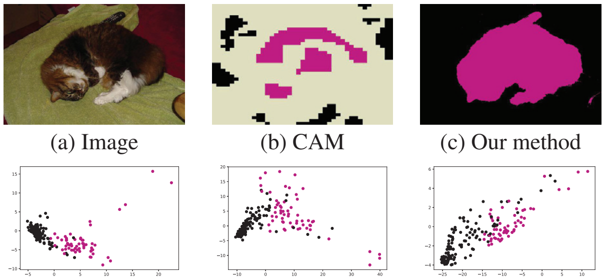
Learning Saliency-Free Model with Generic Features for Weakly-Supervised Semantic Segmentation

Wenfeng Luo,[1] **Meng Yang**[1,2]*
[1]School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China
[2]Key Laboratory of Machine Intelligence and Advanced Computing (SYSU), Ministry of Education
luowf5@mail2.sysu.edu.cn, yangm6@mail.sysu.edu.cn

Abstract

Current weakly-supervised semantic segmentation methods often estimate initial supervision from class activation maps (CAM), which produce sparse discriminative object seeds and rely on image saliency to provide background cues when only class labels are used. To eliminate the demand of extra data for training saliency detector, we propose to discover class pattern inherent in the lower layer convolution features, which are scarcely explored as in previous CAM methods. Specifically, we first project the convolution features into a low-dimension space and then decide on a decision boundary to generate class-agnostic maps for each semantic category that exists in the image. Features from Lower layer are more generic, thus capable of generating proxy ground-truth with more accurate and integral objects. Experiments on the PASCAL VOC 2012 dataset show that the proposed saliency-free method outperforms the previous approaches under the same weakly-supervised setting and achieves superior segmentation results, which are 64.5% on the validation set and 64.6% on the test set concerning mIoU metric.



(a) Image (b) CAM (c) Our method

(e) C41 (f) C42 (g) C43

Figure 1: Comparison of localization ability between CAM (Zhou et al. 2016) and the proposed method. (a) Original images. (b) Object seeds from CAM method. The black regions are pixels with relatively small saliency value. (c) Mask estimated by our method. (e)(f)(g) Convolutional features projected onto 2d planes. Each point denotes a spatial location in the original convolutional features and its color indicates the ground-truth class.

Introduction

Computer vision community has witnessed tremendous progress on the image semantic segmentation problem (Shelhamer, Long, and Darrell 2014; Chen et al. 2016; Zhen et al. 2019) thanks to the successful applications of Deep Convolutional Neural Networks (DCNN). However, training these DCNNs requires large amount of pixel-level annotations, whose collecting procedure is labor-intensive, thus becoming a bottleneck for real-world applications. A promising solution is to develop segmentation methods that could utilize unlabeled or weakly-labeled visual data (Kolesnikov and Lampert 2016; Huang et al. 2018) since they could be acquired in a much faster and cheaper manner.

In this work, we focus on tackling the problem of semantic segmentation under only image-level supervision, since the class information is a more natural supervision and requires the least amount of time for annotation, roughly twenty seconds per image (Russakovsky et al. 2016). Given only the semantic labels of the image, it is very challenging to locate integral object regions since it involves association between high-level semantic and low-level appearance. Many of the previous methods (Huang et al. 2018) relied on class activation maps (CAM) (Zhou et al. 2016) to generate initial supervisions. However, there are two main issues concerning CAM methods: 1) it only localizes sparse discriminative object seeds, as shown in Fig.1(b); 2) it provides no background cues, which are needed for the subsequent segmentation task but are unavailable from the classification network. The latter could be solved by resort to an external saliency detector, whose training also requires extra annotations.

To address the sparseness of initial seeds, many approaches, e.g., adversarial erasing (Wei et al. 2017b; Hou et al. 2018), saliency-aided (Wei et al. 2016; Chaudhry et al. 2017; Oh et al. 2017) and region mining (Huang et al. 2018; Wang et al. 2018), have been proposed for weakly-supervised semantic segmentation. But they all focused on high-level features, which are discriminative enough for classification yet not suitable for dense labeling task. We

*Corresponding Author

instead discover that lower level features could be an alternative cue for finding more integral object regions, as they tend to be more generic and thus are not limited to discriminative areas. Unlike CAM method, the inherent association with the semantic categories is lost when we explore lower level convolution features beyond the ones just before classification layer. Principal Component Analysis (PCA) (Pearson 1901), as demonstrated in (Wei et al. 2017a), exposed in low-dimension space the inner structure between features from foreground and background regions. Figs.1(e), (f), and (g) show in 2d plane the distribution of convolution features from three different layers. The background and foreground features are more (linearly) separable if lower layer features (C41) are adopted.

Based on the above observation, we propose to exploit generic features from lower layers to separate image pixels into background and foreground classes. More specifically, we first learn a normal classification network using the class labels. Then the training images are reorganized into semantic subsets according to their class labels. For a specific subset, we extract the convolution features for each image, which are then projected into low-dimension space to decide on a decision boundary. With some refinements, we could estimate accurate initial masks of integral object, as shown Fig.1(c). After obtaining proxy ground-truth, we simply learn a fully convolutional segmentation network to perform the dense labeling task.

