Distinct Class-specific Saliency Maps for Weakly Supervised Semantic Segmentation

Wataru Shimoda Keiji Yanai

Department of Informatics, The University of Electro-Communications, Tokyo
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585 JAPAN
{shimoda-k,yanai}@mm.inf.uec.ac.jp

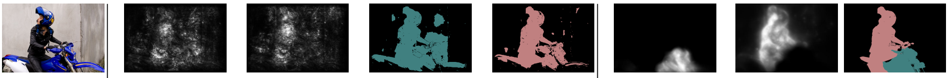
Abstract. In this paper, we deal with a weakly supervised semantic segmentation problem where only training images with image-level labels are available. We propose a weakly supervised semantic segmentation method which is based on CNN-based class-specific saliency maps and fully-connected CRF. To obtain distinct class-specific saliency maps which can be used as unary potentials of CRF, we propose a novel method to estimate class saliency maps which improves the method proposed by Simonyan et al. [1] significantly by the following improvements: (1) using CNN derivatives with respect to feature maps of the intermediate convolutional layers with up-sampling instead of an input image; (2) subtracting the saliency maps of the other classes from the saliency maps of the target class to differentiate target objects from other objects; (3) aggregating multiple-scale class saliency maps to compensate lower resolution of the feature maps. After obtaining distinct class saliency maps, we apply fully-connected CRF [2] by using the class maps as unary potentials. By the experiments, we show that the proposed method has outperformed state-of-the-art results with the PASCAL VOC 2012 dataset under the weakly-supervised setting.

Keywords: semantic segmentation, weakly supervised segmentation, fully convolutional neural network, fully connected CRF

1 Introduction

Due to the recent advent of deep learning methods, convolutional neural network (CNN) based methods have outperformed most of the previous state-of-the-art in various kinds of image recognition tasks. In the task of semantic segmentation, CNN achieved about 50% improvement [3,4]. Semantic image segmentation is a task to add object class labels to each of all the pixels in a given image, which is more challenging task than object classification and object detection. Semantic segmentation is expected to contribute detailed analysis of images in various practical tasks such as food calorie estimation [5,6]. However, most of the CNN based semantic segmentation methods assume that pixel-wise annotation is available, which is costly to obtain in general.

On the other hand, collecting images with image-level annotation is easier than those with pixel-level annotation, since many images attached with tags



(A) sample image | Simonyan et al. [1] (B) motorbike (C) person | GrabCut (D) motorbike (E) person | our saliency maps (F) motorbike (G) person | (H) CRF result

Fig. 1. (From the left) (A)sample image, (B)(C)its class saliency maps with respect to "motorbike" and "person" by [1], (D)(E)estimated regions of them by GrabCut, (F)(G)class saliency maps by the proposed method, and (H) estimated regions by Dense CRF.

are available on hand-crafted open image datasets such as ImageNet as well as on the Web. In this work, we focus on weakly-supervised semantic segmentation which requires not pixel-wise annotation as well as bounding box annotation but only image-level annotation.

In this paper, we propose a weakly supervised semantic segmentation method which is based on CNN-based class saliency maps and fully-connected CRF [2]. To obtain class saliency maps which are so distinct that we can use them as unary potentials terms of CRF, we propose a novel method to estimate class saliency maps which improves the method proposed by Simonyan et al. [1] significantly. Simonyan et al. [1] proposed class saliency maps based on the gradient of the class score with respect to the input image, which showed weakly-supervised object localization could be done by back-propagation-based visualization. However, their class saliency maps are vague and not distinct (Fig.1(B)(C)). In addition, when different kinds of target objects are included in the image, the maps tend to respond to all the object regions. Although they adopted GrabCut for weakly-supervised segmentation based class saliency maps in their paper, their method is unable to distinguish multiple object regions (Fig.1(D)(E)). To resolve the weaknesses of their method, we propose a new method to generate CNN-derivatives-based saliency maps. The proposed method can generate more distinct class saliency maps which discriminate the regions of a target class from the regions of the other classes (Fig.1(F)(G)). The generated maps are so distinct that they can be used as unary potentials of CRF as they are (Fig.1(H)). We call our new method for generating class saliency maps as "Distinct Class Saliency Maps (DCSM)".

To obtain DCSM, we propose three improvements over Simonyan et al. [1]: (1) using CNN derivatives with respect to feature maps of the intermediate convolutional layers with up-sampling instead of an input image; (2) subtracting the saliency maps of the other classes from the saliency maps of the target class to differentiate target objects from other objects; (3) aggregating multiple-scale class saliency maps to compensate lower resolution of the feature maps. After obtaining distinct class saliency maps, we apply fully-connected CRF [2] by using the class maps as unary potentials. As a CNN, we use the VGG-16 pre-trained with 1000-class ILSVRC datasets and fine-tune it with multi-class training using only image-level labeled dataset. By the experiments, we show that the proposed method has outperformed state-of-the-arts on the PASCAL

VOC 2012 dataset in the task of weakly supervised semantic segmentation under the standard condition.

To summarize our contributions in this paper, they are as follows:

- We propose a new weakly supervised segmentation method which combines distinct class saliency maps (DCSM) and fully connected CRF.
- We propose a novel method to estimate distinct class saliency maps:
  - based on CNN derivatives with respect to feature maps of the intermediate convolutional layers.
  - subtracting class saliency maps from each other.
  - aggregating multiple-scale class saliency maps.
- The obtained result outperforms those by the current state-of-the-arts on the Pascal VOC 2012 segmentation dataset under the weakly supervised setting.

