giving detailed explanation about our method in the Section 2, the experiments setting and sufficient results for the method are presented in the Section 3. Finally, we draw a conclusion and plan the future work in the Section 4.

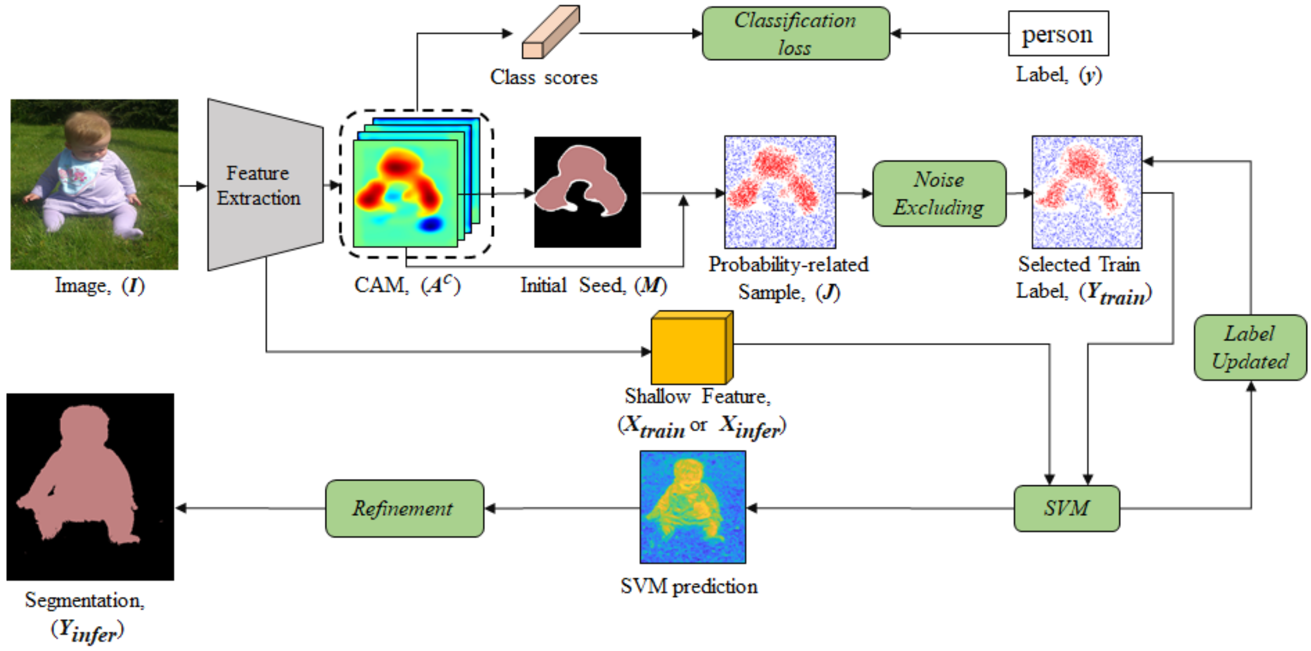

Figure 1 – Pipeline of the proposed method.

2 Proposed Method

This section describes the proposed method, as shown in Figure 1. Firstly, we explained the process to obtain CAM and initial seed from a trained classification network. Then, pixel-wise train sample is generated. In order to collect data with high precision labels, we use probability-related selection to sample data with high confidence, and image-specific background prototype to exclude noisy foreground. Finally, the self-enhanced SVM is trained to predict pixel-wise labels for each image.