In summary, the main contributions of our work are threefold:

- Instead of working on the CAM (Zhou et al. 2016) from a single image to obtain sparse object seeds, we creatively explore the generic features across images from the same semantic category to discover the underlying structures.
- Our approach is a self-contained and saliency-free segmentation system, thus eliminating the demand of an external saliency detector.
- Experimental results demonstrate that the proposed method achieves state-of-the-art segmentation results under weakly-supervised setting. In particular, it achieves 64.5% and 64.6% mIoU scores on val and test set of PASCAL VOC 2012 dataset.

Related Works

In this section, we introduce image co-localization and weakly-supervised semantic segmentation methods which are related to our work.

Image Co-Localization

Image co-localization addresses the problem of simultaneously localizing some common object across a set of unlabeled images. Among the popular approaches, Deep Descriptor Transformation (DDT) (Wei et al. 2017a) provided great insights into the feature correlations across images via Principal Component Analysis. By projecting the convolution features onto the first component, DDT revealed the positions of common objects. DDT was designed for co-localizing common objects, namely predicting the boxes surrounding them, so it can not provide precise class assignment to individual pixels. Besides, co-localization has a strong assumption that every image contains object from a single category, so DDT is unable to handle multi-class images. Instead, we discover that feature depth matters when performing dense labeling tasks and thus propose to explore the inner structure of convolution features from lower network layer.

Weakly-Supervised Semantic Segmentation

Here we mainly review weakly-supervised segmentation methods under image-level labels, which share the same weak supervision with our method. Many of the weakly-supervised methods utilized classification network to generate initial seed cues, which then supervised the training of segmentation networks. Seed, expand and constrain (SEC) (Kolesnikov and Lampert 2016) proposed to localize object seeds using CAM (Zhou et al. 2016), which were discriminative for classification yet only provided sparse object-related seeds. To address this problem, subsequent methods introduced either "static" or "dynamic" expansion mechanisms to discover more foreground pixels.

Dynamic expansion mechanisms start with sparse initial supervisions and try to update them with more object-related pixels along with the training of segmentation networks. Wang et al. (Wang et al. 2018) proposed to mine common object features between the initial seeds and undetected foreground pixels through a region classification network, which was iteratively updated along with the segmentation network. Deep Seeded Region Growing (Huang et al. 2018) proposed to expand the initial seeds to neighboring pixels using high-level convolution features to compute pixel similarity. FickleNet (Lee et al. 2019) made a major improvement over DSRG on obtaining more accurate seed cues via a stochastic feature selection layer, whose computation though consumed more GPU memory during inference.

In contrast, static mechanisms find as many object-related pixels as possible beforehand, so the estimated supervision stays the same during the training of segmentation network. Some methods (Wei et al. 2016; Chaudhry et al. 2017; Oh et al. 2017) harnessed image saliency as heuristic cues, which might be inaccurate and tended to be cluttered under multi-class scenes. Recently, adversarial erasing (Wei et al. 2017b) was proposed to suppress initially stimulated regions by erasing associated pixels. However, it required retraining of the network after each erasing. Self-erasing network (Hou et al. 2018) shifted the erasing operation to the high-level convolution features and prevented attentions spreading to the background by conditionally reversing signs of feature activations. Multi-dilated convolutions (MDC) (Wei et al. 2018) proposed to adopt convolution kernels of varying dilation rates to enlarge receptive field of DCNN, thus promoting the emergence of non-discriminative object regions. However, overlarge receptive fields tended to downgrade the localization ability on small objects.

It is worth noting that CAM provided no background cues for the subsequent segmentation task, so the CAM-based methods, such as DSRG (Huang et al. 2018), MDC (Wei et