2 Related Work

Recently, CNN-based semantic segmentation are being explored very actively, and the accuracy has been significantly improved compared to the non-CNN-based conventional methods. In this section, first we describe fully-supervised semantic segmentation , and next we explain weakly-supervised segmentation problem which is addressed in this work. Finally we describe some works based on gradient-based class saliency detection.

2.1 CNN-based Fully-Supervised Semantic Segmentation

As early works on CNN-based semantic segmentation, Girshick et al. [7] and Hariharan et al. [8] proposed an object segmentation method using region proposal and CNN-based image classification. Firstly, they generated 2000 region candidates at most by Selective Search [9], and secondly apply CNN image classification by feed-forwarding of the CNN to each of the proposals. Finally they integrated all the classification results by non-maximum suppression and generated the final object regions. Although these methods outperformed the conventional methods greatly, they had a drawback that they required long processing time for CNN-based image classification of many region proposals.

While Girshick et al. [7] and Hariharan et al. [8] took advantage of excellent ability of a CNN on image classification task for semantic image segmentation in a relatively straightforward way, Long et al. [10] and Mostajabi et al. [11] proposed CNN-based semantic segmentation in a hierarchical way which achieved more robust and accurate segmentation. A CNN is much different from conventional bag-of-features framework regarding multi-layered structure consisting of multiple convolutional and pooling layers. Because CNN has several pooling layers, location information is gradually losing as the signal is transmitting from the lower layers to the upper layers. In general, the lower layers hold location information in their activations, while the upper layers holds weak local information. Therefore, it is difficult to estimate object regions by using only information in the last output layer. Long et al. [10] and Mostajabi et al. [11] pointed out that it can complement spatial information in the upper activations by up-sampling of the information in the middle layers and integrating them with the information in the upper layers. Upsampling in a CNN is generally called as "Deconvolution".

Long et al. [10] proposed a CNN-based segmentation method which integrates deconvolution from the intermediate layers and object heat map obtained by replacing all the full connection layers with 1×1 convolutional layers and providing a larger-size image than a usual 256×256 image. This replaces class score vectors with class score maps as outputs of the CNN, which express rough location of objects [12]. This idea was originally proposed by Sermanet et al. [13] and called as "fully convolutional network" or "sliding CNN", which plays important roles to raise performance on CNN-based segmentation. By using larger-size images as input images, more detailed location information can be obtained in the intermediate layers as well as in the class score maps from the last layer. This can be used as unary potentials of CRF [3,4,14].

On the other hand, Mostajabi et al. [11] proposed a method which associates up-sampled activation features of several intermediate layers with super-pixels and treat them as local features, which are called "zoom-out features".

In our work, we also aggregate location information in multiple intermediate layers for image segmentation. However, we adopt a back-propagation-based method, while they adopted feed-forward image segmentation.

2.2 CNN-based Weakly-Supervised Segmentation

Most of the conventional non-CNN-based weakly supervised segmentation method employed Conditional Random Field (CRF) with unary potentials estimated by multiple instance learning [15], extremely randomized hashing forest [16], and GMM [17].

As a CNN-based method, Pedro et al. [18] addressed weakly-supervised segmentation by using multi-scale CNN proposed in [13]. They integrated the outputs which contain location information with log sum exponential, and limited object regions to the regions overlapped with object proposals [19].

Pathak et al. [20,21] and Papandreou et al. [22] proposed weakly-supervised semantic segmentation by adapting CNN models for fully-supervised segmentation to weakly-supervised segmentation. In MIL-FCN [20], they trained the CNN for fully-supervised segmentation proposed in Long et al. [10] with a global max-pooling loss which enables training of the CNN model using only training data with image-level labels which is the same idea as Multiple Instance Learning. Constrained Convolutional Neural Network (CCNN) [21] improved MIL-FCN by adding some constraints and using fully-connected CRF [2]. Papandreou et al. [22] trained the DeepLab model [3] proposed as a fully-supervised model with EM algorithm, which is called as "EM-adopt". Both CCNN and EM-adopt generated pseudo-pixel-level labels from image-level labels using constraints and EM algorithms to train FCN and DeepLab which were originally proposed for fully supervised segmentation, respectively. Both showed Dense CRF [2] were helpful to boost segmentation performance even in the weakly supervised setting.

While all the above-mentioned methods on weakly supervised segmentation employed only feed-forward computation, we adopted a method based on back-propagation (BP) computation. In this paper, our BP-based method outperforms all the methods based on feed-forward computation.

2.3 Gradient-based Region Estimation with Back-Propagation

Simonyan et al. [1] showed that object segmentation without pixel-wise training data can be done by using back-propagation processing which is a method to train a CNN. To train a CNN, in general, we optimize CNN parameters so as to minimize the loss between groundtruth values and output values. In the back-propagation process, derivatives of the loss function are propagated from the top layers to the lower layers. Springenberg et al. [23] also proposed a method for object localization by back-propagating the derivatives of a maximum loss value of the object detected in the feed-forward computation. They achieved more accurate localization by limiting back-propagating values to positive values. Recently, Pan et al. [24] extended BP-based saliency maps by adding superpixel-based region refinements. BP-based methods was extended to temporal localization of events in a video by Gan et al. [25].

Although these methods can localize single objects in given images, it is difficult to localize multiple different kinds of objects in the same image as shown in Fig.1(B)(C). This is because Pan et al. [24] proposed their method for generic salient object detection. BP-based methods have never been introduced into weakly-supervised semantic segmentation for multiple object images such as those of the PASCAL VOC dataset so far. To apply BP-based localization methods to the PASCAL VOC segmentation task, we need to modify them so that they can estimate class-specific saliency. In this paper, we have achieved that, and we show that our class-specific saliency maps (DCSM) are suitable for unary potentials of Dense CRF as shown in Fig.1(F)(G)(H).