2.1 CAM and Initial Seed

Firstly, followed by the steps in the multi-stage WSSS pipeline based on CAM [11], we start to train a classification Convolutional Neural Network (CNN) with only image-level labels. $\boldsymbol{I}$ is defined as the input image with image-level label $\boldsymbol{y}$. The class vector of the dataset is $\boldsymbol{C}$ and the output feature $\boldsymbol{F}$ of the last convolutional layer has $|\boldsymbol{C}|$ channels. Then a global average pooling layer pools feature $\boldsymbol{F}$ to a vector $\boldsymbol{f}$ of size $|\boldsymbol{C}|$. We calculate the classification loss by using a multi-label soft margin loss function, which is formulated as follows:

$$L_{cls} = -\frac{1}{|\boldsymbol{C}|}\sum_{c=1}^{|\boldsymbol{C}|} \boldsymbol{y}^c \log\left(\sigma(\boldsymbol{f}^c)\right) + (1-\boldsymbol{y}^c)\log\left(1-\ \sigma(\boldsymbol{f}^c)\right) \tag{1}$$

where σ is the sigmoid function.

The class activation map $\boldsymbol{A}^c$ for class c is obtained by normalizing the class-specific feature $\boldsymbol{F}^c$ from last convolution layers, as follows:

$$\boldsymbol{A}^c = \frac{\mathrm{ReLU}(\boldsymbol{F}^c)}{\max(\mathrm{ReLU}(\boldsymbol{F}^c))} \tag{2}$$

Then, we directly utilize the CAMs of given classes c' to generate the initial seed $\boldsymbol{M}$ by thresholding their scores with T_{fg} and T_{bg}, for foreground and background respectively, as follows:

$$\boldsymbol{M}_i = \begin{cases} 0, & \text{if } \max\limits_{c'}(\boldsymbol{A}_i^{c'}) < T_{bg} \\ \arg\max\limits_{c'}(\boldsymbol{A}_i^{c'}), & \text{if } \max\limits_{c'}(\boldsymbol{A}_i^{c'}) > T_{fg} \\ -1, & \text{otherwise} \end{cases} \tag{3}$$

$\boldsymbol{M}_i$ is the pixel-wise prediction from CAM for pixel i in the image $\boldsymbol{I}$. $\boldsymbol{M}_i = -1$ is regarded as uncertain regions which exist in the boundary between object and background generally. c' is the given labels that $c' \in \boldsymbol{y}$.

2.2 Data Preparation

The goal of data preparation process is to collect data with high precision labels.

Obtained from CAM, the initial seed $\boldsymbol{M}$ is too noisy to be used as the segmentation result. As extensively elaborated, CAM only focuses on the discriminative regions and does not cover all the object, which causes low precision of foreground in $\boldsymbol{M}$. A two-step process is proposed. Our intuition is to select data as reliable as possible with only in-hand information given from the CAM $\boldsymbol{A}^c$ and the initial seed $\boldsymbol{M}$.

(1) Probability-related selection.

We define p_i^c as the probability to select data from pixel i as a sample for class c:

$$p_i^c = \frac{\boldsymbol{P}_i^c \mathbb{1}(\boldsymbol{M}_i = c)}{\sum_{n\in\boldsymbol{I}} \boldsymbol{P}_n^c * \mathbb{1}(\boldsymbol{M}_n = c)} \tag{4}$$

$\mathbb{1}(\cdot)$ outputs 1 if the argument is true or 0 otherwise,

$$\boldsymbol{P}_i^c = \frac{\max(\boldsymbol{A}_i^c - T_{fg}, 0)}{1 - T_{fg}} \tag{5}$$

In experiments, we found that the background from $\boldsymbol{M}$ gives high precision, which means most of the background labels are reliable. As a result, we select background data uniformly and randomly from the pixels $\{i|\boldsymbol{M}_i = 0\}$, specifically,

$$p_i^0 = \frac{\mathbb{1}(\boldsymbol{M}_i = 0)}{\sum_{n \in I} \mathbb{1}(\boldsymbol{M}_n = 0)} \tag{6}$$

p_i^0 is the probability to select pixel i as background sample.

For each pixel k in the image, if it is not in the uncertain regions, it will be selected at most one time. Based on the rules above, a subset of pixels $\boldsymbol{J}$ is drawn from the image $\boldsymbol{I}$, i.e. $\boldsymbol{J} \subseteq \boldsymbol{I}$.

