Bootstrapping the Performance of Webly Supervised Semantic Segmentation

Tong Shen[1], Guosheng Lin[2], Chunhua Shen[1], Ian Reid[1]
[1]School of Computer Science, The University of Adelaide, Australia
[2]School of Computer Science and Engineering, Nanyang Technological University, Singapore
{tong.shen, chunhua.shen, ian.reid}@adelaide.edu.au
gslin@ntu.edu.sg

Abstract

Fully supervised methods for semantic segmentation require pixel-level class masks to train, the creation of which is expensive in terms of manual labour and time. In this work, we focus on weak supervision, developing a method for training a high-quality pixel-level classifier for semantic segmentation, using only image-level class labels as the provided ground-truth. Our method is formulated as a two-stage approach in which we first aim to create accurate pixel-level masks for the training images via a bootstrapping process, and then use these now-accurately segmented images as a proxy ground-truth in a more standard supervised setting. The key driver for our work is that in the target dataset we typically have reliable ground-truth image-level labels, while data crawled from the web may have unreliable labels, but can be filtered to comprise only easy images to segment, therefore having reliable boundaries. These two forms of information are complementary and we use this observation to build a novel bi-directional transfer learning framework. This framework transfers knowledge between two domains, target domain and web domain, bootstrapping the performance of weakly supervised semantic segmentation. Conducting experiments on the popular benchmark dataset PASCAL VOC 2012 based on both a VGG16 network and on ResNet50, we reach state-of-the-art performance with scores of 60.2% IoU and 63.9% IoU respectively[1].

1. Introduction

Semantic image segmentation is a fundamental problem in computer vision whose aim is to predict a category label for each pixel of an image. Recent approaches [19, 18, 2, 17, 20, 33] based on Deep Convolutional Neural Networks (DCNN) have achieved remarkable success. However, unlike training classification networks [9, 28, 15, 32], which only requires image-level labels, training a network for semantic segmentation involves a large amount of pixel-level labels.

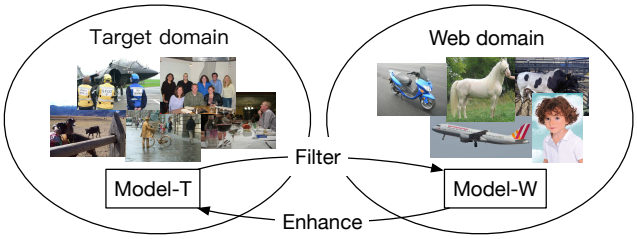

Figure 1: Illustration of the bi-directional framework. Model-T and Model-W are trained in the target domain and the web domain respectively. Model-T uses the knowledge in its domain to help Model-W to filter out image with incorrect tags, yielding a set of high quality easy images. Model-W trained with high quality web images transfers the knowledge back to the target domain, helping Model-T enhance the results.

As shown in [1], annotating pixel-level labels is very time-consuming, taking an average of 239.7 seconds for a single image. In contrast, obtaining image-level labels only takes 20 seconds or less per image. This motivates exploring the possibility of using partially annotated or weakly annotated data to achieve reasonable performance. To this end, a number of semi- or weakly supervised methods have been proposed [24, 1, 26, 16, 13, 30, 6, 23]. These methods utilize different levels of supervision including bounding boxes, scribbles, points, image-level labels, *etc*. Points indicate the location of the object; bounding boxes and scribbles imply the extent of the object; image-level supervision only indicates the presence of the object. Among various types of supervision, image-level supervision is undoubtedly the weakest one. In this paper, we focus explicitly on this task of using the weakest supervision; *i.e*. semantic segmentation with only image-level labels.

We tackle the problem by focusing on generating the

[1]Our code is available at https://github.com/ascust/BDWSS

pixel-wise masks for the training images to create a proxy ground-truth dataset. Using this proxy ground-truth dataset, we train a Fully Convolutional Network (FCN) for the task. Our framework is designed to generate high-quality masks, close in accuracy to those created by humans, and to use these masks to train a network.

Web data exist in large quantities and we can easily collect a group of images associated with a particular class label by using the label (and synonyms) as a query to a search engine. The hope is that these extra data can be used to boost the performance, and indeed a number of papers [30, 11, 26] have previously explored this idea to improve results of weakly supervised methods. There are two hurdles to overcome; the first is that the retrieved web data will often be noisy, in the sense that the image labels (tags) may not match the image content, or be inconsistent with the concept/object we are trying to capture. The second is of course that the retrieved images will not have the ground-truth segmentation masks associated with them.

In this paper we describe a bootstrapping process, in which we leverage bi-directional flow of information between two domains, a target domain (*i.e.* the set of classes for which we want segmentation and a set of training images with accurate image-level labels) and the web domain (*i.e.* images crawled from the web using the target class labels as search keywords). For simplicity, we use Model-T and Model-W to represent models in the target and web domain respectively (see Figure 1). The key insight is that we can use a weakly supervised network (Model-T, trained on the target domain using only image-level labels) to effectively filter the web-retrieved images to eliminate labelling errors and to retain only images that are relatively easy to segment, having a simple background, single semantic class, and decent-sized objects. By doing this, we create a new dataset with high quality images that are easier to segment with only weak supervision. Figure 2 illustrates typical images and segmentation results from the two domains. Images in the target domain usually have a complex scene and multiple, overlapping objects, whereas web images filtered are simpler and therefore easier to segment using a weakly supervised network.

