Weakly Supervised Semantic Segmentation by a Class-Level Multiple Group Cosegmentation and Foreground Fusion Strategy

Fanman Meng, *Member, IEEE*, Kunming Luo, Hongliang Li, *Senior Member, IEEE*, Qingbo Wu, *Member, IEEE*, and Xiaolong Xu

***Abstract*— Weakly supervised semantic segmentation uses image-level labels to extract object regions. The existing methods focus on efficiently training CNN-based segmentation networks using the image-level labels. In contrast to the existing methods, this paper proposes a new fusion-based method, which first segments the foregrounds of each image by multiple group cosegmentation and then generates the semantic segmentation by combining the foregrounds. Specifically, a new CNN-based multiple group cosegmentation network is first proposed to segment foregrounds employing two cues, the discriminative cue and the local-to-global cue. Then, the fusion method is proposed to simply perform semantic segmentation based on the multiple group cosegmentation results. Experiments on the PASCAL VOC 2012 and MS COCO 2017 datasets demonstrate the effectiveness of the proposed method with mIoU values that are obviously larger than those of the existing methods.**

***Index Terms*— Multiple group cosegmentation, weakly supervised semantic segmentation, class activation map, region fusion.**

I. Introduction

SEMANTIC segmentation [1] is a fundamental task in many computer vision applications. It is also a challenging task due to the variations of foregrounds and the interferences of backgrounds. In the past decade, several semantic segmentation methods have been proposed. Based on the annotations given for segmentation, most of the existing segmentation methods can be classified into three categories: supervised [1]–[3], semi-supervised [4], [5] and weakly supervised semantic segmentation methods [6]–[8].

The supervised semantic segmentation method learns a segmentation model from pixel-level annotations. The segmentation results can be obviously improved by using the deep learning-based methods such as the convolutional neural network (CNN) [1]–[3] and long short-term memory (LSTM) [9] network. However, these methods rely on large numbers of manual annotations, which are often unavailable in practice. The semi-supervised semantic segmentation method uses the weak image-level labels, as well as a small number of pixel-level annotations, which can efficiently guarantee the segmentation performance while dramatically decreasing the burden of label annotation. In some practical applications, however, the pixel-level annotations are unavailable.

To overcome the drawbacks of the supervised and semi-supervised manners, researchers have turned to the weakly supervised manner that segments regions by image-level labels. Compared with pixel-level annotations, the image-level labels can be obtained easily. However, because the image-level labels are extremely rough, it is challenging to generate the object priors. To capture the object priors from the image-level labels, two types of training-based methods have been recently proposed: one-stage-based and two-stage-based methods.

One-stage-based methods [10]–[13] aim at training an end-to-end CNN-based segmentation network by image-level labels, where the discriminative regions captured by the classification network are used to form the object priors [14]. Designing efficient segmentation loss is essential to the one-stage-based method, and several types of segmentation losses have been studied, such as the classification loss by image-level labels [11], [12] and the segmentation loss by pseudo-annotations [15]. The attention mechanism [16] is also used to discriminate foregrounds from backgrounds. These studies show that the discriminative regions in terms of the class activation maps can be well captured by the classification network. However, the discriminative regions are usually a part of the region rather than the object region, e.g., the "head" region of a "Cat". Although some strategies such as erasing strategy have been proposed to partially remedy the problem of incomplete extraction regions, it is still difficult to generate object-level activation regions.

Two-stage-based methods [17], [18] consist of two steps. The first step generates pseudo-annotations from multiple images. The second step trains the semantic segmentation model by the pseudo-annotations. The first step is the essential step. Some strategies such as simple-to-complex [19], user interactions [20] and videos [21] are proposed to enhance the generation of the pseudo-annotations. However, it has been proven that generating pseudo-annotations accurately is a challenging task.

Manuscript received December 4, 2018; revised April 14, 2019, June 5, 2019, September 7, 2019, and November 6, 2019; accepted December 8, 2019. Date of publication December 24, 2019; date of current version December 4, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61871087, Grant 61831005, Grant 61525102, Grant 61971095, and Grant 61601102 and in part by the Sichuan Science and Technology Program under Grant 2018JY0141. This article was recommended by Associate Editor H. Lu. *(Corresponding author: Fanman Meng.)*

The authors are with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: fmmeng@uestc.edu.cn).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2019.2962073

Different from the existing strategies above, this paper proposes a new weakly supervised semantic segmentation strategy employing a class-level multiple group cosegmentation and fusion method (a brief version has been accepted by VCIP 2018). Rather than training CNN-based segmentation networks using the image-level labels as used in the existing methods, we use the idea of forming the semantic segmentation result by combining the foreground regions. To this end, we use a discriminative cue to obtain the initial local object priors and then employ a local-to-global cue to segment object regions from the local object priors. Subsequently, we simply combine the cosegmentation regions to form the semantic segmentation results. Although our method can be considered as a two-stage based method, it is totally different from the existing two-stage based method since the difficult step of pseudo-annotation generation is avoided in our method.

