Weakly-Supervised Semantic Segmentation via Self-training

Hao Cheng[1,2], Chaochen Gu[1,2,*] and Kaijie Wu[1,2]

[1]Department of Automation, Shanghai Jiao Tong University, Shanghai, China.
[2]Key Laboratory of System Control and Information Processing, Ministry of China Education.

Corresponding Author: *jacygu@sjtu.edu.cn

Abstract. Weakly-supervised semantic segmentation with image tags is a challenging computer vision task. Unlike pixel-level masks, image tags give high level semantic information, without low level appearance information. In this paper, we propose an iteratively self-training framework to bridge this two information, which expand and refine the pseudo-labels with training process going. Initial masks are generated from classification network. In the top-down step, rendered images and its labels as well as spatially weight loss are added to jointly training the model for alleviate the effect of inaccurate object masks. Then in the bottom-up step, an adaptive threshold to the confidence model predictions to keep predicted masks reliable. The top-down and bottom-up steps are conducted iteratively to extract the fine object mask. Experiments on our self-build dataset and GTA5 to CityScapes demonstrate the effectiveness of proposed framework.

1. Introduction

Semantic segmentation with image tags only aims at performing pixel-wise classification with only image tags provided. Semantic segmentation models require both high-level semantic and low-level appearance information during training. What makes weakly supervised semantic segmentation challenging is that images tags could only provide high level semantic information. With images tags, classification network is used to generate localization. With no other annotation tools assisted, localization regions from classification network are coarse and inaccurate, which is far away from the requirement of segmentation models and even harm the performance.

With the issues addressed, we take a different view to this task. We propose a self-training framework where the model uses previous predictions to update its parameters. The self-training framework has been used in semi-supervised learning for a better classifier [1]. However, most of these methods still depend on well labelled data. The subtle difference between the previous work and our proposed framework is that there is no pixel-wise label to measure the prediction during training.

Our proposed self-training framework contains bottom-up and top-down two steps, which could tolerate inaccurate predictions with the refinement operation. Our motivation comes from the classification network could give coarse localizations [2], as well as certain discriminative regions, and previous PixelNet [3] has shown that fewer pixels for computing the loss could give the same result as full pixels used. Given a set of training images, the latent pixel-wise label could be extracted for the target object. In the top-down step, we take the initial mask with the assistance of synthetic images and the corresponding labels, and spatially weighted loss is introduced to alleviate effect of inaccurate pseudo-labels. Then in the bottom-up step, the predictions get refined according to the confidence of predictions and the information from the image itself. Incorrect mask could be refined with the iteration continues.

Concretely, a classification network is firstly trained to get the Classification Activation Maps (CAM) [2] and localize the discriminative regions. To best use the information from the given image itself, the image is over-segmented to super-pixel map to remove some of wrongly labelled pixels. In the top-down step, joint training with synthetical images and pseudo-labels to predict the new mask of object. As pseudo labels may not accurate, spatial-weight loss is introduced to alleviate the effect of wrongly labelled pixels. In the bottom-up step, all confidences of all model predictions are sorted to remove possible wrong labels, and then the super-pixel map and the CRF layer [4] are utilized to refine the pseudo-labels as well as the precise boundaries. With the mentioned procedure, the latent label gets refined iteratively with distinct boundaries.

Our contributions in this paper can be summarized as:

We proposed an iteratively self-training framework towards the weakly supervised semantic segmentation task, which could extract labels from coarse to fine. Specially, experiments show the pseudo-labels could be extracted from images themselves.

To address the issue the inaccuracy of pseudo-labels, we proposed the spatial-weighted loss to soft the pseudo-labels.

2. Related Work

In this section, we make a brief introduction on the recent progress on both fully and weakly-supervised semantic segmentation methods which are related to our work.

2.1. Fully Supervised Semantic Segmentation

Current predominant fully supervised methods usually train models end-to-end with entire images and the corresponding pixel-wise labels incorporated. Fully convolutional network (FCN) [5] uses skip connections for pixel-wise predictions. DeepLab [6] uses convolution layer with dilation to expand feature map size as well as the receptive field. PSPNet [7] tries capture the contextual information with pooling operations at different scales. A large number of works [8]-[9] have been proposed based on similar resumption.