(2) Excluding noisy foreground.

The high precision of background in CAM is suitable to model an image-specific background prototype. Since background has limited specific semantic, compared with deep feature, it is sufficient to explore background in low-level visual information. Therefore, our background prototype $\boldsymbol{S_{bg}}$ is computed by shallow feature and defined as follows:

$$\boldsymbol{S_{bg}} = \frac{\sum_{j \in \boldsymbol{J}} \boldsymbol{S}_j * \mathbb{1}(\boldsymbol{M}_j = 0)}{\sum_{j \in \boldsymbol{J}} \mathbb{1}(\boldsymbol{M}_j = 0)} \tag{7}$$

$\boldsymbol{S}_j$ is the shallow feature generated from the trained classification network in pixel j.

We assume that there are false positive predictions in foreground class which share similar feature with background. In this case, such noise is excluded using background prototype. Cos-similarity is calculated between the background prototype $\boldsymbol{S}_{bg}$ and feature $\boldsymbol{S}_{j^+}$ in foreground pixel j^+ of $\boldsymbol{J}$, i.e. $\boldsymbol{M}_{j^+} > 0$.

$$\boldsymbol{w}_{j^+} = \frac{\boldsymbol{S_{bg}}^\top \cdot \boldsymbol{S}_{j^+}}{||\boldsymbol{S_{bg}}^\top|| ||\boldsymbol{S}_{j^+}||} \tag{8}$$

$\boldsymbol{S_{bg}}^\top$ is the transpose of $\boldsymbol{S_{bg}}$. If $\boldsymbol{w}_{j^+} > T_s$, foreground pixel j^+ shares too much similarity with background prototype, which might be a false positive in prediction and should be excluded.

Finally, the pixel set $\boldsymbol{K}$ is generated, which is a subset of $\boldsymbol{J}$, i.e. $\boldsymbol{K} \subseteq \boldsymbol{J}$. For the pixel-wise SVM classifier, the train label, referenced as $\boldsymbol{Y_{train}}$, is composed by $\boldsymbol{M}_k$ in spatial index of each selected pixel $k \in \boldsymbol{K}$. We unleash the potential of shallow feature not only to detect noisy part, but also to train SVM. Therefore, the train data, referenced as $\boldsymbol{X_{train}}$, is collected from all the shallow feature $\boldsymbol{S}_k$ in the same spatial index of pixel set $\boldsymbol{K}$ correspondingly.

2.3 Self-enhanced SVM Training

After data preparation process, we obtain the image-specific train set $\{\boldsymbol{X_{train}},\ \boldsymbol{Y_{train}}\}$ and use it to train a pixel-wise SVM classifier for each image. The image-specific infer data, referenced as $\boldsymbol{X_{infer}}$, is composed by shallow feature $\boldsymbol{S}_i$ in spatial index of all pixel i in the image $\boldsymbol{I}$. Taking $\{\boldsymbol{X_{infer}}\}$ as input, the trained SVM is used to give labels $\{\boldsymbol{Y_{infer}}\}$ for each pixel, and finish the segmentation task for the image.

It is obvious that the performance of prediction from SVM is related to the train label quality. However, even after the designed data preparation process, the train label $\boldsymbol{Y_{train}}$ is still not perfect, which brings false prediction. In experiments, we found that SVM is robust to noise input and able to correct false train labels by itself. Therefore, we propose a self-enhanced SVM training procedure.

Specially, when a SVM is trained, $\boldsymbol{X_{train}}$ is used as input again to update train label $\boldsymbol{Y_{train}}$ into $\boldsymbol{Y'_{train}}$ by its output. Then, the new train set $\{\boldsymbol{X_{train}},\ \boldsymbol{Y'_{train}}\}$ is used to train a new SVM. Such process is repeated after t times and the output of the final SVM is used as the segmentation results.