Since the model trained with the target dataset can filter the web data and provides us with a high quality dataset, we propose to learn a model with these web images and in return help enhance our results. As shown in Figure 3, the first two masks are estimated by the model trained with the target dataset and web images respectively. We observe that the model trained with the target dataset is good at distinguishing semantic classes but provides bad boundaries, while the model trained with web images gives good boundaries but tends to merge different semantic regions. By our merging strategy, the enhanced mask, shown in right bottom of Figure 3, takes advantage from both masks and makes high

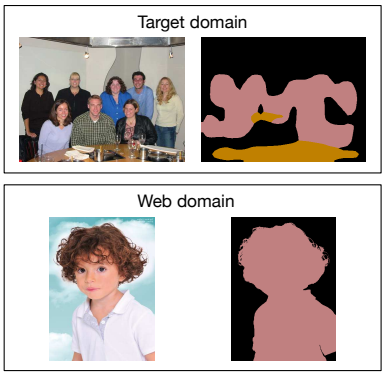

Figure 2: Mask estimation in two domains. In the upper part, the mask is given by the model trained in the target domain, which is coarse due to complex scene and overlapping objects of the images. The lower part shows an example given by the model trained in the web domain, which is better because of the simple context.

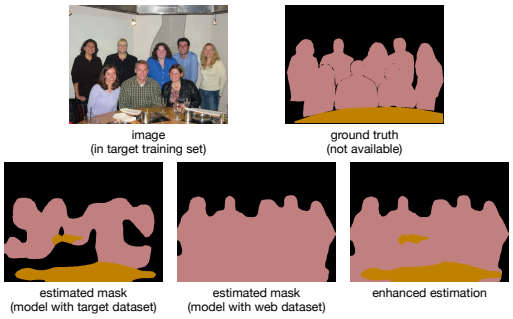

Figure 3: Illustration of enhancing mask. The upper part shows an image in the training set and ground truth (which is not available in our weakly supervised setting). In the lower part, the first two masks are estimated by the model in the target domain and the web domain respectively and the last one is the enhanced mask.

quality estimation. There is also the ground truth annotation in upper right for visual comparison, which is not available in our weakly supervised setting.

Our contributions can be summarized as follows:

- We propose a bidirectional transfer learning framework for bootstrapping webly supervised semantic segmentation.
- We propose an effective approach to filter web data and find high quality images, which are suitable for weakly supervised semantic segmentation.
- We transfer the knowledge learnt from the web domain to the target domain and generate high quality masks.
- By using the high quality masks as proxy ground truth, we train a standard FCN and achieve state-of-the-art

performance. The gap between weakly supervised methods and fully supervised methods is further reduced.

2. Related Work

Semantic segmentation has greatly benefited from FCN based networks that enable training dense prediction models in an end-to-end fashion. Many methods have been proposed [19, 18, 2, 17, 20, 33] and achieved remarkable success. However these methods are designed in fully supervised setting and require pixel-level masks, which involves a large amount of human labour and time to obtain.

In order to reduce the effort of annotation, many semi- and weakly supervised methods have been proposed [24, 1, 26, 16, 13, 30, 6, 23]. In these methods, various forms of supervision are investigated to achieve reasonable performance compared with fully supervised methods. In [6], Dai *et al*. propose a bounding box supervised method where they extract object masks based on the bounding box by using MCG. In [16], Lin *et al*. use scribbles as supervision and construct a graphical model to tackle the problem. In [1] only points are used as supervision to train a model. Among these supervisions, the most challenging one is image-level annotation. Pathak *et al*. [23] introduce a constrained convolutional neural network with assumptions on object size, foreground and background. Pinheiro *et al*. [24] propose a Multiple Instance Learning (MIL) based method for the problem. In [13], a "seed, expand and constrain" (SEC) framework is proposed using only image-level labels where localization cues from classification networks are used to find the object; a weighted rank pooling loss is used to constrain the object extent; CRF is used to refine the boundaries. Our method uses SEC model as a starting point and use web images to learn better features.

Our method is closely related to webly supervised learning [4, 14, 31, 5], which is focused on extracting useful knowledge or features from noisy web data. Many webly based semantic segmentation methods have also been proposed [11, 26, 30, 10]. In [11, 30], a network is firstly trained with simple images from the internet and the corresponding masks estimated using saliency detection. Then the network is adapted to the target domain with progressive improvement. Shen *et al*. [26] use co-segmentation to extract the masks of web images and train the network. Hong *et al*. [10] use data from the web crawled videos and extract masks based on temporal information and attention cues.

3. Method

The pipeline of our framework is described in Figure 4. Our goal is to estimate the masks for training images in the target domain, which will then be used as a proxy for ground truth to train the final segmentation network. The models in two domains interact with each other to transfer knowledge and finally provide us with high quality masks for the training images.

In detail, our bi-directional framework is based on the two domains:

- In the **target domain**, we train Initial-SEC on VOC images with only image-level labels and get initial estimation of the masks. Details are presented in Section 3.1.
- In the **web domain**, we transfer the knowledge from target domain by using Initial-SEC as a filter to clean noisy web data. Then we have three steps to learn the knowledge from the web domain by training Web-SEC (Section 3.2.2), using Grabcut refinement (Section 3.2.3) and training Web-FCN (Section 3.2.4).
- Back to the **target domain**, we transfer the knowledge from the web domain back to enhance the initial estimation of the masks, which is described in Section 3.3.
- Finally Final-FCN is trained using the estimated masks, as described in Section 3.4.