2.2. Weakly-Supervised Semantic Segmentation

Annotating images for semantic segmentation task acquires large time and labour. Recent researches exploited weakly supervised methods to reduce the annotation cost, including bounding box, line, and image tags. For weakly supervised sematic segmentation task with image tags, most methods start from classification networks.

Classification networks is always utilized to get initial localization and train segmentation networks supervised with them. Kolesnikov et al. [10] introduced expand loss and constrain loss to expand the initial mask seeds with a boundary awareness. Wei Y et al.[11] considered classification network with different dilated convolution to get more concrete localization. Wang, Xiang et al. [12] train an extra patch classification network to mine common object feature for refining initial masks. Most of previous works rely on classification network generating discriminative regions sequentially. In this work, the proposed framework could tolerant the inaccurate predictions, and masks can be extracted with clear boundary.

3. Proposed Framework

Classification networks could only produce a heatmap with coarse localization information, and domain gap between the virtual and reality makes synthetically trained model drop heavily. Our proposed framework tends to combine both two information to extract fine segmentation masks during iteratively training.

Our proposed framework contains two steps, as shown in Figure 1, bottom-to-up mask refinement step and top-to-down joint training step. At the bottom-to-up step, the class activation map (CAM) and the super pixel map by SLIC [13] are utilized to refine the model predictions. And at the top-to-down joint-training step, the pseudo-masks get refined at bottom-to-up step are used to learn robust representations. As training continues, pseudo-masks get more and more accurate. The overall procedure is described as Algorithm 1. For inference, the segmentation model of last iteration is used, with only RGB image needed.

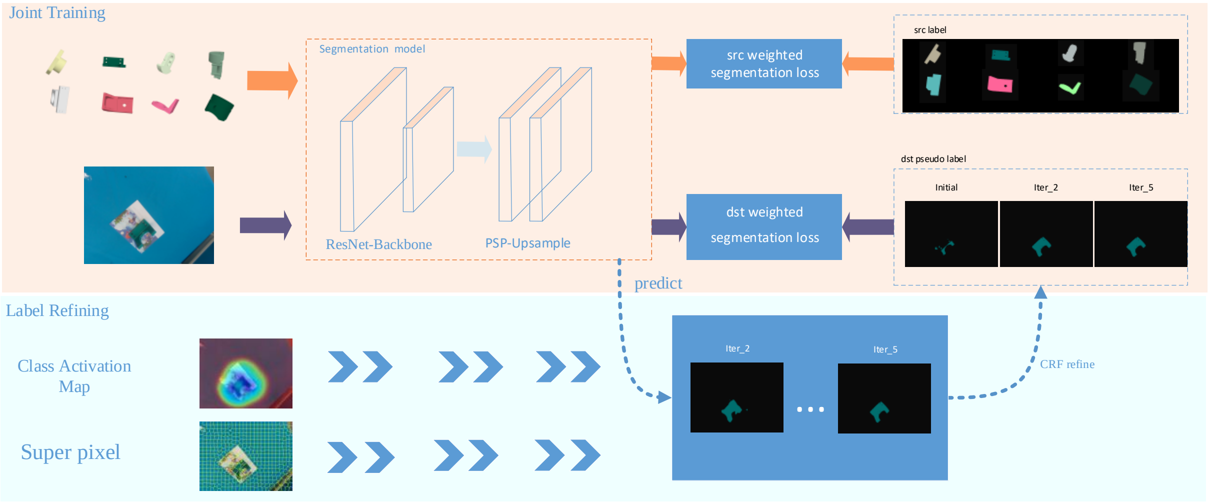
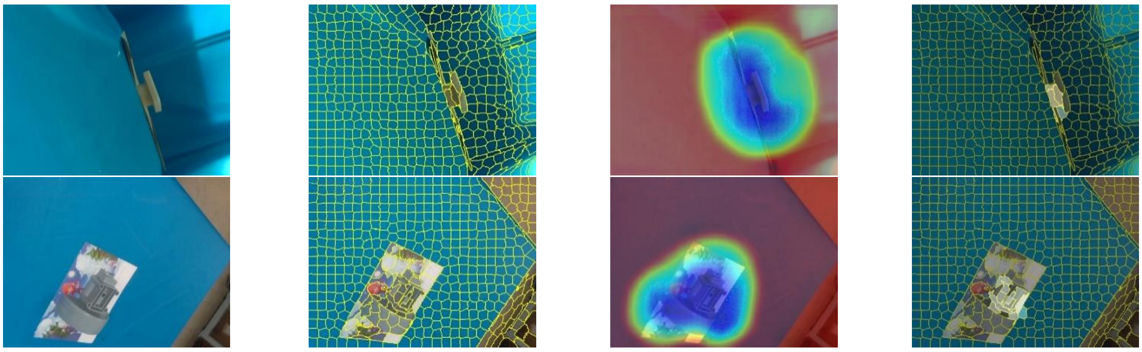