Moreover, all the existing BP-based methods used only the derivatives of the loss function with respect to an input image for object localization, and did not use derivatives or feed-forward activations in the intermediate layers. In our work, we obtained more distinctive class-specific saliency maps by using the derivatives of multiple intermediate layers.

3 Methods

In this section, we overview the proposed method, and explain the detail of the method which consists of three elements: multi-label training of CNN, multi-class object saliency map estimation which was inspired by [1], and fully connected CRF [2].

3.1 Overview

To achieve semantic segmentation for a given image, we (1) perform multi-label classification on a given image by feed-forward computation of the CNN, (2)

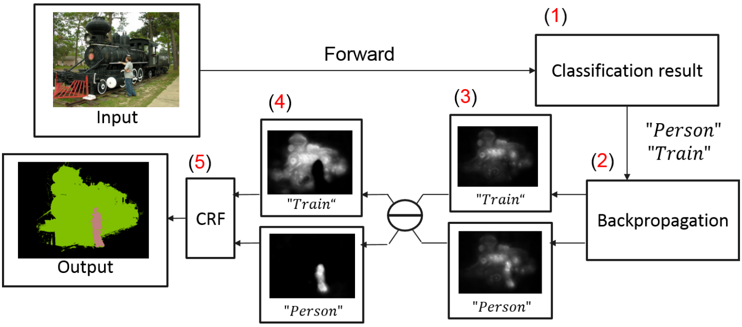


Fig. 2. Processing flow of the proposed method: (1) multi-label classification (2) computation of back-propagation with respect to each of the detected class labels (3) generating raw class saliency maps (4) subtracting raw saliency maps of the other candidate classes from the saliency maps of the target class (5) applying Dense CRF with subtracted class saliency maps as unary potential

calculate CNN derivatives with respect to feature maps of the intermediate convolutional layers with back-propagation by using each of the detected class labels as supervised signals in the loss function, (3) aggregate CNN derivatives of several intermediate layers with up-sampling to generate raw class saliency maps, (4) subtract raw saliency maps of the other candidate classes from the saliency maps of the target class, and (5) apply fully-connected CRF (Dense CRF) with subtracted class saliency maps as unary potential. Finally we obtain a segmentation result. The processing flow is shown in Fig. 2.

3.2 Training CNN

For preparation, we train a CNN with a multi-label loss function. As an off-the-shelf basic CNN architecture, we use the VGG-16 [26] pre-trained with 1000-class ILSVRC datasets. In our framework, we fine-tune a CNN with training images with only image-level multi-label annotation.

Recently, fully convolutional networks (FCN) which accept arbitrary-sized inputs are used commonly in works on CNN-based detection and segmentation such as [12] and [10], in which fully connected layers with n units were replaced with the equivalent convolutional layers having n 1×1 filters. Following them, we introduce FCN to enable multi-scale generation of class saliency maps. When training, we insert global max pooling before the final loss function layer to deal with larger input images than the images used for pre-training of the VGG-16. We use images which are normalized to 500×500 by rescaling to have the largest size of the 500 pixels and zero-padding for training and testing in the same way as [12]. For multi-scale training, we resize training images randomly between the ratio 0.7 and 1.4 within a mini-batch.

To carry out multi-label training of the CNN, we use a Sigmoid cross entropy loss which is a standard loss function for multi-label annotation instead of a soft-max loss used in the original VGG-16 in the same way as [12] and [20]. The

Sigmoid cross entropy loss function is represented in the following equation:

$$\text{loss} = \sum_{n=1}^{K}[-p_n \log \hat{p_n} - (1 - p_n)\log(1 - \hat{p_n})] \tag{1}$$

where K is the number of classes, $p_n = \{0, 1\}$ which represents the existence of the corresponding class label, and $\hat{p_n}$ means the output of Sigmoid function of the class score $f_n(x)$ represented in the following equation:

$$\hat{p_n} = \frac{1}{1 + e^{-f_n(x)}} \tag{2}$$

3.3 Class Saliency Maps

We propose a new method to estimate class-specific saliency maps by enhancing the method proposed by Simonyan et al. [1] greatly. It consists of (1) extracting CNN derivatives with respect to feature maps of the intermediate convolutional layers, (2) subtracting class saliency maps between the target class and the other classes, and (3) aggregation of multi-scale saliency maps.

Extracting CNN derivatives In [1], they regarded the derivatives of the class score with respect to the input image as class saliency maps. However, the position of an input image is the furthermost from the class score output on the deep CNN, which sometime causes weakening or vanishing of gradients. Instead of the derivatives of the class score with respect to the input image, we use the derivatives with respect to feature maps of the relatively upper intermediate layers which are expected to retain more high-level semantic information. We select the maximum absolute values of the derivatives with respect to the feature maps at each location of feature maps across all the kernels, and up-sample them with bilinear interpolation so that their size becomes the same as an input image (Fig.3 (C)-(G)). Finally we average them to obtain one saliency map (Fig.3 (B)). The idea on aggregating of information extracted from multiple feature layers was inspired by the work of [10], although they extracted not CNN derivatives but feature maps calculated by feed-forwarding.