3.1. Training Initial-SEC in the Target Domain

Our framework starts in the target domain, where we train a SEC model, termed Initial-SEC, on VOC images. We first review the SEC architecture [13]. Let $I = \{(\mathbf{X}_n, \mathbf{Y}_n)\}^{N_1}$ be our target dataset, *e.g*. PASCAL VOC 2012, which consists of N_1 images. Each Image $\mathbf{X}_n$ is annotated by image-level labels $\mathbf{Y}_n \in \{0,1\}^C$ where C is the number of classes. The goal is to train a DCNN $f(\mathbf{X})$, short for $f(\mathbf{X};\theta)$, that is parameterized by θ and models category probabilities for each pixel. The SEC model is trained by three losses:

$$\mathcal{L} = \sum_{n}^{N_1} \mathcal{L}_{seed}(f(\mathbf{X}_n), \mathbf{Y}_n) + \mathcal{L}_{expand}(f(\mathbf{X}_n), \mathbf{Y}_n) + \mathcal{L}_{constrain}(f(\mathbf{X}_n), \mathbf{X}_n) \quad (1)$$

$\mathcal{L}_{seed}$ supervises the network with localization cues obtained from Class Activation Mapping (CAM) [34]. $\mathcal{L}_{expand}$ controls how to aggregate the heat maps to be consistent with image-level labels where a global weighted rank pooling (GWRP) is proposed. $\mathcal{L}_{constrain}$ makes the predictions respect the boundaries of objects.

In the original paper, the trained model is the final model. Unlike their approach, we apply the model back to the training images to generate their masks. These masks are coarse, as shown in left bottom of Figure 3, and will be enhanced by the model trained in the web domain.

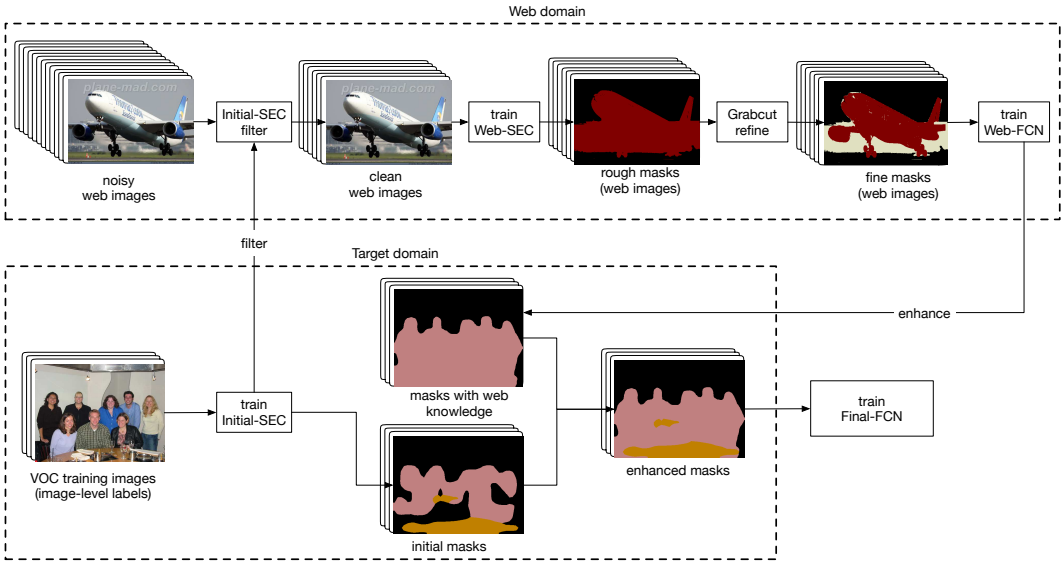

Figure 4: Illustration of our pipeline. Assuming the target dataset is PASCAL VOC 2012, the target domain contains the training images in VOC with image-level labels, shown in the lower rectangle with dashed lines. The web domain has noisy (*i.e.* incorrectly labelled) images, represented in the upper rectangle with dashed lines. Beginning with the target domain, we first train Initial-SEC to generate rough initial masks. We then use this model as a filter to clean the noisy web data and remove complex images, retaining easy-to-segment ones. In the (filtered) web domain, we train another SEC model (Web-SEC) to get rough masks for the web images and Grabcut refinement to further refine the masks. Then a FCN (Web-FCN) is trained on these data to represent the knowledge in the web domain. This model in turn enhances the estimation of the initial masks to generate high quality masks. The last step is to train Final-FCN using the proxy ground truth.

Since we have access to image-level labels, we use them to further refine the masks of the training images as follows:

$$m_i = \underset{j \in \{1,..,C\}}{\arg\max}\, y_i f_{ij} \tag{2}$$

where m_i is the mask prediction for ith pixel (*i.e.* we choose the class label as the most likely one from the set of valid labels). An example is illustrated in Figure 5. Compared with the raw prediction on left bottom, the confusion is removed in the refined prediction shown on right bottom. We also use the ground truth annotation for visual comparison, which is not available in our setting.

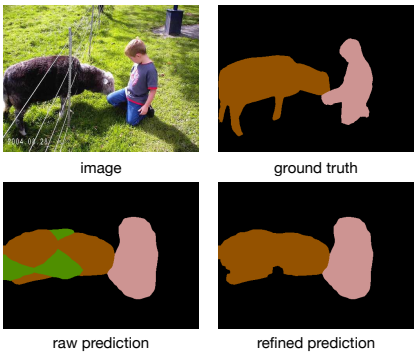

Figure 5: Illustration of removing confusions of the initial masks by using image-level labels. Given an image in upper left, the raw estimation is shown in lower left. Using this information, we get cleaned estimation in right bottom. We also use the ground truth annotation for visual comparison(not available in our setting).