Figure 1. Pipeline of Proposed Network. At beginning, initial masks are generated from CAM and super-pixel maps. Then the joint training with rendered images using initial masks with spatially weighted loss to alleviate the effect of inaccurate masks. Predicted masks get refined as described in Section 3.4.

3.1. Initial Object Mask Generation

For images in Figure 2(a), a classification network is trained to obtain the heat map, using CAM method, for each object. As shown in Figure 2(c), localization information could be gotten from the heat map, while it is too coarse to be used for training. To get the initial object mask, images are over-segmented to super-pixel regions via SLIC algorithm[13], as shown in Figure 2(b). Then masks are firstly generated combined with the heat map and an adaptive threshold to get enough information while with less inaccurate information introduced in. As super-pixel regions could give boundary of the object, only super-pixel regions that are fully labelled with the mask from CAM are labelled as positive regions.



(a) images (b) super-pixel regions by SLIC (c) heat map of CAM (d) refined with super-pixel map described in Section 3.1

Figure 2. Examples of Initial Generated Labels.

Algorithm 1 Procedure of Proposed Framework

```
Input: Images $\{\mathcal{I}_S, \mathcal{I}_T\}$; Labels $\{\mathcal{L}_S\}$; Super-pixels $\{S_T\}$;
Output: Refined Object Mask $\{\mathcal{L}_T\}$; The Jointly trained Semantic Segmentation Model;
1: Initialize: Generate initial object mask $\{\mathcal{L}_{T_0}\}$ at iteration $\{t = 0\}$.
2: while extracted mask labels $\{\mathcal{L}_{T_t}\}$ is not satisfying do
3:     Joint training described in Section 2.3
4:     Predict the object mask $\mathcal{P}_{T_t}$
5:     Refine Predictions object masks $\mathcal{M}$ described in Section 2.4
6:     Update $\mathcal{L}_{T_t} = \mathcal{M}, t = t + 1$
7: end while
```

3.2. Spatially weighted loss

For generated pseudo labels, because not all pixels are rightly labelled, directly trained with these pseudo-labels may hurt the performance. As shown in Figure 2(d), for each masked pixel, closer to the center of pseudo-label mask, more possible be rightly labelled. Based on this observation, high probability should get more attention. So, we proposed the spatially weighted loss to make the loss function pay more attention to rightly labelled pixels and less sensitive to some labelled pixels which might not be classified rightly. The loss function could be formulated as

$$\begin{gathered}\min_{\mathbf{w}} \mathcal{L}_W(\mathbf{w}) = -\sum_{t=1}^{T}\sum_{n=1}^{N} a_n \hat{\mathbf{y}}_{n,t}^{\top} \log(\mathbf{f}_n(\mathbf{w},\mathbf{I}_t\})) \\ \text{s.t. } \hat{\mathbf{y}}_n \in \{\mathbf{e}^{(i)} | \mathbf{e}^{(i)} \in \mathcal{R}^C\}, \forall n\end{gathered} \tag{1}$$