Specifically, a class-level multiple group cosegmentation model is proposed. We first train a CNN-based image classification network by the image-level labels to coarsely locate the object regions. Since the discriminative regions are usually local regions of the objects, we next train a local-to-global network that introduces the global priors and segments the objects from the local priors. After obtaining the foregrounds of each class, we form the semantic segmentation results by fusing the cosegmentation regions based on the image-level labels. Since the pixels may be assigned to multiple classes of labels, the probabilities of the pixels belonging to different classes are used to determine the class labels simply. We verify the proposed method on the PASCAL VOC 2012 and MS COCO 2017 datasets. The experimental results show that the proposed method can obtain better results.

The contributions of the proposed method are listed as follows.

- We propose a new strategy for weakly supervised semantic segmentation, which first obtains the foregrounds of each class by cosegmentation, and then forms the semantic segmentation by combining the cosegmentation results.
- A new multiple group cosegmentation model and the corresponding convolutional neural network are proposed. The proposed method is an extension of the cosegmentation task by considering the multiple sets of images belonging to different classes.
- A new method for CAM generation is proposed, and a CNN-based local-to-global segmentation network is advanced.

The rest of this paper is organized as follows. The related work is introduced in Section II. The proposed method is presented in Section III. We verify the proposed method in Section IV. Finally, the conclusion is drawn in Section V.

II. Related Work

A. Weakly Supervised Semantic Segmentation

Weakly supervised semantic segmentation is a challenging task due to the gap between image-level labels and pixel-level labels. To generate more object priors, multiple images are simultaneously considered, and two types of weakly supervised semantic segmentation methods, namely, one-stage-based and two-stage-based methods, are proposed.

One-stage-based weakly supervised semantic segmentation accomplishes segmentation by an end-to-end network [10]. The image-level labels are used to formulate the network loss [11], [12]. For example, Hong *et al.* [15] propose a weakly supervised semantic segmentation by weak-related annotations. Both the image-level classification loss and the pixel-level segmentation loss (to describe the weak-related annotations) are used to form the network loss. Kolesnikov and Lampert [14] propose a weakly supervised convolutional neural network (SEC) by considering three losses, i.e., seed loss, expansion loss and boundary fitness loss. The three losses are formulated by the class activation regions, the global weighted rank pooling and the conditional random field (CRF) and are minimized simultaneously. Several extensions to the method [14] are proposed. Kim *et al.* [13] use an iterative suppression process to obtain more global discriminative regions. Also based on [14], Kwak *et al.* [22] propose a new layer named the superpixel pooling layer (SPL) to enable the discriminative region to achieve a better fitness that extends to the image edges. Li *et al.* [16] have recently generated an attention map by employing a discriminative map and a self-guided generation of parts to refine the foreground probability map. To generate robust foreground priors, some methods focus on capturing more cues. Roy and Todorovic [23] solve a weakly supervised semantic segmentation problem by combining bottom-up, top-down, and smoothness cues. In the considered framework, the bottom-up classification loss is implemented for image-level labels, the top-down segmentation loss is achieved by the attention model, and the CRF-RNN model addresses the smoothness. Durand *et al.* [24] generate a probability map from a classification network via a multimap transfer layer that can capture the regions with more discriminative parts. Based on the image-level labels, Pathak *et al.* [25] propose a new loss for training the segmentation network. Three linear constraints, such as the suppression constraints for the foreground, background and the object size, are used to form the loss. Vezhnevets *et al.* [26] propose a multiple image segmentation model based on the similarity of superpixels, which clusters similar superpixels among images to form the segmentation results.

The two-stage-based weakly supervised semantic segmentation consists of two steps [17], [18]. The first step uses image-level labels to generate pseudo-annotations. The second step then trains the segmentation network utilizing the pseudo-annotations. Moreover, such a strategy depends on sufficient pseudo-annotations, and their generation is a challenging task. However, several methods have been proposed to generate pseudo-annotations.

1) Simple Images: Some researchers generate pseudo-annotations from simple images. For example, Hou *et al.* [19] generate pseudo-annotations from simple images first and then use the annotations to segment complicated images. Wei *et al.* [10] generate initial pseudo-annotations from simple images by implementing saliency detection. The segmentation model is first learned from the initial pseudo-annotations

obtained by saliency detection. Then, the segmentation model is used to generate pseudo-annotations from the complicated images. The segmentation model is repeatedly updated until reaching convergence.