3.2. Training Models in the Web Domain

The masks estimated from Section 3.1 are still too rough to be used as the ground truth, as shown in right upper of Figure 2. In this section we show how we can leverage web-crawled data, transferring knowledge from the target domain to the web domain and learn new knowledge in the web domain.

3.2.1 Crawl and Filter Web Images

High quality web data processed by good filtering methods are crucial to learning good segmentation models. In this section, we show how to transfer the knowledge from the target domain to filter web data.

We first search for images based on class names using search engines (Bing in our experiments). The class names are used as seeds, along with synonyms, and similar words

suggested by the search engines. For example, when searching for "dog", "German Shepherd dog", "Pitbull dog" *etc*. are also suggested. After greedily crawling all related images, we use the Initial-SEC model trained on VOC images as our filter to clean the web data.

Applying the SEC model to web images, we are able to obtain masks with per-pixel class labels. Based on the dense masks information, we can easily identify qualified images by scene complexity of the image, extent of the object and semantic relevance. Specifically, we select the images according to two criteria: (i) the number of pixels for the target class must lie in a predefined range, $t_1 < \frac{1}{N}\sum_i \mathbb{1}(m_i = c) < t_2$; and (ii) the number of other foreground pixels should be lower than a threshold, $\frac{1}{N}\sum_i \mathbb{1}(m_i \neq c \text{ and } m_i \neq background) < t_3$. The intuition is we want to select images with a "proper" size for the foreground. It is expected that such images can be easily segmented. Different from existing filtering approaches [30, 10], our method is based on dense masks and provides richer information of the images.

3.2.2 Training of Web-SEC

The filtering process described above creates a dataset of accurately labelled, high quality web images from a noisy web search. Our goal now is to improve the estimates of their masks. To this end, we train another SEC model on the web data which we term "Web-SEC". Unlike in the target domain, where images are associated with multiple class labels, images in the web domain are much simpler, filtered to be likely to contain only one class, and therefore easier to segment.

The Web-SEC model is able to generate masks for these web images of higher quality than Initial-SEC. Figure 6 shows a qualitative comparison between these two models (Initial-SEC and Web-SEC). The middle masks are from Initial-SEC trained in the target domain. It gives basic semantic information and rough extent of the object. Clearly the masks on the right, outputs from Web-SEC, are well adapted to the web domain and provide more accurate estimation.

3.2.3 Grabcut Refinement

The masks generated by Web-SEC are good at capturing the whole object but sometimes overestimate the object, as illustrated in the second column of Figure 7. To further refine the masks, we develop a Grabcut based refinement method. It is similar to [12], but we use the mask as prior knowledge to indicate the foreground and background instead of the bound box. We simply jitter the window that tightly surrounds the mask and perform Grabcut [25]. By multiple samples, we are able to get a probability heat map of the foreground as shown in the third column of Figure 7, and

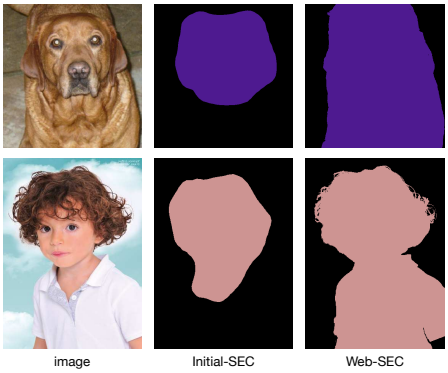

Figure 6: Comparison of the estimated mask for web images between Initial-SEC model and Web-SEC model. The middle column shows the masks estimated from Initial-SEC model, which are coarse. The masks on the right are from the Web-SEC model, which provide more accurate estimation.

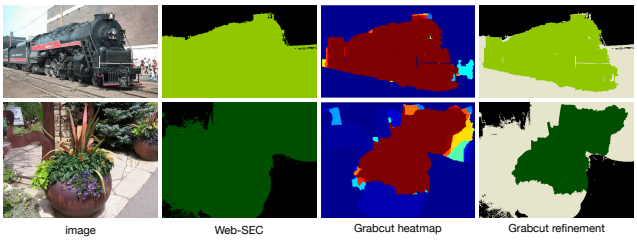

Figure 7: Illustration of Grabcut refinement. The second column shows the masks from Web-SEC model. The third column shows the probability heat map after Grabcut. The last column shows the refined masks.

we retain as foreground only the pixels with high probability.

For a mask estimated by Web-SEC, $l_i \in \{1, ..., C\}$ is the label for ith pixel. After Grabcut refinement, we have $p_i \in [0, 1]$ for ith pixel representing the probability of being kept. The refined mask is defined as:

$$\hat{l}_i = \begin{cases} l_i & \text{if } p_i \geq t \\ background & \text{if } p_i < t \text{ and } l_i = background \\ void & \text{if } p_i < t \text{ and } l_i \neq background \end{cases} \quad (3)$$

where $\hat{l}_i$ is the new label for ith pixel; t is the threshold; $void$ indicates unclear regions.

We are able to control the balance between precision and recall by choosing a proper threshold. By using a high threshold, we have high confidence about the pixels being kept. Since those with low probability are set to void, they will be ignored during the training and not have a big impact.