where $\mathbf{I}_t$ denotes the image $(t = 1, 2, 3, ..., T)$, $\hat{\mathbf{y}}_{n,t}$ denotes n_{th} pixel label $(n = 1, 2, 3, ..., N)$, $\mathbf{w}$ refers to the parameters of the network. $\mathbf{f}_n(\mathbf{w}, \mathbf{I}_t)$ is the predicted probabilities at pixel n, and α_n is spatial weight for image $\mathbf{I}_t$ at pixel n. Specially $\mathbf{e}^{(i)}$ indicates a one-hot vector whose i_{th} entry is 1, and if $\mathbf{e}^{(i)}$ is a zero, it is equivalent to the *ignore_index*.

3.3. Joint Training Procedure

To provide the external shape awareness, the boundaries of labels from rendered images are used as the spatial weighted matrix. For implementation details, canny method is used to get the boundary, then the dilate operation is taken to expand the detected boundaries. For target images, distance transformation is used as the confidence of the pseudo labels, which is used as the spatial weight, then sigmoid function is utilized to map the values into $[0, 1]$. Total loss function is

$$\begin{aligned}\mathcal{L}_{\text{total}} &= \mathcal{L}_{W_S}(\mathcal{I}_S, \mathcal{L}_S, \mathcal{W}_S) + \mathcal{L}_{W_T}\left(\mathcal{I}_T, \hat{\mathcal{L}}_T, \mathcal{W}_T\right) \\ &= -\sum_{s=1}^{S}\sum_{n=1}^{N} a_{s,n} \mathbf{y}_{s,n}^{\top} \log(\mathbf{f}_n(\mathbf{w}, \mathbf{I}_s)) \\ &\quad -\sum_{t=1}^{T}\sum_{n=1}^{N} a_{t,n} \hat{\mathbf{y}}_{t,n}^{\top} \log(\mathbf{f}_n(\mathbf{w}, \mathbf{I}_t)) \\ &\quad \text{s.t. } \hat{\mathbf{y}}_{t,n} \in \{\mathbf{e}^{(i)} | \mathbf{e}^{(i)} \in \mathcal{R}^C\}, \forall t, n\end{aligned} \tag{2}$$

3.4. Label Refine Step

Simply iteration with raw predictions may make the model go unexpected for the wrong predictions exists. It's necessary to refine the predictions via removing the undesired predictions. The variance of the output probabilities is getting larger with the training procedure continuing, and the possible wrong labelled is likely to have lower probabilities. With this assumption, during the label refine step, the predicted pixels whose probabilities are lower than the adaptive threshold are labelled as negative. Details are described in Algorithm 2:

Algorithm 2 Label Refine

```
Input: Images { $\mathcal{I}_T$ }, ratio $p$
Output: Refined Object Mask { $\mathcal{L}_T$ }
for $i = 1$ to $T$ do
    $P_i$ = model ($I_{Ti}$)
    $S_i$ = max($P_i$, axis = 0)
    $S = [S, S_i]$
end for
$S$ = To_vector($S$)
$S$ = sort($S$, descend = True)
idx = length($S$) $*$ $p$
threshold = S[idx]
for $i = 1$ to $T$ do
    $P_i$ = model ($I_{Ti}$)
    $L_{Ti}$ = argmax($P_i$)
    $L_{Ti}$ < threshold = unlabelled
end for
```

4. Experiments

4.1. Dataset

4.1.1. Workpiece Dataset. The proposed framework is trained and evaluated on our self-built dataset composed of images of 8 different texture-less workpieces with the size 480 × 640. These images are collected from different viewpoints, with synthetical images rendered from CAD model, and real images captured from unstructured environments, including illumination variation, 4 different backgrounds and blurry imaging as shown in Figure 3. For each background, about 8000 images are collected.