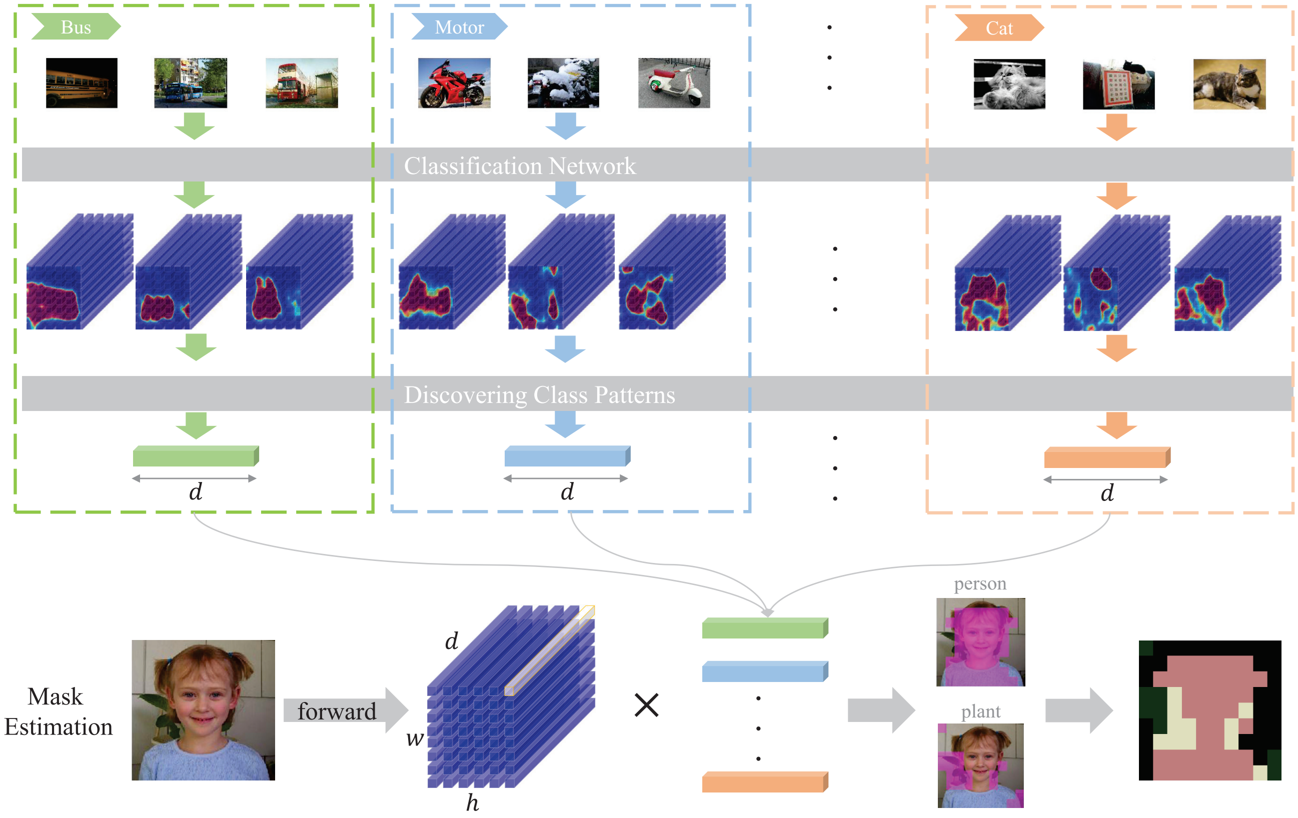


Figure 2: The overall architecture of the proposed method. Training images are first organized into different subsets based on class labels. Images from each subset are passed through classification network to extract convolution features, which are forwarded to the Discovering Class Patterns module to output a d-dimensional vector. During mask estimation, we perform inner product between the convolution features and the class patterns to generate segmentation masks for each class present in the image, which are then combined to yield the final mask estimation.

al. 2018), FickleNet (Lee et al. 2019), etc., required an external saliency detector to extract background pixels. It is undesirable since extra annotations are needed to train a saliency detector.

Methods

As aforementioned, CAM produces sparse object seeds and relies on external saliency detector to provide background cues. To handle the spareness, we here propose to explore more generic features from lower network layer, other than those discriminative ones for classification. Specifically, the training images are reorganized into semantic subsets according to their class labels. The generic features from the same semantic subsets are used to Discover Class Patterns (DCP), which serves as a per-pixel classifier to decide whether an individual pixel is background or foreground, thus naturally eliminating the usage of an external saliency detector. Fig.2 gives an overview of the proposed approach. The upper part of Fig.2 denotes the DCP module where training images are reorganized into semantic subsets to extract class pattern inherent in the generic features. Then we compute the correlation between the feature and class patterns at each spatial location to estimate the initial supervision, as shown in the lower part of Fig.2. Before diving into the technical details, the following section introduces the intuition behind our approach.

Observation

Principal component analysis (PCA) (Pearson 1901), widely used for dimension reduction, is mathematically defined as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by some projection of the data comes to lie on the first coordinate, the second greatest variance on the second coordinate, and so on. As shown in Fig.1, there is a clear distinction between foreground and background features, especially from

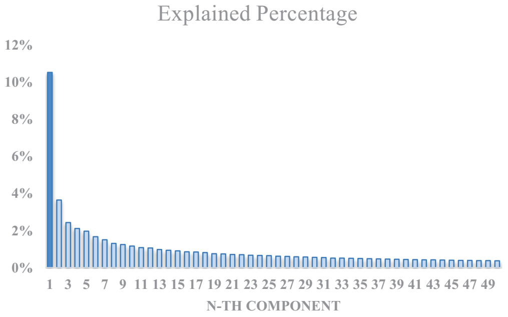


Figure 3: Percentage of variance explained by the first 50 components for the *dog* class.

lower network layer. But how does one decide on a decision boundary to separate them? Fig.3 shows the percentage of variance explained by the first 50 components for the *dog* class. We can see that the first component contributes over 10% for the variance, but it drops dramatically down to less than 4% for the second component. Therefore, the first component serves as a representative *class pattern* as it captures the most feature variation. Features with positive correlation with the class pattern are more likely to be associated with foreground pixels and vice versa, which indicates the origin serves a good decision boundary. The following section presents the calculation of class pattern for each semantic category.

Discovering Class Pattern

We denote the set of training data as $\{(X_1, y_1), (X_2, y_2), ..., (X_m, y_m)\}$, where the class labels y_i belong to a set of foreground categories $\mathcal{C}$. We obtain a feature extractor $f(X; \theta)$, parameterized by θ, from classification CNN by dropping the last few layers. For brevity we simply write it as $f(X)$ and the associated parameter is determined by the context.