3.2.4 Training of Web-FCN

After Section 3.2.3, we obtain a web image dataset with estimated masks. Let $W = \{(\mathbf{X}_n, \mathbf{M}_n)\}^{N_2}$ be the dataset with N_2 images, where $\mathbf{X}_n$ and $\mathbf{M}_n$ are the image and the estimated mask respectively. We now are able to train a standard FCN (Web-FCN), which is used to estimate masks for our target dataset. The architecture we adopt here is a 1/8 resolution FCN with dilated convolution kernels, similar to DeepLab [2]. This becomes a "fully supervised" problem and the objective is to minimize a softmax loss:

$$\mathcal{L} = \sum_{n}^{N_2} \mathcal{L}_{softmax}(f(\mathbf{X}_n), \mathbf{M}_n) \tag{4}$$

The Web-FCN trained in the web domain encodes the knowledge in this domain. The knowledge will be transferred to the target domain by applying this model to the target dataset.

3.3. Enhancing the Initial Estimation

In this section, we describe how to transfer the knowledge learnt from the web domain to the target domain and improve the estimation. Recall that in lower part of Figure 3, the first two masks are from models in the target and the web domain respectively. We observe that the model in the target domain is good at distinguishing classes because it is trained with confident image-level labels. In contrast, the model in the web domain provides better boundaries and captures more complete extent but is prone to making mistakes about the class labels. We address this by fusing the estimations from both domains and get the final enhanced mask, shown in right bottom of Figure 3.

More specifically, let $M^{(t)}$ be the mask from the target domain and $M_i^{(t)} \in \{1, ..., C\}$ represent the category for ith pixel. Likewise, $M^{(w)}$ and $M^{(f)}$ represent the mask from the web domain and the final enhanced mask respectively. The fusion strategy is as follows:

$$M_i^{(f)} = \begin{cases} M_i^{(t)} & \text{if } M_i^{(w)} \neq background \\ M_i^{(t)} & \text{if } M_i^{(w)} = background \\ & \text{and } \sum_k \mathbb{1}(M_k^{(w)} = M_i^{(t)}) < \epsilon \\ M_i^{(w)} & \text{otherwise} \end{cases} \tag{5}$$

where ϵ is a small number.

The intuition for this strategy is that for foreground pixels in $M^{(w)}$, the category labels will follow $M^{(t)}$ because it has better ability to distinguish classes. For background pixels in $M^{(w)}$, if the number of pixels for a valid class is lower than a threshold, we also follow the label in $M^{(t)}$. This indicates if a class is shown in image-level labels, we should guarantee some pixels for this class, otherwise the information for this class will be lost. In any other cases, we follow $M^{(w)}$.

3.4. Training Final-FCN

After obtaining the enhanced masks, the problem is similar to a "fully supervised" problem. The target dataset becomes $I = \{(\mathbf{X}_n, \mathbf{Y}_n, \mathbf{M}_n)\}^{N_1}$, where we have pixel-wise masks besides image-level labels. This enables us to train a standard FCN model. The structure we adopt in our experiment is a FCN with dilated kernels, similar to DeepLab [2]. Besides, we also adopt a global-multi label branch for scene consistency, as in [27]. We train Final-FCN by minimizing two loss functions:

$$\mathcal{L} = \sum_{n}^{N_1} \mathcal{L}_{softmax}(f(\mathbf{X}_n), \mathbf{M}_n) + \mathcal{L}_{multi}(g(\mathbf{X}_n), \mathbf{Y}_n) \tag{6}$$

where $g(\mathbf{X}_n)$ is the output for global multi-label and $\mathcal{L}_{multi}$ is a logistic multi-label loss.

4. Experiments

4.1. Dataset

Retrieved Dataset: We retrieve images from Bing based on class names. We use class names as seeds and greedily search for related images, including synonyms, words suggested by the searching engine. By using our Initial-SEC as a filter and setting a threshold for each class as the maximum number of images, we obtain a retrieved dataset with 76683 images. All images are resized so that the larger dimension is 500. In term of the parameters mentioned in Section 3.2.1, $t_1 = 0.3$, $t_2 = 0.7$ and $t_3 = 0.1$.

PASCAL VOC 2012: We use this dataset as our target dataset and evaluate the performance based on this. The original dataset [7] contains 1464 training images, 1449 validation images and 1456 testing images. As common practice, we also use the augmented data from [8], which gives 10582 training images in total. There are 21 classes including a background class. The result is evaluated with Intersection over Union (IoU) averaged over 21 classes.

4.2. Implementation Details

The implementation is based on MXNet [3]. For details of training SEC models, Initial-SEC and Web-SEC, please refer to the original paper [13]. We follow the same parameters except that for training Web-SEC, we use a smaller initial learning rate of 1e-4. For Grabcut refinement, Section 3.2.3, we set the threshold $t = 0.7$. For Web-FCN we use DeepLab-based [2] structure, which has output resolution of 1/8. For Final-FCN, apart from the basic structure, a global multi-label branch is also introduced to encourage scene consistency, similar to [27]. We use standard

Method	val	test	Extra Supervision
Chen *et al*. [2]	**63.7**	**66.4**	Fully supervised
Lin *et al*. [16]	63.1	-	Scribble
Dai *et al*. [6]	62.0	64.6	Bounding box+MCG
Oh *et al*. [21]	55.7	56.7	Bounding box
Bearman *et al*. [1]	46.1	-	Point
Wei *et al*. [29]	55.0	55.7	Supervised saliency
STC [30]	49.8	51.2	Supervised saliency
EM-Adapt [22]	33.8	39.6	-
CCNN [23]	35.3	35.6	-
SEC [13]	50.7	51.7	-
Hong *et al*. [10]	58.1	58.7	-
Ours-VGG16	58.8	60.2	-
Ours-Res50	**63.0**	**63.9**	-

Table 1: Comparison with methods using other supervisions.