The class score derivative v_i^c of a feature map in the i-th layer is the derivative of class score Sc with respect to the layer L_i at the point (activation signal) L_i^0:

$$v_i^c = \left.\frac{\partial S_c}{\partial L_i}\right|_{L_i^0} \tag{3}$$

v_i^c can be computed by back-propagation. After obtained v_i^c, we up-sample it to w_i^c with bilinear interpolation so that the size of an 2-D map of v_i^c becomes the same as an input image. Next, the class saliency map $M_i^c \in \mathcal{R}^{m \times n}$ is computed as $M_{i,x,y}^c = \max_{k_i} |w_{i,h_i(x,y,k)}^c|$, where $h_i(x, y, k)$ is the index of the element of w_i^c. Note that each value of the saliency map is normalized by $\tanh(\alpha M_{i,x,y} / \max_{x,y} M_{i,x,y})$ for visualization in Fig.3 and all the other figures with $\alpha = 3$.

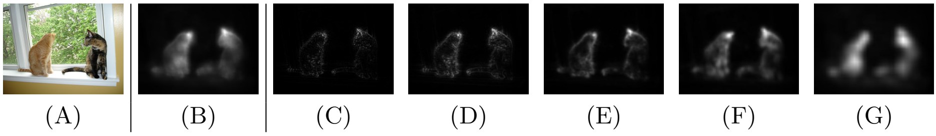


Fig. 3. Class saliency maps obtained from the VGG16-net fine-tuned with the PASCAL VOC 2012 dataset. (A) an input image, (B) average of [(E)(F)(G)], (C) conv1_1, (D) conv2_1, (E) conv3_2, (F) conv4_2, (G) conv5_2

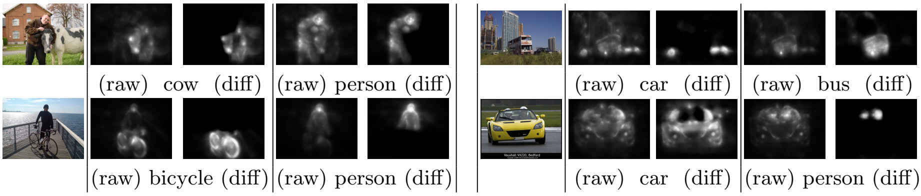


Fig. 4. (raw) raw maps without subtraction (diff) maps with subtraction of other class maps.

Subtracting raw class saliency maps As shown in Fig.1(B)(C), the saliency maps of two or more different classes tend to be similar to each other especially in the image-level. The saliency maps by [1] are likely to correspond to foreground regions rather than specific class regions. This problem is relaxed in the proposed methods, because we use saliency maps obtained from intermediate layers. However, the saliency regions of different classes are still overlapped with each other (Fig.4 (raw)).

To resolve that, we subtract saliency maps of the other candidate classes from the saliency maps of the target class to differentiate target objects from other objects. Here, we assume that we use the CNN trained with multi-label loss, and select several candidate classes the class score of which exceed a pre-defined threshold with a pre-defined minimum number. (In the experiments, we set 0.5 to the threshold and 3 to the minimum number.)

The improved class saliency maps with respect to class c, $\tilde{M}_i^c$, are represented as:

$$\tilde{M}^c_{i,x,y} = \sum_{c' \in candidates} \max\left(M^c_{i,x,y} - M^{c'}_{i,x,y}, 0\right) [c \neq c'], \tag{4}$$

where *candidates* is a set of the selected candidate classes. Fig.4 shows results without subtraction in the left (raw) and ones with subtraction in the right (diff). As we can see, subtraction of saliency maps resolved overlapped regions among the maps of the different classes.

Aggregating multi-scale class saliency maps We use fully convolutional networks (FCN) which accept arbitrary-sized inputs for multi-scale generation of class saliency maps. If the larger input image than one for the original CNN is given to the fully-convolutionalized CNN, the output becomes class score maps represented as $h \times w \times C$ where C is the number of classes, and h and w are larger than 1. To obtain CNN derivatives with respect to enlarged feature maps, we simply back-propagate the target class score map which is define as $S_c(:,:,c) = 1$

(in the Matlab notation) with 0 for all the other elements, where c is the target class index.

The final class saliency map $\hat{M}^c$ averaged over the layers and the scales is obtained as follows:

$$\hat{M}^c_{x,y} = \frac{1}{|S||L|} \sum_{j \in S} \sum_{i \in L} \tanh(\alpha \tilde{M}^c_{j,i,x,y}), \tag{5}$$

where L is a set of the layers for which saliency maps are extracted, S is a set of the scale ratios, and α is a constant which we set to 3 in the experiments. Note that we assume the size of $\tilde{M}_{j,i}$ for all the layers are normalized to the same size as an input image before calculation of Eq.5.

In the experiments, we adopted guided back-propagation (GBP) [23] as back-propagation method instead of normal back-propagation (BP) used in [1]. The difference between two methods is in the backward computation through ReLU. GBP can visualize saliency maps with less noise components than normal BP by back-propagating only positive values of CNN derivatives through ReLU [23].

3.4 Fully Connected CRF

Conditional Random Field (CRF) is a probabilistic graphical model which considers both node priors and consistency between nodes. Because class-specific saliency maps obtained in the previous subsection represent only probability of the target classes on each pixel and have no explicit information on object region boundaries, we apply CRF to estimate object boundaries. In this paper, we use fully connected CRF (noted as "FC-CRF" or "Dense CRF") [2] where every pixel is regarded as a node, and every node is connected to every other node. The energy function is defined as follows:

$$\mathrm{E}(\mathbf{c}) = \sum_i \theta_i(c_i) + \sum_{i,j} \theta_{i,j}(c_i, c_j) \tag{6}$$

where c_i represents a class assignment on pixel i. The first unary term of the above equation is calculated from class saliency maps $\hat{M}^c_i$. We defined it as $\theta_i(c_i) = -\log(\hat{M}^c_{x,y})$.