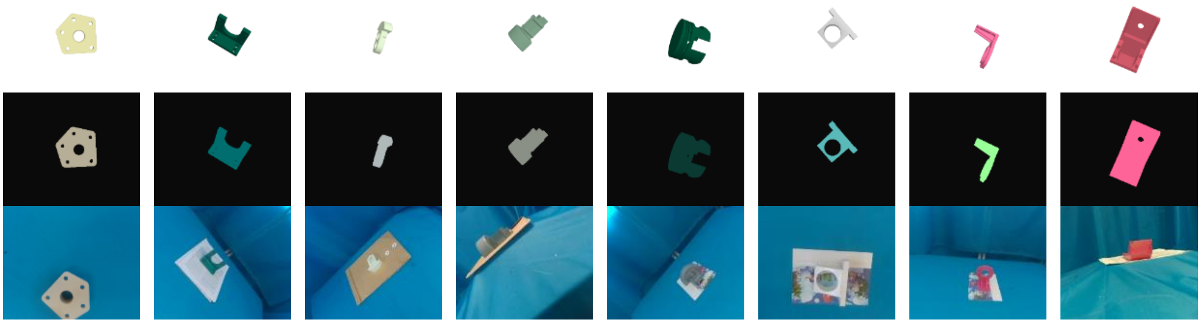


Figure 3. Our workpiece dataset. Workpiece 1-8 are listed left to right. Rows correspond images rendered by CAD, labels of rendered images and images captured from the real.Real images are captured with 4 different background, second column is background 2,third column is is background 1, sixth column is background 3.

4.1.2. Scenario Dataset. We also consider the synthetic-to-real scenarios with GTA5 [14]→CityScapes [15]. The GTA5 Dataset includes 24,966 annotated images of size 1052×1914 rendered by the GTA5 game engine. The Cityscapes train dataset is treated as target domain. Specially, we take this task as kind of transfer learning task, using models trained on GTA5 dataset as the initial.

4.2. Implementation Detail

4.2.1. Workpiece Dataset. For the initial mask generation described in Section 2.1, threshold is set as 1 with decay 0.99 in the step mask generated from CAM and stopped while masked pixels are more than 2% of total pixels. PSP-Net [7] is used for segmentation network with backbone network ResNet-18. Our implementations are based on Pytorch. Optimizer is Adam with initial learning rate 10^{-4}.

4.2.2. GTA5 to CityScapes. The initial mask comes from synthetically trained model on GTA5 dataset, and other operation remains same as described above. To boost the performance, a more powerful backbone network Resnet-101 is used as better feature-excavator. Optimizer is SGD. Same as PSP-Net, the learning rate updates with the base one multiplying $\left(1-\frac{iter}{max\_iter}\right)^{power}$, initial learning rate 10^{-5}, power 0.98, max iteration is set 40k.

4.3. Result

4.3.1. Workpiece Dataset. For our workpiece dataset, 18 images from different background and different viewpoint are labelled to evaluate the proposed framework. The mIoU results are shown in Table 1, images and labels are list in Figure 4. Multi-workpiece also tested as shown in Figure 5.