2) User Interactions: Some methods use a few user interactions to enhance the generation of pseudo-annotations. For example, Russakovsky *et al.* [27] use user-given pixel seeds to constrain a weakly supervised semantic segmentation model. Two types of losses, namely, point-level loss and image-level loss, are proposed. Arbelaez *et al.* [28] use bounding boxes to generate pseudo-annotations from images. The superpixels are combined based on the windows to form the segmentation annotations. Based on DeepLab, Papandreou *et al.* [29] train the segmentation model by bounding boxes and image-level labels. Using the two steps of pseudo-annotation generation and network training, object regions are first segmented from weak labels, and then, the segmentation network is trained by the pseudo-annotations. The expectation maximum (EM) algorithm is employed to refine the segmentation model and the segmentation results alternatively. Lin *et al.* [30] implement user scribbles to generate initial annotations. The FCN [2] is then trained to update the segmentation annotations. Rather than windows, Khoreva *et al.* [31] use GrabCut to generate better annotations. Qi *et al.* [32] first generate object windows automatically and then generate the annotations by merging the superpixels through use of the windows. Tang *et al.* [20] introduce the classical normalized cut loss in the CNN method to generate the pseudo-annotations from user scribbles.

3) Video: Some methods generate more accurate pseudo-annotations from videos. Tokmakov *et al.* [33] generate pseudo-annotations from videos by motion cues. Hong *et al.* [21] use web-crawled videos to generate pseudo-masks, where the attention regions are employed to enhance the generation of the segmentation mask.

4) Image-Level Labels: Image-level labels are also used to generate pseudo-annotations directly. Saleh *et al.* [34] combine class activation maps and multiple levels of the convolution layer to obtain the foreground probability map, which is then fed to the CRF to output the segmentation regions. Ge *et al.* [35] generate pseudo-annotations employing the three steps of object localization, pseudo-annotation generation, and network training. Oh *et al.* [36] generate pseudo-annotations with discriminative cues and saliency cues, and better results are obtained. Jiwoon and Suha [37] generate the segmentation masks by the random walk process, where an affinity matrix that describes the similarity of nearby pixels is calculated. We refer readers to [38] for more detailed reviews of weakly supervised semantic segmentation.

B. Cosegmentation

Cosegmentation [6], [39]–[42] aims at extracting common objects from multiple images. It is usually formulated as an energy minimization problem with the consistency constraint of the foregrounds. This process is challenging due to the variations of foregrounds. Several cosegmentation strategies are proposed, which can be classified into single group- and multiple group-based cosegmentation.

Single group-based cosegmentation [43]–[46] considers a set of images and is usually formulated as an energy minimization problem. The energy consists of two terms, namely, the segmentation term and the foreground consistency term. The segmentation term is formulated as a traditional single-image-based segmentation model, such as Markov random fields (MRF), active contours, and spectral clustering-based segmentation models that segment foregrounds from backgrounds. The foreground consistency term is used to force the foregrounds to be similar. Several foreground consistency terms are proposed to consider the trade-off between the similarity measurement and the energy minimization. The successful segmentation is obtained when the backgrounds are different. However, when the images contain similar backgrounds, the performance is dramatically reduced. Although additional priors, such as the saliency and objectness priors, are used to avoid the interference of similar backgrounds, cosegmentation by the single-image group is still challenging.

In contrast, multiple group cosegmentation considers multiple sets of images, such as groups of simple images and complicated images. The multiple group cosegmentation has two advantages. First, the intergroup cue can be used to enhance the cosegmentation. Second, different types of multiple group images can be collected to provide additional discriminative cues. For example, in [39], the simple image group is used to help the cosegmentation of the complicated image group. The image groups containing different noise images are combined to obtain the noise images. Multiple group cosegmentation has shown superior performance over single group cosegmentation. However, the existing multiple group cosegmentation focuses on only one class, such as multiple sets of images of a "Cat". However, in many applications such as semantic segmentation, the multiple groups are usually related to different classes, such as "Cat", "Bird" and "Car". Here, we consider the multiple group cosegmentation of different classes and name it class-level multiple group cosegmentation.

C. Co-Saliency Detection

Another related work is co-saliency detection [47]–[50], which aims at locating common salient objects from multiple images, and is similar to cosegmentation. The essential step of co-saliency is how to efficiently locate common salient regions between images. Some strategies have been proposed. For example, Cong *et al.* [47] use depth map as additional cue to formulate three constraints such as the similarity constraint, the cluster-based constraint and the saliency consistency to locate the common salient regions. Wang *et al.* [48] use a two-branch-based network to simultaneously learn high-level group-wise semantic representation and deep visual features, which are then combined to form top-down semantic guidance to improve common salient region detection. Zhang *et al.* [49] use multiple instance learning (MIL) to learn the discriminative classifiers, and measure the intra-image contrast and the inter-image similarity to find common salient regions. Cao *et al.* [50] propose a method to generate co-saliency map by combing saliency detection results of multiple saliency