Since the CNN we trained has no background class, we have no class maps on background class. To use CRF for image segmentation, a unary potential on the background class is needed as well as foreground potential. We estimate a unary potential on the background class from the maps of the candidate classes selected in the previous step by the following equation.

$$\hat{M}^{BG}_{x,y} = 1 - \max_{c \in target} \hat{M}^c_{x,y} \tag{7}$$

where $\hat{M}^{BG}_{x,y}$ is a saliency map of background class, and *target* represents a set of the selected candidate classes.

The pairwise term of Eq.6 is represented by $\theta_{i,j}(c_i, c_j) = u(c_i, c_j)k(f_i, f_j)$ where $u(x_i, x_j) = \begin{cases} 1 & (c_i \neq c_j) \\ 0 & others \end{cases}$ and $k(\mathbf{f}_i, \mathbf{f}_j)$ is a Gaussian kernel. Note that $\mathbf{f}_i, \mathbf{f}_j$ represents some kinds of image features extracted from pixel i and j. Following [2], we adopt bilateral position and color terms, and the kernels are

$$k(\mathbf{f}_i, \mathbf{f}_j) = w_1 \exp\left(-\frac{|p_i - p_j|^2}{2\gamma_{\alpha^2}} - \frac{|I_i - I_j|^2}{2\gamma_{\beta^2}}\right) + w_2 \exp\left(-\frac{|p_i - p_j|^2}{2\gamma_{\gamma^2}}\right) \tag{8}$$

where the first kernel depends on both pixel positions (denoted as p) and pixel color intensities (denoted as I), and the second kernel only depends on pixel positions. The hyper parameters γ_α, γ_β, and γ_γ control the scale of the Gaussian kernels. This model is amenable to efficient approximate probabilistic inference proposed by [2].

4 Experiments

We evaluated the proposed methods using the PASCAL VOC 2012 data. We compared the results with state-of-the-arts, and show significant improvements by the proposed methods.

4.1 Dataset

In the experiments, we used the PASCAL VOC 2012 segmentation data [27] to evaluate the proposed method. The PASCAL VOC 2012 segmentation dataset consists of 1464 training images, 1449 validation images, and 1456 test images including 20 class pixel-level labels as well as image-level labels. For training, we used the augmented PASCAL VOC training data including 10582 *train_aug* images provided by Hariharan et al. [28] in the same way as the other works on weakly-supervised segmentation such as MIL-FCN [20], EM-Adapt [22] and CCNN [21].

For evaluation, we used a standard intersection over union (IoU) metric which was the official evaluation metric in the PASCAL VOC segmentation task.

4.2 Experimental Setup

We used VGG-16 [26] as a basic CNN, modified it regarding Sigmoid entropy loss for multi-label training, random resizing of training images and global max pooling for multi-scale training following [12], and fine-tuned it with PASCAL VOC *train_aug* dataset. We used Caffe [29] to train the CNN with batchsize 2, learning rate 1e-5, momentum 0.9 and weight decay 0.0005. Note that we followed [21] regarding very small batchsize for fine-tuning of VGG-16. For the first 30000 iterations, we fine-tuned only the upper layers of the modified VGG-16 than Pool_5, and for the next 20000 iterations, we fine-tuned all the layers.

Table 1. Results of the mean IoU by Simonyan et al. and ours on Pascal VOC 2012 *val set*

method \ α	2	2.5	3	4	5	6	7	8	9	10	15
Simonyan et al.	-	-	10.0	20.6	28.3	32.7	33.4	33.8	33.8	33.3	28.7
DCSM (ours)	40.0	44.0	44.1	40.6	36.4	-	-	-	-	-	

As hyperparameters of the fully-connected CRF, we used the following parameters which were chosen by grid search with validation data: $w_1 = w_2 = 1$, $\gamma_\alpha = 30$, $\gamma_\beta = 10$, and $\gamma_\gamma = 3$.

Using GPU, it takes about 0.3 seconds to perform segmentation for one image.

4.3 Evaluation on Class Saliency Maps

First, we compare the class saliency maps estimated by the proposed method (noted as DCSM (Distinct Class-specific Saliency Maps)) with ones by Simonyan et al. [1] qualitatively. Fig.5 shows both the results by Simonyan et al. and our method for three multiple object images and one single object images. From these results, it is shown that our method is much more effective for not only multiple object images but also single object images than the previous method. This figure shows our results are significantly better than [1], because we aggregate gradients in the multiple intermediate layers and carry out subtraction of raw class saliency maps. Our results clearly discriminated multiple regions of the different classes.

Fig.6 shows the results for images containing three or more objects. In even such cases, all the class saliency maps except for "chair" in the top-right sample were estimated successfully.

To compare both quantitatively, we carried out weakly supervised segmentation using estimated class saliency maps and Dense CRF. To obtain maps for unary potentials of CRF, we used Eq.5 which contains a hyper-parameter, α. As shown in Table 1, we searched for the best values of α for both of Simonyan et al. and our method. As results, ours achieved 44.1% as the best meanIoU with $\alpha = 3$, while Simonyan et al. achieved 33.8% with $\alpha = 8$(or 9). From both the best results, our method is superior to Simonyan et al. significantly as a method to estimate CRF unary potentials under the weakly supervised setting.