Based on the image-level labels, we first reorganize the training images into semantic subsets $\{\mathcal{I}^c | c \in \mathcal{C}\}$, where $\mathcal{I}^c$ denotes the set of images in which class c exists. Notably, some images contain more than one classes and thus will be divided into several semantic subsets. Inside each subsets $\{X^1, X^2, ..., X^{|\mathcal{I}^c|}\}$, we forward each image through the classification network up to a desired layer to obtain the features:

$$T^i = f(X^i), T^i \in R^{h \times w \times d} \tag{1}$$

The pixel descriptor at each spatial location is denoted as $t_u^i \in R^d$, where u loops through the spatial dimension. Then the mean vector and covariance matrix could be calculated across all pixel locations and distinct images:

$$\begin{aligned} \bar{t}_c &= \frac{1}{K} \sum_i \sum_u t_u^i \\ \Sigma_c &= \frac{1}{K} \sum_i \sum_u (t_u^i - \bar{\mathrm{t}}_c)(t_u^i - \bar{t}_c)^\intercal \end{aligned} \tag{2}$$

where K is the total number of spatial locations. For images with the same input dimension, K equals to $|\mathcal{I}^c| \times h \times w$.

We then perform eigendecomposition on the covariance matrix to obtain the eigenvector ξ_c associated with the largest eigenvalue. Repeat the preceding procedure for each class and we could obtain all the class patterns $\{\xi_c | c \in \mathcal{C}\}$. Note that there is no class pattern associated the background class.

Per-Pixel Mask Estimation

The class patterns could serve as a per-pixel classifier when we project the original convolution features onto them to see the correlation with each semantic categories:

$$q_c = (t_u - \bar{t}_c)^\intercal \xi_c \tag{3}$$

where q_c is essentially the value of the first principal component and a positive q_c denotes that pixel u belongs to class c. Since some pixels might have positive correlation with several classes, we come up with the following mechanism to handle conflicts. For a specific spatial location u, there are three common cases:

- Case 1: It does not have positive correlation with any of the class patterns and is assigned to background class c^{bg};
- Case 2: It has positive correlation with only one of the class patterns ξ_c and is assigned to a foreground category c;
- Case 3: It has positive correlation with more than one of the class patterns and is treated as uncertain $c^{uncertain}$, which is ignored during training.

We also perform fully-connected CRF as post refinement only for the single-class images since multi-class images tend to be more cluttered. Algorithm 1 summarizes the procedure for estimating the proxy ground-truth for one image.

Generic vs. Discriminative Features

Here we visualize the estimated masks to further verify our assumption that feature map matters for weakly supervised semantic segmentation. To this end, we consider the features from three different blocks in the last stage (C4) of ResNet-101, namely C41, C42 and C43. Block C43 is closest to the classification layer and thus is discriminative enough to support decision making. Fig.4 shows some estimated masks for the training images using features from different network layers. As could be seen in the first two rows, the masks from C43 are not as complete as those from C41. This coarse estimation from block C43 suffices for object detection since we could find the smallest surrounding box. However, it is harmful for the dense labeling task. We argue that lower residual block yields more generic features and thus not limit itself to the most discriminative regions, like the front of the car and head of the bird. Another notable phenomenon is that C41 seems to generate more compact object boundary, while mask from C43 tends to be coarse and overlarge, which could easily be seen in the third row. It could be because block C43 has relatively larger receptive

```
Algorithm 1: Estimating proxy ground-truth
Input: Image $X$, class labels $y$ and class patterns
       $\{\xi_c | c \in \mathcal{C}\}$
Extract the activation $T = f(X) \in R^{h \times w \times d}$;
Initialize $M = zeros(n)$, $Q = zeros(n, |y|)$, where n
  equals to $h \times w$;
for position $u$ in $\{1, 2, ..., n\}$ do
    for class $c$ in $y$ do
        $q_c = (t_u - \bar{t}_c)^\intercal \xi_c$;
        $Q(u, c) = I(q_c > 0)$;// $I(\cdot)$ is the
            indicator function
    end
end
for position $u$ in $\{1, 2, ..., n\}$ do
    $s = \sum_c Q(u, c)$;
    if $s = 0$ then
        // case 1
        $M(u) = c^{bg}$;
    else if $s = 1$ then
        // case 2
        $M(u) = \arg\max_c Q(u, c)$;
    else
        // case 3
        $M(u) = c^{uncertain}$;
    end
end
Output: Estimated mask supervision $M$
```

field. The last two rows demonstrate the masks for multi-class images. More thorough comparison is presented in the experimental section.

Experiments

Experimental Setup

Dataset and Evaluation Metric The proposed method is evaluated on two benchmark datasets, PASCAL VOC 2012 (Everingham et al. 2012) and COCO (Lin et al. 2014). **PASCAL VOC**: It consists of 20 foreground classes plus one background class. As a common practice in (Kolesnikov and Lampert 2016; Huang et al. 2018), we also include the extra annotations by (Hariharan et al. 2011) in addition to the officially provided *train* set (1,464 images) and end up with a *trainaug* set with 10,582 images. We report the mean intersection over union (mIoU) on both *val* and *test* set. **COCO**: We use the train-val split setting of competition in 2017, where 112k images are used for training and the remaining 5k are reserved for evaluation. We report the same mIoU metric in the 5k validation images over 81 semantic categories.