T-domain	Web-domain				
Initial-SEC	Web-SEC	GC	Web-FCN	post	IoU
✓					49.3
✓	✓				52.6
✓	✓		✓		55.7
✓	✓	✓	✓		56.6
✓	✓	✓	✓	✓	**58.8**

Table 2: Comparison under different settings on the PASCAL VOC 2012 validation set.

Stochastic Gradient Descent (SGD) for optimization. For post-processing, multi-scale inference and dense-CRF are used as common practice.

4.3. Experiment Results

The results on PASCAL VOC validation set and test set are shown in Table 4 and Table 5 respectively. According to the tables, the one with VGG16 [28], same as the other's base network, already achieves state of the art performance, 60.2[2]. By using another base net, Resnet 50 [9], we achieve much better result 63.9[3], which significantly outperforms other methods.

Table 1 also shows a comparison with methods using different supervision, where the extra supervision is explained in the last column. In the upper half of the table, we list methods with stronger supervision than image-level labels. It is worth noting that our method does not use any other auxiliary methods that involve extra supervision. Some qualitative examples are shown in Figure 8.

4.4. Ablation Study

4.4.1 Analysis of Different Modules

To analyse the effectiveness of our bi-directional transfer learning framework, we conduct ablation study with different settings. Recall that our goal is to generate high quality masks for the training images and train a FCN using the estimated masks. Therefore, the quality of the masks directly affects the final performance. Table 2 shows a comparison under different settings. Starting with the simplest one where only target domain is involved, we only get 49.3 by using Initial-SEC. With the web domain introduced, we train Web-SEC for the web images, which gives us 3.3 point improvement. This indicates the effectiveness of the knowledge transferred from the web domain. We continue training Web-FCN without using Grabcut refinement and further improve the result to 55.7. By using Grabcut refinements, we get almost one more point of improvement. The final score is obtained by post-processing including multi-scale inference and dense-CRF as common practice.

[2]http://host.robots.ox.ac.uk:8080/anonymous/X0CH0F.html
[3]http://host.robots.ox.ac.uk:8080/anonymous/GKJXB6.html

Number of web images	IoU
76.7k	**56.6**
58.1k	56.4
39.1k	56.3
20.0k	56.4
10.0k	56.4
6k	55.7
2k	55.3
80.0k without filtering	49.8

Table 3: Ablation study using different number of web images on the PASCAL VOC 2012 validation set.

4.4.2 Analysis of Number of Web Images

It is also interesting to analyse how the number of web images involved affects the result. Table 3 shows an ablation study using different numbers of web images.The best performance is obtained by using 76.7k images. We also run experiments with different numbers of images by varying the threshold of maximum images for each class. It is interesting that the performance does not drop much with the number of web images decreasing. Even the number of images is decreased to 2k, the performance only drops by 1.3%. This indicates that our bi-directional framework is pretty robust to noise and the filtered images are high quality. Furthermore, we also show an experiment without filtering the images, which is shown in the last row. Using 80k noisy web images, we only get score of 49.8, which is 6.8 lower than the best one. This again indicates the importance of using knowledge learnt in target domain to filter web data.

5. Conclusion

In this paper, we tackle the problem of weakly supervised semantic segmentation using only image-level labels. Apart from the target dataset with confident image-level labels, we propose to use noisy web data to boost the perfor-

Method	bk	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mean
EM-Adapt [22]	67.2	29.2	17.6	28.6	22.2	29.6	47.0	44.0	44.2	14.6	35.1	24.9	41.0	34.8	41.6	32.1	24.8	37.4	24.0	38.1	31.6	33.8
CCNN [23]	68.5	25.5	18.0	25.4	20.2	36.3	46.8	47.1	48.0	15.8	37.9	21.0	44.5	34.5	46.2	40.7	30.4	36.3	22.2	38.8	36.9	35.3
MIL+seg [24]	79.6	50.2	21.6	40.9	34.9	40.5	45.9	51.5	60.6	12.6	51.2	11.6	56.8	52.9	44.8	42.7	31.2	55.4	21.5	38.8	36.9	42.0
SEC [13]	82.4	62.9	26.4	61.6	27.6	38.1	66.6	62.7	75.2	22.1	53.5	28.3	65.8	57.8	62.3	52.5	32.5	62.6	32.1	45.4	45.3	50.7
STC [30]	84.5	68.0	19.5	60.5	42.5	44.8	68.4	64.0	64.8	14.5	52.0	22.8	58.0	55.3	57.8	60.5	40.6	56.7	23.0	57.1	31.2	49.8
WebS [11]	84.3	65.3	27.4	65.4	**53.9**	46.3	70.1	69.8	79.4	13.8	61.1	17.4	73.8	58.1	57.8	56.2	35.7	66.5	22.0	50.1	46.2	53.4
Hong *et al.* [10]	**87.0**	69.3	32.2	70.2	31.2	58.4	73.6	68.5	76.5	26.8	63.8	29.1	73.5	69.5	66.5	70.4	46.8	72.1	27.3	**57.4**	50.2	58.1
Ours-VGG16	85.0	**74.4**	24.9	76.2	20.7	58.2	82.3	**73.6**	81.0	25.9	71.3	37.4	71.8	69.6	70.3	71.0	44.1	73.8	34.1	48.4	40.0	58.8
Ours-Resnet50	86.8	71.2	**32.4**	**77.0**	24.4	**69.8**	**85.3**	71.9	**86.5**	**27.6**	**78.9**	**40.7**	**78.5**	**79.1**	**72.7**	**73.1**	**49.6**	**74.8**	**36.1**	48.1	**59.2**	**63.0**

Table 4: Results on the PASCAL VOC 2012 validation set.