Enhanced by the robustness of SVM, the train label's quality is improved as the training goes on. Uncertainty is decreased and mistakes made by initial errors are corrected for the output of SVM.

3 Experiments

In this section, we give the details for experimental settings like dataset, evaluation metrics and implementation details at first. Secondly, we compare our results quantitative and qualitative with recent methods using the PASCAL VOC 2012 train set.

3.1 Experimental Settings

Dataset and Metrics. Our approach is evaluated on PASCAL VOC 2012 dataset [3], which has 20 foreground classes and 1 background class. The official dataset is split into train set, validation set and test set, which contains 1464, 1449 and 1456 images. For now, we evaluated our segmentation results on the train set. Following most previous work, the classification network is firstly pre-trained on ImageNet [2], then further trained on an augmented train set [4] of PASCAL VOC 2012 dataset. The augmented set contains 10582 images.

We compare results obtained from the proposed method with the published results from the other recent methods [1, 7, 8, 10]. As a common evaluation metric, we use mean intersection over union (mIoU) to evaluate segmentation results.

Implementation Details. In our experiments, we use SIPE [1] as the baseline method to obtain CAM and initial seed. As implemented by SIPE, ResNet-50 [5] is adopted as the backbone network. In order to improve the quality of initial seeds, CAM is generated by using multi-scale images as inputs. The scale ratios are $\{0.5, 1.0, 1.5, 2.0\}$. We select feature from the second stage of the backbone network as the shallow feature S since it is in high resolution and has appropriate semantic information. The spatial size of selected feature is 1/4 of initial image size. In order to improve the divisibility of features, shallow feature is z-score normalized. After the prediction of SVM, dCRF is used as post-processing to refine the generated localization maps. For each class including background, we select 2500 samples for the sample set $\boldsymbol{J}$. We empirically set T_{fg} = 0.2 and T_{bg} = 0.05 to obtain initial seed, and set similarity threshold T_s = 0.2 for excluding noisy in foreground. In our experiments, self-improved SVM cyclic is repeated 5 times.

3.2 Results

Table 1 shows the quantitative comparison between our method with the baseline and other state-of-the-art methods. Compared with the baseline method [1], our method improve the mIoU by 2.65% on initial seed and 1.47% after dCRF process. Besides, our method exceeds other state-of-the-art