Training and Testing Setting We use ResNet-101 for classification network, which is initialized by parameters pre-trained on the ImageNet (Deng et al. 2009). For the segmentation network, we adopt the DeepLab-CRF-LargeFOV model (Chen et al. 2016). For fair comparison, both VGG16 and ResNet101 are adopted as backbone for evaluation. We use Adam optimizer (Kingma and Ba 2014) with a learning rate of 5e-6 for the backbone and 1e-4 for the randomly initialized layers. With a batch size of 16, we train the segmentation network for 20 epochs. During training, the image batches are resized to fixed dimension of 328 × 328.

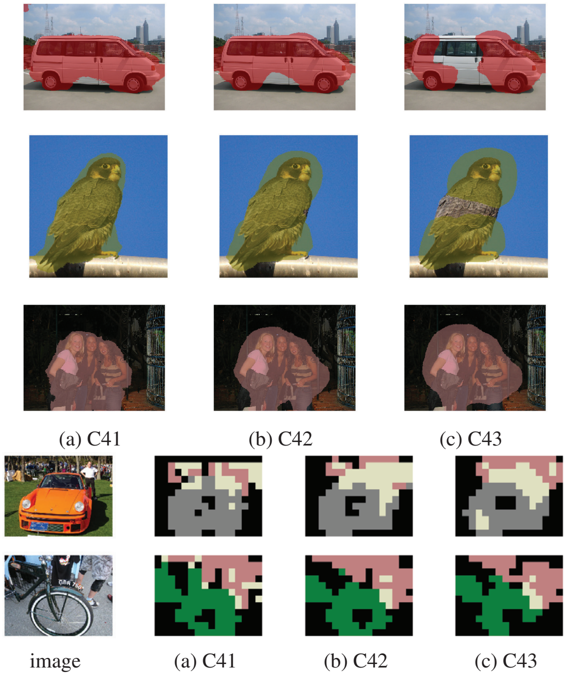

(a) C41 (b) C42 (c) C43

image (a) C41 (b) C42 (c) C43

Figure 4: Demonstration of training images. The estimated mask of each column comes from different convolution layers, namely C41, C42 and C43.

In the test phase, we adopt multi-scale testing with input dimensions of 241, 328 and 401. The test image is first forwarded through the segmentation network and the scores from different input dimensions are aggregated by taking the average. As common practice in (Kolesnikov and Lampert 2016; Huang et al. 2018), we apply a fully-connected CRF (Krähenbühl and Koltun 2011) as post refinement.

Ablation Study

In this section, we conduct experiments to provide more insights into the proposed method. All the comparison experiments are done on the PASCAL VOC dataset. In order to verify our assumption, we conduct experiments concerning different network features while generating proxy ground-truth for the segmentation network. We find that there is no need to use all the images from a semantic subset. In our experiments, at most 800 images for each subset are chosen at random to estimate the covariance matrix. The extracted class patterns are nearly identical to the one from all images. Using fewer images saves a lot of the computation and also avoids numerical instability. The overall results are shown in table 1.

The first three rows in Table 1 demonstrate that the segmentation performance increases dramatically when using features from lower residual blocks. Block C43 is right next

Table 1: Segmentation results on PASCAL VOC 2012 *val* set under different experimental settings of our method

Features	bg	plane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	mIoU
C43	74.9	33.9	25.0	37.9	29.4	47.3	68.9	59.9	58.1	23.0	50.3	34.0	54.1	47.5	56.7	48.8	41.3	48.3	27.9	51.5	48.0	46.0
C42	80.8	43.0	31.0	49.0	32.7	57.0	74.0	67.3	64.2	25.0	57.4	32.5	60.7	61.4	66.3	55.2	44.7	57.9	30.5	56.0	52.7	52.4
C41	86.1	53.7	33.8	66.1	40.2	64.2	82.2	73.4	82.4	24.6	71.1	30.6	77.3	72.8	72.1	66.9	45.3	70.3	32.5	67.4	53.1	60.3
C41 + Retrain	88.6	64.1	35.4	78.8	50.8	61.0	85.8	77.7	84.6	26.7	75.2	40.8	79.1	77.4	76.0	70.4	48.3	69.2	39.0	69.9	58.3	64.5

to the classification layer and achieves accuracy of 46.0%, close to the result 45.4% (Kolesnikov and Lampert 2016) that was obtained by training on the object seeds from CAM. Going lower to block C42 improves the performance by 6.4%. Block C41 achieves the highest accuracy of 60.3%, up from 52.4% as in C42. This coincides with our intuition that lower convolutional layers yield more generic features and thus provides more integral and compact object masks as proxy ground-truth.

Since there are uncertain pixels $c^{uncertain}$ for multi-class images due to the design of algorithm 1, we would like to predict the proxy ground-truth for those uncertain pixels by making use of the trained model. To be specific, we retrain the segmentation model from scratch using the segmentation masks predicted by the trained model ”C41”. The retraining improves the final result from 60.3% to 64.5%. Further retraining brings negligible improvement so we decide to perform it once.