4.4 Effects of Parameter Choices

Intermediate layers In the proposed method, we extract CNN derivatives from intermediate layers of the VGG-16, and averaged them to estimate class saliency maps. We examined the effects on which layers we use to extract derivatives from. Table 2 shows the results evaluated with VOC *val set* varying the layer combinations. "Block1" in the table means the average of conv1_1 and conv1_2 in VGG-16, and "average Block 3,4,5" means the average of Block 3, Block 4, and Block 5. Among the single blocks, Block 4 achieved the best result, and among the block combinations, the combination of Block 3,4,5 achieved

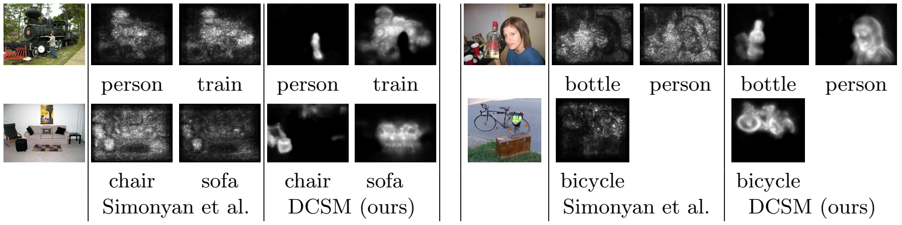


Fig. 5. Obtained class saliency maps (Left) by Simonyan et al. [1] (Right) by the proposed method (DCSM).

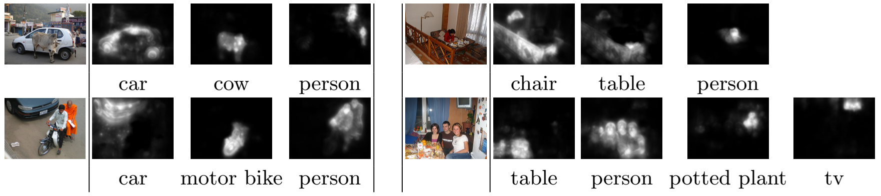


Fig. 6. Obtained class saliency maps for images containing three or more classes.

Table 2. Effects by layers from which CNN derivatives are extracted.

layer	mean IoU
block1 (conv1_1, conv1_2)	5.5
block2 (conv2_1, conv2_2)	21.5
block3 (conv3_1, conv3_2, conv3_3)	32.5
block4 (conv4_1, conv4_2, conv4_3)	**40.3**
block5 (conv5_1, conv5_2, conv5_3)	26.3
average block 1,2,3,4,5	41.3
average block 2,3,4,5	42.2
average block 3,4,5	**42.8**
average block 4,5	42.5
average block 3,4	37.97

Table 3. Effects by input image size and multi-scale aggregation.

input image size	mean IoU
(1) 300 × 300	34.5
(2) 400 × 400	41.0
(3) **500 × 500**	**42.4**
(4) 600 × 600	41.8
(5) 700 × 700	40.0
(6) 800 × 800	34.5
average (1),(2),(3)	41.1
average (2),(3)	42.9
average (2),(3),(4)	**43.5**
average (3),(4)	42.9
average (3),(4),(5)	42.5
average (3),(4),(5),(6)	42.8

Table 4. Effects on the number of raw class maps for subtraction.

class N	0	1	2	3	4	5	10	15
mean IoU	38.2	42.2	43.5	44.1	44.2	44.0	43.7	43.3

Table 5. Effects on the way of back-propagation.

method	BP	GBP
mean IoU	41.2	44.1

the best. Although Block 5 itself was less effective, adding Block 5 to combinations was effective to boost performance. This shows that aggregation of CNN derivatives extracted from multiple upper layers is the better choice.

Size of input images Because we use fully convolutional CNN which can deal with arbitrary-sized input images, we examined the effects on input image size and multi-scale combination of input images. Note that we used bilinear up-scaling when the size of the original images were less than the indicated size. Table 3 shows the results, which indicates 500 × 500 was the best, and the combination of 400 × 400, 500 × 500 and 600 × 600 was the best. This is partly because we used training images with random resizing from 350 to 700 pixels. From these results, multi-scale aggregation helped boost performance.

Minimum number of the raw class maps for subtraction We use the raw class saliency maps of the top-N classes for raw class map subtraction which

Table 6. Results on PASCAL VOC 2012 *val set*.

Methods	bg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	**mIoU**
Weakly Supervised:																						
MIL-FCN [20]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	25.7
EM-Adapt [22]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	38.2
CCNN [21]	65.9	23.8	17.6	22.8	19.4	36.2	47.3	46.9	47.0	**16.3**	36.1	22.2	43.2	33.7	44.9	39.8	29.9	33.4	22.2	**38.8**	36.3	34.5
MIL-sppxl [18]	77.2	37.3	18.4	25.4	28.2	31.9	41.6	48.1	50.7	12.7	45.7	14.6	50.9	44.1	39.2	37.9	28.3	44.0	19.6	37.6	35.0	36.6
MIL-bb [18]	78.6	46.9	18.6	27.9	30.7	38.4	44.0	49.6	49.8	11.6	44.7	14.6	50.4	44.7	40.8	38.5	26.0	45.0	20.5	36.9	34.8	37.8
MIL-seg [18]	**79.6**	**50.2**	21.6	40.6	**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
DCSM w/o CRF (ours)	72.5	35.0	23.7	33.6	20.9	32.8	57.6	49.2	51.4	14.7	42.9	32.8	51.9	43.1	53.2	47.6	35.2	47.7	28.6	40.3	36.7	40.5
DCSM w/ CRF (ours)	76.7	45.1	**24.6**	**40.8**	23.0	34.8	**61.0**	**51.9**	52.4	15.5	45.9	**32.7**	54.9	48.6	**57.4**	**51.8**	**38.2**	**55.4**	**32.2**	**42.6**	**39.6**	**44.1**

Table 7. Results on PASCAL VOC 2012 *test set*.