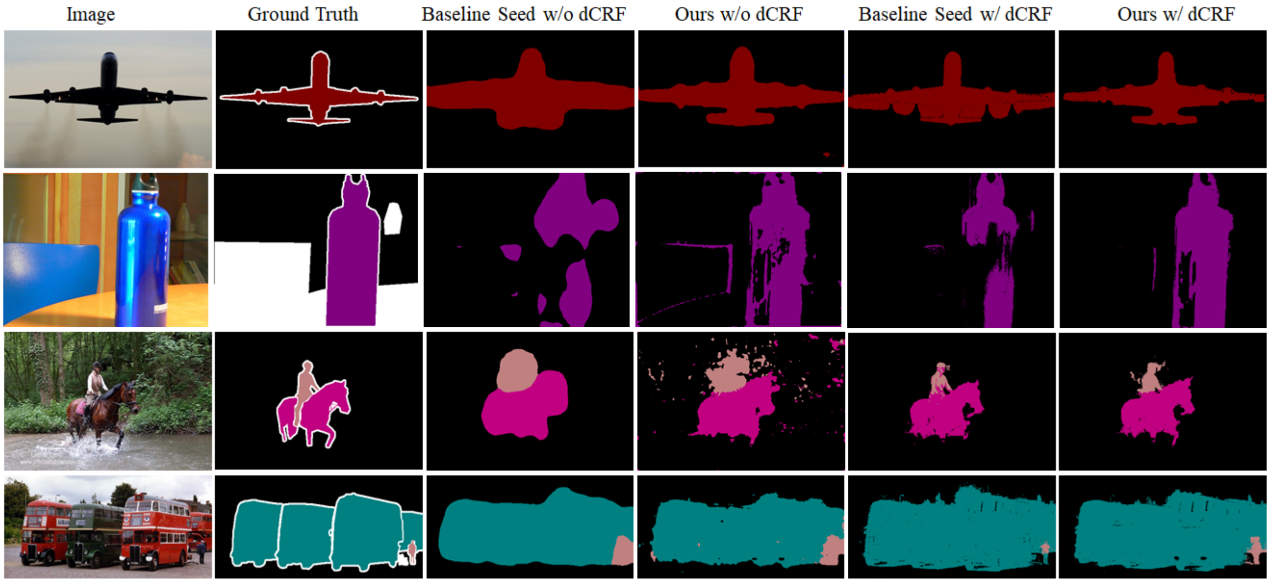

Figure 2 – Qualitative segmentation results on VOC 2012 train.

methods. These results suggest that, our approach is able to effectively leverage shallow feature to recover details for the objects, and predictions of proposed method are more accurate to match the ground truth segmentation masks.

Figure 2 presents qualitative results from the baseline method and our method without (w/o) and with (w/) dCRF process. Generally, compared with baseline method, our method shows better performance in recovering details and distinguish background and objects. The dCRF refinement process is able to remove noise produced by isolated pixels, since dCRF takes the surrounded pixels' prediction into consideration.

Table 1 – mIoU (%) of seeds and refined maps on PASCAL VOC 2012 train set. The best results are shown in bold.

Method	Backbone	Seed	+ dCRF
VWL-L [8]	ResNet-101	57.3	63.0
MCL [10]	EfficientNet [9]	58.4	64.6
SIPE [1]	ResNet-50	58.6	64.7
Iter_dCRF [7]	ResNet-50	60.6	62.7
Ours	ResNet-50	**61.4**	**66.2**

4 Conclusion

Using only image-level label, we propose a refined pixel-labeling process for weakly supervised semantic segmentation method. In the proposed method, we generate CAM from an image classification CNN and efficiently sample reliable data from the given CAM by probability-related selection and exclusion of ambiguous regions. In addition, a self-enhanced image-specific cyclic SVM trained with shallow feature is used to output pixel-wise prediction and improve the segmentation. Our experiments demonstrate that shallow feature with high resolution is effective in improving details of segmentation, and image-specific pixel-wise classifier is beneficial for WSSS. In the future work, we will use multi-level feature to improve our approach and conduct more experiments on the larger data set likes MS COCO 2014.

References

[1] Q. Chen, L. Yang, and J. Lai et al. Self-supervised image-specific prototype exploration for weakly supervised semantic segmentation. In *CVPR*, 2022.

[2] J. Deng, W. Dong, and R. Socher et al. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[3] M. Everingham, S.M. Eslami, and L. Van Gool et al. The pascal visual object classes challenge: a retrospective. *IJCV*, 2015.

[4] B. Hariharan, P. Arbeláez, and L. Bourdev et al. Semantic contours from inverse detectors. In *ICCV*, 2011.

[5] K. He, X. Zhang, and S. Ren et al. Deep residual learning for image recognition. In *CVPR*, 2016.

[6] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*, 2011.

[7] Y. Li, J. Sun, and Y. Li. Weakly-supervised semantic segmentation network with iterative dcrf. *T-ITS*, 2022.

[8] L. Ru, Y. Zhan, and B. Yu et al. Learning affinity from attention: End-to-end weakly-supervised semantic segmentation with transformers. In *CVPR*, 2022.

[9] M. Tan and Q. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019.

[10] K. Yuan, G. SChaefer, and y. Lai et al. A multi-strategy contrastive learning framework for weakly supervised semantic segmentation. *PR*, 2023.

[11] B. Zhou, A. Khosla, and A. Lapedriza et al. Learning deep features for discriminative localization. In *ICCV*, 2016.