Comparison with State-of-arts

The segmentation results on PASCAL VOC 2012 are presented in table 2. All the results for comparison were obtained using the VGG16 backbone unless specified otherwise. We compare our method with the previous state-of-art weakly-supervised segmentation approaches. Some of the methods are only provided for reference since they used stronger supervisions including scribbles (Lin et al. 2016), bounding boxes (Dai, He, and Sun 2015; Song et al. 2019) and image saliency (Wei et al. 2016; 2017b; Chaudhry et al. 2017; Wang et al. 2018). Besides, both WebS-i2 (Jin et al. 2017) and Hong *et al*. (Hong et al. 2017) used extra webly-crawled images (19k and 900k), whose collection might take extra efforts.

Results in table 2 show that our method achieves state-of-art segmentation performance and outperforms all the previous approaches by a significant margin. Among the techniques that use only class labels, DSRG (Huang et al. 2018) and FickleNet (Lee et al. 2019) achieve the best performance. However, as MCOF (Wang et al. 2018) and MDC (Wei et al. 2018), both DSRG and FickleNet utilize similar CAM techniques to extract foreground object seeds and thus the background cues are extracted from an external saliency detector, whose training requires extra pixel-level annotations. Our approach directly estimate the background regions from the classification network, hence eliminating the need for an external saliency detector. AffinityNet (Ahn and Kwak 2018) also avoided the usage of saliency prior but an addition network was learned for semantic affinities. In contrast, our method makes full use of the classification network and requires no additional network training. With the refinement of CRF and ResNet101 backbone, our method achieves mIoU results of 64.5% on *val* set and 64.6% on the *test* set. The result on *test* set is available on the website (http://host.robots.ox.ac.uk:8080/anonymous/USWTK1.html).

Table 2: Segmentation results of different methods on PASCAL VOC 2012 *val* and *test* Set.

Methods	Train Set	Val	Test
Supervision: Scribbles			
Scribblesup (Lin et al. 2016)	10k	63.1	-
Supervision: Box			
BoxSup (Dai, He, and Sun 2015)	10k	62.0	64.2
Box-driven (Song et al. 2019)	10k	66.8	-
Supervision: Class labels			
STC (Wei et al. 2016)	10k + sal	49.8	51.2
WebS-i2 (Jin et al. 2017)	19k	53.4	55.3
AE-PSL (Wei et al. 2017b)	10k + sal	55.0	55.7
MCOF (Wang et al. 2018)	10k + sal	56.2	57.6
Hong *et al*. (Hong et al. 2017)	10k + video	58.1	58.7
DCSP (Chaudhry et al. 2017)	10k + sal	58.6	59.2
EM-Adapt (Papandreou et al. 2015)	10k	38.2	39.6
DCSM (Shimoda and Yanai 2016)	10k	44.1	45.1
SEC (Kolesnikov and Lampert 2016)	10k	50.7	51.7
Multi-Cues (Roy and Todorovic 2017)	10k	52.8	53.7
DSRG (Huang et al. 2018)	10k	59.0	60.4
AffinityNet (Ahn and Kwak 2018)	10k	58.4	60.5
SeeNet (Hou et al. 2018)	10k	61.1	60.7
MDC (Wei et al. 2018)	10k	60.4	60.8
FickleNet (Lee et al. 2019)	10k	61.2	61.9
Ours (VGG16)	10k	61.2	-
Ours (ResNet101)	10k	**64.5**	**64.6**

Qualitative Results

Fiugure 5 shows some qualitative sample images from PASCAL VOC 2012 *val* set. Our method can produce decent segmentation mask which maintains low-level object boundary. As shown in the first two rows, it perfectly predicts the integral object regions. For more complex scenes involving multiple classes, our approach still performs very well in locating different objects, which further verifies the effectiveness of the proposed method.

Results on COCO

To verify the practicability of our method, we conduct further experiment on Microsoft COCO dataset, which contains a lot more semantic categories (81) and images (118k), thus posing a challenge for current weakly-supervised methods. We first learn a classification network using the available class labels and achieve a classification accuracy of 80.4%. As aforementioned, we only use at most 2000 images from each semantic subset to compute the class patterns since there are more images in each semantic category concerning