Methods	bg	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	motor	person	plant	sheep	sofa	train	tv	**mIoU**
Fully Supervised:																						
O2P [30]	85.4	69.7	22.3	45.2	44.4	49.6	66.7	57.8	56.2	13.5	46.1	32.3	41.2	59.1	55.3	51.0	36.2	50.4	27.8	46.9	44.6	47.6
SDS [8]	86.3	63.3	25.7	63.0	39.8	59.2	70.9	61.4	54.9	16.8	45.0	48.2	50.5	51.0	57.7	63.3	31.8	58.7	31.2	55.7	48.5	51.6
FCN-8s [10]	-	76.8	34.2	68.9	49.4	60.3	75.3	74.7	77.6	21.4	62.5	46.8	71.8	63.9	76.5	73.9	45.2	72.4	37.4	70.9	55.1	62.2
Zoom out [11]	89.8	81.9	35.1	78.2	57.4	56.5	80.5	74.0	79.8	22.4	69.6	53.7	74.0	76.0	76.6	68.8	44.3	70.2	40.2	68.9	55.3	64.4
DeepLab [3]	**93.1**	**84.4**	54.5	**81.5**	63.6	65.9	**85.1**	79.1	**83.4**	30.7	74.1	59.8	79.0	76.1	**83.2**	**80.8**	**59.7**	82.2	**50.4**	73.1	63.7	71.6
CRF as RNN [4]	87.5	39.0	**79.7**	64.2	**68.3**	**87.6**	80.8	**84.4**	30.4	**72.0**	**78.2**	**60.4**	**80.5**	**77.8**	83.1	80.6	59.5	**82.8**	47.8	**78.3**	**67.1**	**72.0**
Size tag Supervised:																						
CCNN w/size [21]	-	42.3	24.5	56.0	30.6	39.0	58.8	52.7	54.8	14.6	48.4	34.2	52.7	46.9	61.1	44.8	37.4	48.8	30.6	47.7	41.7	45.1
Weakly Supervised:																						
MIL-FCN [20]	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	24.9
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 [21]	-	21.3	17.7	22.8	17.9	38.3	51.3	43.9	51.4	15.6	38.4	17.4	46.5	38.6	53.3	40.6	34.3	36.8	20.1	32.9	38.0	35.5
MIL-ILP-seg [18]	**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
DCSM w/o CRF (ours)	73.9	34.1	24.4	39.6	18.0	37.6	57.8	49.0	51.5	13.3	42.3	33.5	47.8	44.2	63.7	44.3	34.5	48.3	31.2	35.7	37.1	41.0
DCSM w/ CRF (ours)	78.1	43.8	**26.3**	**49.8**	19.5	**40.3**	**61.6**	53.9	52.7	13.7	47.3	**34.8**	50.3	48.9	**69.0**	**49.7**	**38.4**	**57.1**	**34.0**	**38.0**	40.0	**45.1**

are estimated by feed-forward multi-class classification[1] of the input image in addition to the maps of the classes the output scores of which are more than the pre-defined threshold, 0.5. We examined effects on the top-N varying N. Note that exceptionally $N = 0$ means that subtraction was never carried out, that is, the results without subtraction. As shown in Table 4, using the top-4 ($N = 4$) raw class maps was the best[2]. Compared with $N = 0$, subtraction is always helpful to raise segmentation performance.

Guided BP vs. BP We compared normal back propagation (BP) used in Simonyan et al. [1] with guided back propagation (GBP) proposed by Springenberg et al. [23]. GBP was better than BP as shown in Table 5.

4.5 Comparison with Other Methods

In the final subsection, we compare our results (DCSM) with other results by CNN-based methods quantitatively. Table 6 and Table 7 show the results for PASCAL VOC 2012 *val set* and *test set*, respectively.

While MIL-FCN [20], EM-Adapt [22], CCNN [21] and our methods used PASCAL VOC training data and augmented training data provided by [28],

[1] The classification accuracy of multi labels by the fine-tuned VGG-16 was 85.2%.

[2] We used $N = 3$ in all the other experiments to save computation.

MIL-{sppxl,bb,seg} by Pedro et al.[18] used their original additional training images which contains 700,000 images. Our method is different from other methods in terms of the way to use a CNN. While the existing methods employed only feed forward computation [20,18,22,21], we use backward computation as well as feed forward computation. Although the way to train CNN is the same as MIL-FCN [20] and MIL-{sppxl,bb,seg} [18], the method to localize objects is essentially different.

As shown in the tables, our results by DCSM with CRF outperformed all the *val* and *test* results by the weakly-supervised methods including MIL-{sppxl,bb,seg} which used about 70 times as many training images with the margin, 2.1 points and 4.5 points, respectively. Note that "CCNN w/size" used additional information on size of training images, the mean IoU of which was equivalent to ours.

In Table 7, we also compared our results with the fully supervised methods. Our result is close to the result by one of the best non-CNN-based fully supervised method, O2P [30]. Their difference is only 2.5 points.

Finally, we show qualitative results by the proposed method without/with Dense CRF in Fig.7.

5 Conclusions

In this paper, we proposed a new weakly-supervised semantic segmentation method consisting of a novel method of class saliency map estimation and Dense CRF. The proposed distinct class saliency maps (DCSM) outperformed the maps by Simonyan et al. [1] both qualitatively and quantitatively. The experimental results proved the effectiveness of the proposed method, which achieved the state-of-the-arts on the PASCAL VOC 2012 weakly supervised segmentation [3].