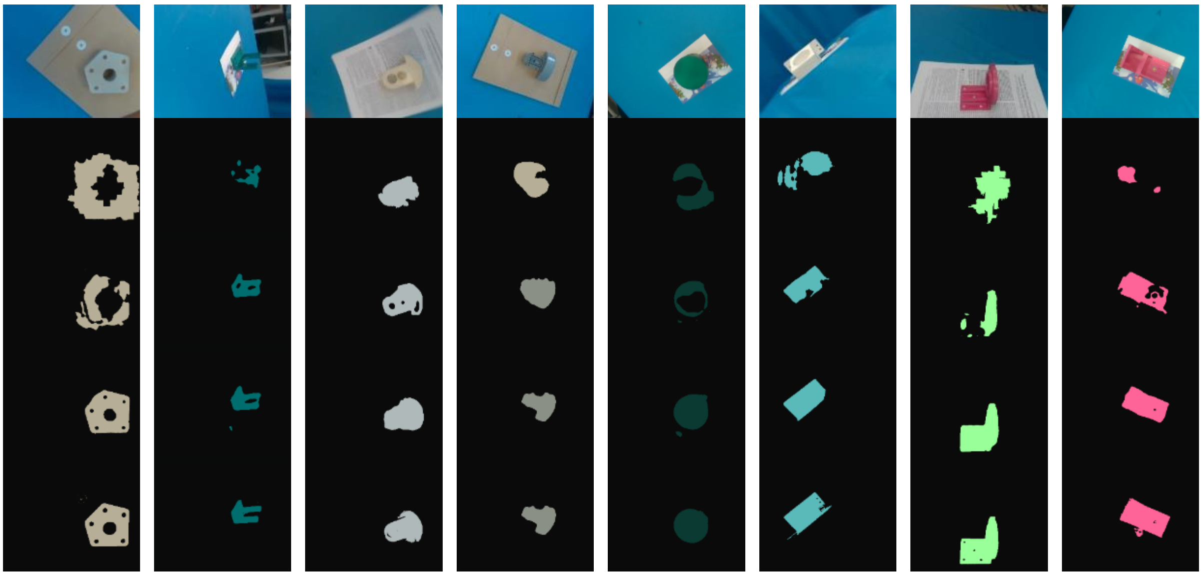


Figure 4. Result on workpiece dataset. Columns correspond to workpiece 1-8, rows correspond to images, the initial mask, predictions after first update, final predictions, the predictions after CRF refinement.

Table 1. Experimental results for workpiece dataset

		wp1	wp2	wp3	wp4	wp5	wp6	wp7	wp8	mIoU
no	initial	48.29	61.27	67.85	68.19	75.53	76.51	77.14	48.74	65.44
	final	**86.94**	**66.64**	**83.89**	**100.00**	**100.00**	**100.00**	**100.00**	**98.66**	**92.02**
1	initial	39.83	49.19	48.03	65.72	58.22	63.47	63.66	55.86	55.50
	final	**71.81**	**83.61**	**67.94**	**88.00**	**86.37**	**78.20**	**81.19**	**88.37**	**80.69**
2	initial	25.94	67.40	37.28	60.35	50.00	33.93	54.00	57.48	48.30
	final	**77.98**	**84.21**	**57.04**	**89.66**	**90.54**	**63.90**	**86.16**	**89.05**	**79.82**
3	initial	28.33	36.07	44.00	43.93	41.54	41.81	51.22	38.20	40.64
	final	**59.98**	**80.62**	**69.22**	**84.40**	**88.79**	**65.84**	**79.50**	**84.15**	**76.56**

4.3.2. Scenario Dataset. Result of the experiments on the Scenario Dataset (GTA5→CityScapes) is shown on Table 2 and the result compared with other methods is shown on Table 3. Figure 6 gives the visualization result.

Table 2. Experimental results for GTA5→CityScapes

	Road	SW	Build	Wall	Fence	Pole	TL	TS	Veg.	Terrain
initial	64.42	22.05	67.56	5.30	5.45	24.15	13.82	14.72	72.73	16.48
ours	**66.01**	11.36	57.62	**7.62**	1.83	17.54	**18.72**	**15.08**	**76.60**	**29.66**
	Sky	PR	Rider	Car	Truck	Bus	Train	Motor	Bike	mIoU
initial	62.54	34.55	1.67	65.22	3.40	3.70	0.38	5.55	0.37	32.69
ours	**62.66**	**42.07**	**3.51**	**72.99**	**19.91**	**12.99**	**1.26**	**16.84**	**4.29**	**40.16**

Table 3. Compared with other methods

Method	mIoU
FCN wild[16]	27.1
Curr. DA[17]	28.9
CyCADA (pixel)[18]	39.5
I2I Adapt[19]	35.7
Ours	**40.16**