Method	bk	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mean
EM-Adapt [22]	76.3	37.1	21.9	41.6	26.1	38.5	50.8	44.9	48.9	16.7	40.8	29.4	47.1	45.8	54.8	28.2	30.0	44.0	29.2	34.3	46.0	39.6
CCNN [23]	70.1	24.2	19.9	26.3	18.6	38.1	51.7	42.9	48.2	15.6	37.2	18.3	43.0	38.2	52.2	40.0	33.8	36.0	21.6	33.4	38.3	35.6
MIL+seg [24]	78.7	48.0	21.2	31.1	28.4	35.1	51.4	55.5	52.8	7.8	56.2	19.9	53.8	50.3	40.0	38.6	27.8	51.8	24.7	33.3	46.3	40.6
SEC [13]	83.5	56.4	28.5	64.1	23.6	46.5	70.6	58.5	71.3	23.2	54.0	28.0	68.1	62.1	70.0	55.0	38.4	58.0	39.9	38.4	48.3	51.7
STC [30]	85.2	62.7	21.1	58.0	31.4	55.0	68.8	63.9	63.7	14.2	57.6	28.3	63.0	59.8	67.6	61.7	42.9	61.0	23.2	52.4	33.1	51.2
WebS [11]	85.8	66.1	30.0	64.1	**47.9**	58.6	70.7	68.5	75.2	11.3	62.6	19.0	75.6	67.2	72.8	61.4	44.7	71.5	23.1	42.3	43.6	55.3
Hong *et al.* [10]	**87.2**	63.9	**32.8**	72.4	26.7	64.0	72.1	70.5	77.8	23.9	63.6	32.1	77.2	75.3	76.2	71.5	45.0	68.8	35.5	**46.2**	49.3	58.7
Ours-VGG16	85.3	**77.6**	26.2	**76.6**	17.3	61.4	82.4	74.8	83.8	25.7	66.9	46.2	74.0	75.6	79.2	70.8	48.3	73.1	40.5	38.8	39.0	60.2
Ours-Resnet50	**87.2**	76.8	31.6	72.9	19.1	**64.9**	**86.7**	**75.4**	**86.8**	**30.0**	**76.6**	**48.5**	**80.5**	**79.9**	**79.7**	**72.6**	**50.1**	**83.5**	**48.3**	39.6	**52.2**	**63.9**

Table 5: Results on the PASCAL VOC 2012 test set.

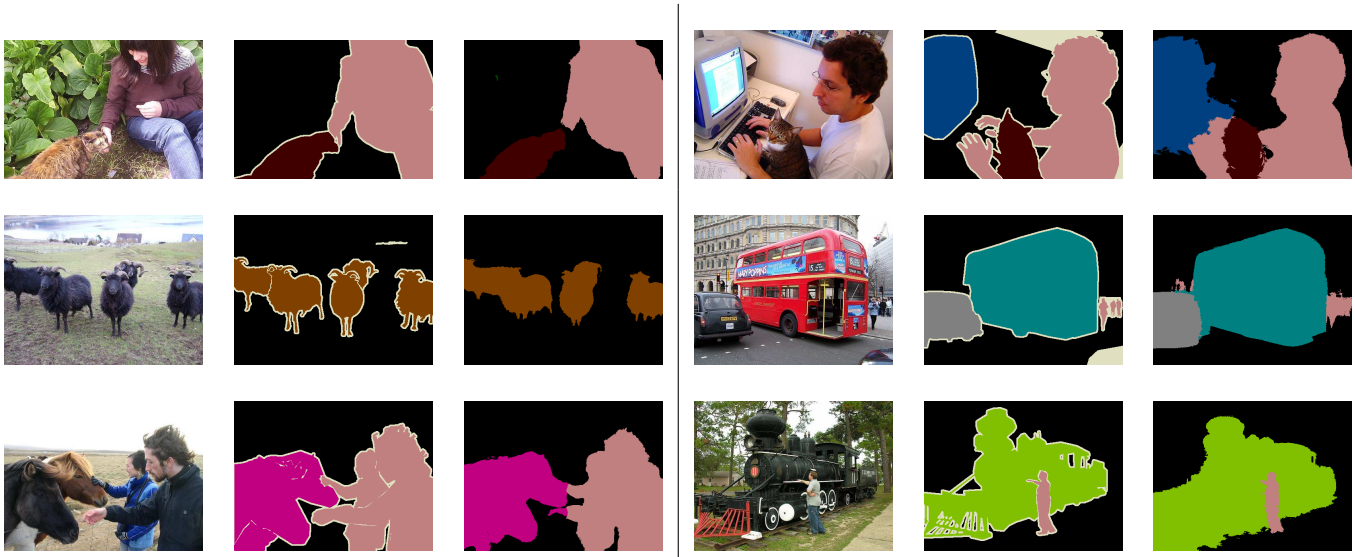

Figure 8: Qualitative results on PASCAL VOC 2012 validation set.

mance. To leverage the data in two domains, target domain and web domain, we propose a novel bi-directional transfer learning framework that is able to generate high quality masks for the training images. Using these masks as proxy ground truth, we achieve state-of-the-art performance and further narrow down the gap between weakly and fully supervised methods.