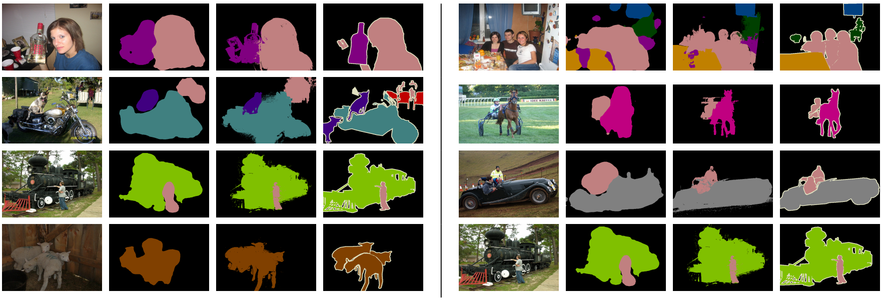


Fig. 7. Qualitative results on VOC 2012. Each row shows (left) input image, (middle left) results estimated from class maps, (middle right) results after applying FC-CRF, and (right) groundtruth. Please see the supplementary material for more results.

[3] At the time of publishing of this paper, the two works published as arXiv papers [31,32] achieved the state-of-the-art results which were more than 50% on the PASCAL VOC 2012.

References

1. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: Visualising image classification models and saliency maps. In: Proc. of International Conference on Learning Representations. (2014)
2. Krahenbuhl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. In: Advances in Neural Information Processing Systems. (2011)
3. Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., L., Y.A.: Semantic image segmentation with deep convolutional nets and fully connected CRFs. In: Proc. of International Conference on Learning Representations. (2015)
4. Zheng, S., Jayasumana, S., Paredes, B.R., Vineet, V., Su, Z.: Conditional random fields as recurrent neural networks. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2015)
5. Myers, A., Johnston, N., Rathod, V., Korattikara, A., Gorban, A., Silberman, N., Guadarrama, S., Papandreou, G., Huang, J., Murphy, K.: Im2calories: Towards an automated mobile vision food diary. In: Proc. of IEEE International Conference on Computer Vision. (2015)
6. Shimoda, W., Yanai, K.: CNN-based food image segmentation. In: Proc. of International Workshop on Multimedia Assisted Dietary Management (MADIMA). (2015)
7. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2014)
8. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Simultaneous detection and segmentation. In: Proc. of European Conference on Computer Vision. (2014)
9. Uijlings, J.R.R., van de Sande, K.E.A., Gevers, T., Smeulders, A.W.M.: Selective search for object recognition. Volume 104. (2013) 154–171
10. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2015)
11. Mostajabi, M., Yadollahpour, P., Shakhnarovich, G.: Feedforward semantic segmentation with zoom-out features. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2015)
12. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Is object localization for free? - weakly-supervised learning with convolutional neural networks. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2015)
13. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: Integrated recognition, localization and detection using convolutional networks. In: Proc. of International Conference on Learning Representations. (2014)
14. Bell, S., Upchurch, P., Snavely, N., Bala, K.: Material recognition in the wild with the materials in context database. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2015)
15. Vezhnevets, A., Buhmann, J.M.: Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2010)
16. Vezhnevets, A., Buhmann, J.M.: Weakly supervised structured output learning for semantic segmentation. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2012)
17. Zhang, L., Gao, Y., Xia, Y., Lu, K., Shen, J., Ji, R.: Representative discovery of structure cues for weakly-supervised image segmentation. IEEE Transactions on Multimedia **16**(2) (2014) 470–479

18. Pedro, P., Ronan, C.: From image-level to pixel-level labeling with convolutional networks. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2015)
19. Pablo, A., Jordi, P., J.T., Ferran, M., Jitendra, M.: Multiscale combinatorial grouping. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2014)
20. Pathak, D., Shelhamer, E., Long, J., Darrell, T.: Fully convolutional multi-class multiple instance learning. In: Proc. of International Conference on Learning Representations. (2015)
21. Pathak, D., Krahenbuhl, P., Darrell, T.: Constrained convolutional neural networks for weakly supervised segmentation. In: Proc. of IEEE International Conference on Computer Vision. (2015)
22. Papandreou, G., Chen, L.C., Murphy, K., Yuille, A.L.: Weakly-and semi-supervised learning of a dcnn for semantic image segmentation. In: Proc. of IEEE International Conference on Computer Vision. (2015)
23. Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M.: Striving for simplicity: The all convolutional net. In: Proc. of International Conference on Learning Representations. (2015)
24. Pan, H., Hui, J.: A deep learning based fast image saliency detection algorithm. arXiv preprint arXiv:1602.00577 (2016)
25. Gan, C., Wang, N., Yang, Y., Yeung, D., Hauptmann, A.G.: Devnet: A deep event network for multimedia event detection and evidence recounting. In: Proc. of IEEE Computer Vision and Pattern Recognition. (2015)
26. Simonyan, K., Vedaldi, A., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: Proc. of International Conference on Learning Representations. (2015)
27. Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. International Journal of Computer Vision **111**(1) (2015) 98–136
28. Hariharan, B., Arbelaez, P., Bourdev, L., Maji, S., J., M.: Semantic contours from inverse detectors. In: Proc. of IEEE International Conference on Computer Vision. (2011)
29. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093 (2014)
30. Carreira, J., Caseiro, R., Batista, J., Sminchisescu, C.: Semantic segmentation with second-order pooling. In: Proc. of European Conference on Computer Vision. (2012)
31. Wei, Y., Liang, X., Chen, Y., Shen, X., Cheng, M., Zhao, Y., Yan, S.: STC: A simple to complex framework for weakly-supervised semantic segmentation. In: arXiv:1509.03150. (2015)
32. Kolesnikov, A., Lampert, C.: Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In: arXiv:1603.06098. (2016)