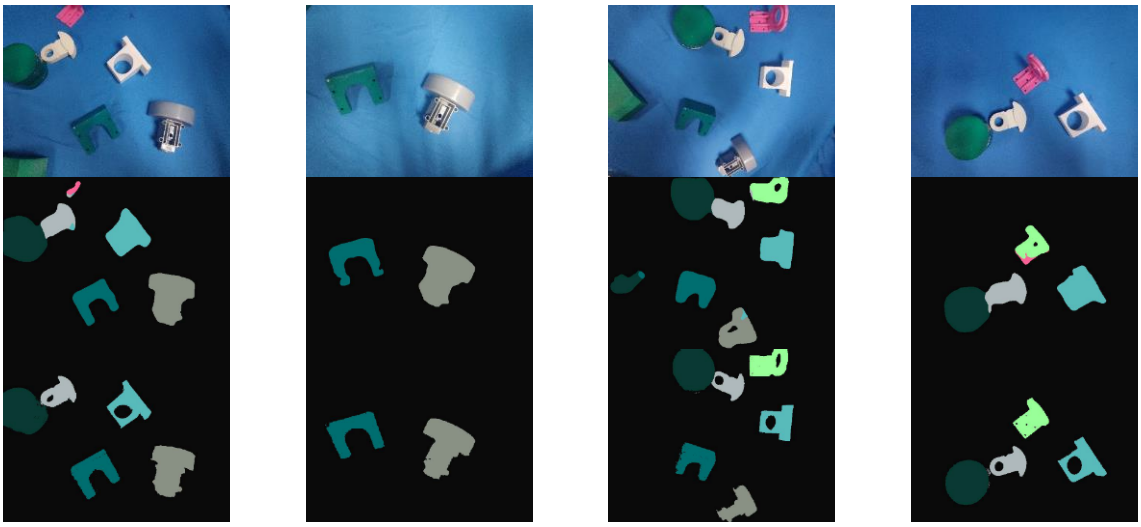


Figure 5. Result on multi workpiece dataset. Rows correspond to images, predictions, and predictions after CRF refinement

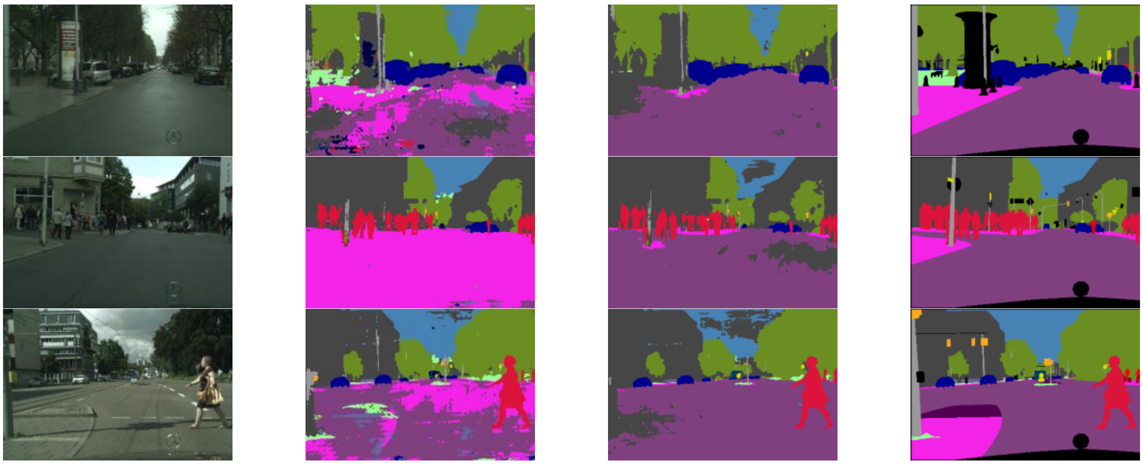


Figure 6. Result on GTA5 to CityScapes. Columns correspond to images, initial predictions, final predictions and ground truth

4.4. Ablation Study

To evaluate the effectiveness of the different part in the proposed framework, some experiments are conducted to explore how each design influences the overall performance. Table 4 details the improvement by considering one more factor at each stage in our proposed framework with background 2.