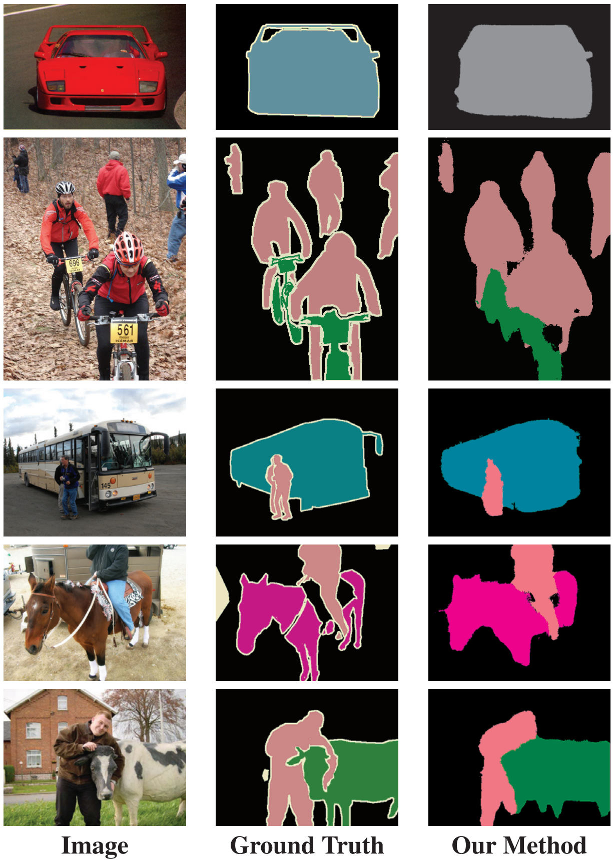


Figure 5: Demonstration of sample images. Left column: The original image. Middle column: ground truth mask. Right column: segmentation result from our method.

COCO. We compare our method with two other popular approaches, SEC (Kolesnikov and Lampert 2016) and DSRG (Huang et al. 2018). Per-class IOUs are shown in table 3. Our approach seems to perform much better on several supercategories, such as *Food*, *Appliance*, *Indoor*, and outperforms DSRG by 3.9% on *val* set on aggregate. One notable problem on COCO dataset is that weakly-supervised methods fail to detect several classes, such as *fork* and *hair dryer*. More investigation is needed to address the problem of detecting small objects.

Table 3: Per-class IoU on COCO *val* set.

Cat.	Class	SEC	DSRG	Ours	Cat.	Class	SEC	DSRG	Ours
BG	background	74.3	80.6	73.9	Kitchenware	wine glass	22.3	24.0	27.2
P	person	43.6		48.7		cup	17.9	20.4	21.7
Vehicle	bicycle	24.2	30.4	45.0		fork	1.8	0.0	0.0
	car	15.9	22.1	31.5		knife	1.4	5.0	0.9
	motocycle	52.1	54.2	59.1		spoon	0.6	0.5	0.0
	airplane	36.6	45.2	26.9		bowl	12.5	18.8	7.6
	bus	37.7	38.7	52.4	Food	banana	43.6	46.4	52.0
	train	30.1	33.2	42.4		apple	23.6	24.3	28.8
	truck	24.1	25.9	36.9		sandwich	22.8	24.5	37.4
	boat	17.3	20.6	23.5		orange	44.3	41.2	52.0
Outdoor	traffic light	16.7	16.1	13.3		broccoli	36.8	35.7	33.7
	fire hydrant	55.9	60.4	45.1		carrot	6.7	15.3	29.0
	stop sign	48.4	51.0	43.4		hot dog	31.2	24.9	38.8
	parking meter	25.2	26.3	33.5		pizza	50.9	56.2	69.8
	bench	16.4	22.3	26.3		donut	32.8	34.2	50.8
Animal	bird	34.7	41.5	29.9		cake	12.0	6.9	37.3
	cat	57.2	62.2	62.1	Furniture	chair	7.8	9.7	10.7
	dog	45.2	55.6	57.5		couch	5.6	17.7	9.4
	horse	34.4	42.3	40.7		potted plant	6.2	14.3	21.8
	sheep	40.3	47.1	54.0		bed	23.4	32.4	34.6
	cow	41.4	49.3	47.2		dining table	0.0	3.8	1.1
	elephant	62.9	67.1	64.3		toilet	38.5	43.6	43.8
	bear	59.1	62.6	58.9	Electronics	tv	19.2	25.3	11.5
	zebra	59.8	63.2	60.7		laptop	20.1	21.1	37.0
	giraffe	48.8	54.3	45.1		mouse	3.5	0.9	0.0
Accessory	backpack	0.3	0.2	0.0		remote	17.5	20.6	37.2
	umbrella	26.0	35.3	46.1		keyboard	12.5	12.3	19.0
	handbag	0.5	0.7	0.0		cell phone	32.1	33.0	38.1
	tie	6.5	7.0	15.5	Appliance	microwave	8.2	11.2	43.4
	suitcase	16.7	23.4	43.6		oven	13.7	12.4	29.2
Sport	frisbee	12.3	13.0	23.2		toaster	0.0	0.0	0.0
	skis	1.6	1.5	6.5		sink	10.8	17.8	28.5
	snowboard	5.3	16.3	10.9		refrigerator	4.0	15.5	23.8
	sports ball	7.9	9.8	0.6	Indoor	book	0.4	12.3	26.3
	kite	9.1	17.4	14.0		clock	17.8	20.7	13.4
	baseball bat	1.0	4.8	0.0		vase	18.4	23.9	27.1
	baseball glove	0.6	1.2	0.0		scissors	16.5	17.3	37.0
	skateboard	7.1	14.4	7.6		teddy bear	47.0	46.3	58.9
	surfboard	7.7	13.5	17.6		hair dryer	0.0	0.0	0.0
	tennis racket	9.1	6.8	38.1		toothbrush	2.8	2.0	11.1
	bottle	13.2	22.3	28.4		**mean IoU**	22.4	26.0	29.9