Acknowledgements

This research was supported by the Australian Research Council through the Australian Centre for Robotic Vision (CE140100016). C. Shen's participation was supported by an ARC Future Fellowship (FT120100969). I. Reid's participation was supported by an ARC Laureate Fellowship (FL130100102).

References

[1] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei. What's the point: Semantic segmentation with point supervision. In *ECCV*, 2016. 1, 3, 7

[2] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. In *ICLR*, 2015. 1, 3, 6, 7

[3] T. Chen, M. Li, Y. Li, M. Lin, N. Wang, M. Wang, T. Xiao, B. Xu, C. Zhang, and Z. Zhang. MXNet: A Flexible and Efficient Machine Learning Library for Heterogeneous Distributed Systems. *NIPS workshop*, 2015. 6

[4] X. Chen and A. Gupta. Webly supervised learning of convolutional networks. In *ICCV*, pages 1431–1439, 2015. 3

[5] X. Chen, A. Shrivastava, and A. Gupta. NEIL: Extracting Visual Knowledge from Web Data. In *ICCV*, 2013. 3

[6] J. Dai, K. He, and J. Sun. [M] BoxSup: Exploiting Bounding Boxes to Supervise Convolutional Networks for Semantic Segmentation. In *ICCV*, 2015. 1, 3, 7

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010. 6

[8] B. Hariharan, P. Arbel, L. Bourdev, S. Maji, and J. Malik. Semantic Contours from Inverse Detectors. In *ICCV*, 2011. 6

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016. 1, 7

[10] S. Hong, D. Yeo, S. Kwak, H. Lee, and B. Han. Weakly Supervised Semantic Segmentation using Web-Crawled Videos. In *CVPR*, 2017. 3, 5, 7, 8

[11] B. Jin, M. V. O. Segovia, and S. Susstrunk. Webly Supervised Semantic Segmentation. *CVPR*, 2017. 2, 3, 8

[12] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele. Simple Does It: Weakly Supervised Instance and Semantic Segmentation. In *CVPR*, 2017. 5

[13] A. Kolesnikov and C. H. Lampert. Seed , Expand and Constrain : Three Principles for Weakly-Supervised Image Segmentation. In *ECCV*, 2016. 1, 3, 6, 7, 8

[14] J. Krause, B. Sapp, A. Howard, H. Zhou, A. Toshev, T. Duerig, J. Philbin, and L. Fei-Fei. The unreasonable effectiveness of noisy data for fine-grained recognition. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9907 LNCS:301–320, 2016. 3

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1

[16] D. Lin, J. Dai, J. Jia, K. He, and J. Sun. ScribbleSup: Scribble-Supervised Convolutional Networks for Semantic Segmentation. In *CVPR*, 2016. 1, 3, 7

[17] G. Lin, A. Milan, C. Shen, and I. Reid. RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation. In *CVPR*, 2017. 1, 3

[18] G. Lin, C. Shen, A. van dan Hengel, and I. Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *CVPR*, 2016. 1, 3

[19] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. In *CVPR*, pages 3431–3440, 2015. 1, 3

[20] H. Noh, S. Hong, and B. Han. Learning Deconvolution Network for Semantic Segmentation. In *ICCV*, volume 1, 2015. 1, 3

[21] S. J. Oh, R. Benenson, A. Khoreva, Z. Akata, M. Fritz, and B. Schiele. Exploiting saliency for object segmentation from image level labels. In *CVPR*, 2017. 7

[22] G. Papandreou, L.-C. Chen, K. Murphy, and A. L. Yuille. Weakly- and Semi-Supervised Learning of a DCNN for Semantic Image Segmentation. In *ICCV*, 2015. 7, 8

[23] D. Pathak, P. Krahenbuhl, and T. Darrell. Constrained Convolutional Neural Networks for Weakly Supervised Segmentation. In *ICCV*, 2015. 1, 3, 7, 8

[24] P. H. O. Pinheiro and R. Collobert. From Image-level to Pixel-level Labeling with Convolutional Networks. In *CVPR*, 2015. 1, 3, 8

[25] C. Rother, V. Kolmogorov, and A. Blake. "GrabCut": interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309, 2004. 5

[26] T. Shen, G. Lin, L. Liu, C. Shen, and I. Reid. Weakly Supervised Semantic Segmentation Based on Web Image Cosegmentation. In *BMVC*, 2017. 1, 2, 3

[27] T. Shen, G. Lin, C. Shen, and I. Reid. Learning Multi-level Region Consistency with Dense Multi-label Networks for Semantic Segmentation. In *IJCAI*, 2017. 6

[28] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*, 2015. 1, 7

[29] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan. Object Region Mining with Adversarial Erasing: A Simple Classification to Semantic Segmentation Approach. In *CVPR*, 2017. 7

[30] Y. Wei, X. Liang, Y. Chen, X. Shen, M.-M. Cheng, Y. Zhao, and S. Yan. STC: A Simple to Complex Framework for Weakly-supervised Semantic Segmentation. *TPAMI*, 2017. 1, 2, 3, 5, 7, 8

[31] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang. Learning From Massive Noisy Labeled Data for Image Classification. In *CVPR*, 2015. 3

[32] S. Zagoruyko and N. Komodakis. Wide Residual Networks. In *BMVC*, 2016. 1

[33] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid Scene Parsing Network. In *CVPR*, 2017. 1, 3

[34] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization. In *CVPR*, 2016. 3