Table 4. Ablation Study on Workpiece Dataset

	wp1	wp2	wp3	wp4	wp5	wp6	wp7	wp8	mIoU
Initial mask	25.94	67.40	37.28	60.35	50.00	33.93	54.00	57.48	48.30
Initial Only	38.04	43.08	1.21	47.84	59.53	5.38	13.54	48.66	32.16
CAD added	39.27	45.28	34.61	66.15	52.17	67.81	38.50	35.85	47.46
Spatial loss	50.90	82.81	60.83	82.53	83.64	28.79	74.79	63.36	65.96
Label refine	77.98	84.21	57.04	89.66	90.54	63.90	86.16	89.05	79.82

5. Conclusion

In this paper, we proposed a self-training framework for the weakly supervised semantic segmentation task. This framework aims at refining pseudo-labels with local relativeness from the image itself. The spatially weighted loss is also introduced to alleviate the effect of wrong predictions. Experiment demonstrates that our method could use the information from the images to refine the pseudo-labels, which also suggests self-training-based approach could be quite effective.

Acknowledgement

This work is supported by National Natural Science Foundation of China under the grants 61503243 and U1405251, and in part by the National Instrument Development Special Program of China (2013YQ03065101)

References

[1] Grandvalet Y, Bengio Y. Semi-supervised learning by entropy minimization. 2005. In: *Advances in neural information processing systems*; p. 529–536.

[2] Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A. Learning deep features for discriminative localization. 2016. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; p. 2921–2929.

[3] Bansal A, Chen X, Russell BC, Gupta A, Ramanan D. 2017. PixelNet: Representation of the pixels, by the pixels, and for the pixels. CoRR; abs/1702.06506. Available from: http://arxiv.org/abs/1702.06506.

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

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

[6] Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL. 2016. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *arXiv preprint* arXiv:160600915.;.

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

[8] Lin G, Milan A, Shen C, Reid I. 2017. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*; p. 1925–1934.

[9] Peng C, Zhang X, Yu G, Luo G, Sun J. 2017. Large Kernel Matters–Improve Semantic Segmentation by Global Convolutional Network. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*; p. 4353–4361.

[10] Kolesnikov A, Lampert CH. 2016. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In: *European Conference on Computer Vision*. Springer; p. 695–711.

[11] Wei Y, Xiao H, Shi H, Jie Z, Feng J, Huang TS. 2018. Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; p. 7268–7277.

[12] Wang X, You S, Li X, Ma H. 2018. Weakly-supervised semantic segmentation by iteratively mining common object features. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*; p. 1354–1362.

[13] Achanta R, Shaji A, Smith K, Lucchi A, Fua P, Süsstrunk S, et al. 2012. SLIC super pixels compared to state-of-the-art super pixel methods. *IEEE transactions on pattern analysis and machine intelligence.*; **34(11)** : p 2274–2282.

[14] Richter SR, Vineet V, Roth S, Koltun V. 2016. Playing for Data: Ground Truth from Computer Games. In: Leibe B, Matas J, Sebe N, Welling M, editors. *European Conference on Computer Vision (ECCV)*. vol. **9906** of LNCS. Springer International Publishing; p. 102–118.

[15] Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, et al. 2016. The cityscapes dataset for semantic urban scene understanding. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*; p. 3213–3223.

[16] Hoffman J, Wang D, Yu F, Darrell T. 2016. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv preprint* arXiv:161202649.;.

[17] Zhang Y, David P, Gong B. 2017. Curriculum domain adaptation for semantic segmentation of urban scenes. In: *The IEEE International Conference on Computer Vision (ICCV)*. vol. **2**; p. 6

[18] Hoffman J, Tzeng E, Park T, Zhu JY, Isola P, Saenko K, et al. 2017. Cycada: Cycle-consistent adversarial domain adaptation. *arXiv preprint* arXiv:171103213.;.

[19] Murez Z, Kolouri S, Kriegman D, Ramamoorthi R, Kim K. 2018. Image to image translation for domain adaptation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; p. 4500–4509.