Conclusion

In this paper, we address two main issues of the classic Class Activation Map (CAM) for weakly-supervised semantic segmentation under image-level supervision. To overcome the sparseness of the object seeds, we propose to explore convolution features from lower network layer, which is more generic for dense labeling task since it is not directly involved in classification. The generic features are extracted from a semantic subset to discover class pattern, which serves as a per-pixel classifier to generate the initial supervision. The experimental results show that the proposed method achieves state-of-art segmentation results on benchmark datasets. In future work, we would focus on extending the current framework for more challenging tasks, like instance segmentation.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (Grant no.61772568), the Natural Science Foundation of Guangdong province (Grant no.2019A1515012029), the Guangzhou Science and Technology Program (Grant no. 201804010288), and the Fundamental Research Funds for the Central Universities (Grant no.18lgzd15).

References

Ahn, J., and Kwak, S. 2018. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In *CVPR*.

Chaudhry, A.; Dokania, P. K.; Torr, P.; and Toor, P. 2017. Discovering class-specific pixels for weakly-supervised semantic segmentation. In *BMVC*, volume abs/1707.05821.

Chen, L.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2016. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI* 40:834–848.

Dai, J.; He, K.; and Sun, J. 2015. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. *ICCV* 1635–1643.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-

Fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Everingham, M.; Van Gool, L.; Williams, C. K. I.; Winn, J.; and Zisserman, A. 2012. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results.

Hariharan, B.; Arbelaez, P.; Bourdev, L.; Maji, S.; and Malik, J. 2011. Semantic contours from inverse detectors. In *ICCV*.

Hong, S.; Yeo, D.; Kwak, S.; Lee, H.; and Han, B. 2017. Weakly supervised semantic segmentation using web-crawled videos. *CVPR* 2224–2232.

Hou, Q.; Jiang, P.; Wei, Y.; and Cheng, M. 2018. Self-erasing network for integral object attention. In *NIPs*.

Huang, Z.; Wang, X.; Wang, J.; Liu, W.; and Wang, J. 2018. Weakly-supervised semantic segmentation network with deep seeded region growing. In *CVPR*.

Jin, B.; Segovia, M. V. O.; Süsstrunk, S.; and Süsstrunk, S. 2017. Webly supervised semantic segmentation. In *CVPR*, 1705–1714.

Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. In *ICLR*, volume abs/1412.6980.

Kolesnikov, A., and Lampert, C. 2016. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *ECCV*, volume abs/1603.06098.

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

Lee, J.; Kim, E.; Lee, S.; Lee, J.; and Yoon, S. 2019. Ficklenet: Weakly and semi-supervised semantic image segmentation using stochastic inference. In *CVPR*.

Lin, T.; Maire, M.; Belongie, S.; Bourdev, L.; Girshick, R.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Lin, D.; Dai, J.; Jia, J.; He, K.; and Sun, J. 2016. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. *CVPR* 3159–3167.

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

Papandreou, G.; Chen, L.; Murphy, K. P.; and Yuille, A. L. 2015. Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *ICCV*, 1742–1750.

Pearson, K. 1901. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2(11):559–572.

Roy, A., and Todorovic, S. 2017. Combining bottom-up, top-down, and smoothness cues for weakly supervised image segmentation. In *CVPR*.

Russakovsky, O.; Bearman, A. L.; Ferrari, V.; and Li, F. 2016. What's the point: Semantic segmentation with point supervision. In *ECCV*.

Shelhamer, E.; Long, J.; and Darrell, T. 2014. Fully convolutional networks for semantic segmentation. *CVPR* 3431–3440.

Shimoda, W., and Yanai, K. 2016. Distinct class-specific saliency maps for weakly supervised semantic segmentation. In *ECCV*.

Song, C.; Huang, Y.; Ouyang, W.; and Wang, L. 2019. Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. In *CVPR*.

Wang, X.; You, S.; Li, X.; and Ma, H. 2018. Weakly-supervised semantic segmentation by iteratively mining common object features. In *CVPR*.

Wei, Y.; Liang, X.; Chen, Y.; Shen, X.; Cheng, M.; Feng, J.; Zhao, Y.; and Yan, S. 2016. Stc: A simple to complex framework for weakly-supervised semantic segmentation. *TPAMI* 39:2314–2320.

Wei, X.; Zhang, C.; Li, Y.; Xie, C.; Wu, J.; Shen, C.; and Zhou, Z. 2017a. Deep descriptor transforming for image co-localization. In *IJCAI*.

Wei, Y.; Feng, J.; Liang, X.; Cheng, M.-M.; Zhao, Y.; and Yan, S. 2017b. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Wei, Y.; Xiao, H.; Shi, H.; Jie, Z.; Feng, J.; and Huang, T. 2018. Revisiting dilated convolution: A simple approach for weakly- and semi-supervised semantic segmentation. In *CVPR*.

Zhen, M.; Wang, J.; Zhou, L.; Fang, T.; and Quan, L. 2019. Learning fully dense neural networks for image semantic segmentation. *AAAI* abs/1905.08929.

Zhou, B.; Khosla, A.; A., L.; Oliva, A.; and Torralba, A. 2016. Learning Deep Features for Discriminative Localization. *CVPR*.