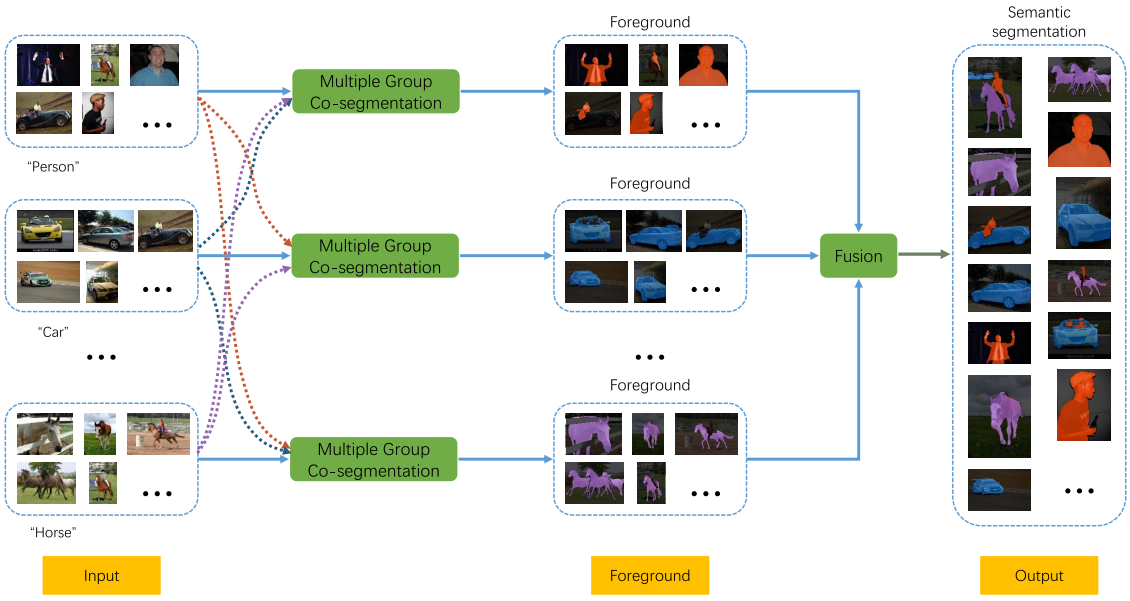


Fig. 1. The pipeline of the proposed method. Multiple group cosegmentation is first used to obtain the foreground of each image by its classification label. By this process, the inter-information and intra-information are considered. Then, the segmentation results are fused (fusion step) to form the final semantic segmentation results.

detection algorithms. The rank constraint is used to explore the relationship between different saliency maps and generate self-adaptive weights for combination.

In general, co-saliency map is used as an important cue to guide cosegmentation. For example, Fu *et al.* [51] use depth information to enhance identification of similar foreground objects via RGBD co-saliency map, and formulate cosegmentation in a fully-connected graph with mutex constraint to handle noisy images. Furthermore, Fu *et al.* [52] extend the work in [51] to deal with cosegmentation of multiple foregrounds in videos via the spatio-temporal smoothness. Hsu *et al.* [53] introduce a new instance cosegmentation task and decompose the task into two steps, of which the first step solves co-peak search by jointly optimizing co-peak, affinity and saliency losses, and the second step solves instance mask segmentation via an efficient proposal ranking algorithm. Kompella and Kulkarni [54] propose a weakly supervised multi-scale recurrent convolutional neural network for co-saliency detection and cosegmentation, which first extracts the superpixel features from representative multi-scale images, and then trains a RCNN to extract the common salient object regions, achieving faster and more accurate performance with small training dataset. Here, the CAM can also be considered as a specific co-saliency map that highlights discriminative regions among classes. We refer readers to [55], [56] for more details of co-saliency.

III. The Proposed Method

A. Overview

We aim at segmenting semantic regions from multiple sets of images by image-level labels. The overview of the proposed method is shown in Fig. 1, where two steps that consist of the multiple group cosegmentation step and the fusion step are used. The first step is to extract common regions of each class based on classification model and local-to-global segmentation model. The second step is to combine these regions to form the semantic segmentation results directly.

In the first step, given multiple images $\mathbf{I} = \{I_1, \cdots, I_n\}$ and the corresponding image-level labels $\mathbf{L} = \{\mathbf{L}_1, \cdots, \mathbf{L}_n\}$, where $\mathbf{L}_i = \{l_{i1}, \cdots, l_{i|\mathbf{L}_i|}\}$ is a set of labels (an image may contain multiple objects), we first classify the images into multiple image groups $\mathbf{C} = \{\mathbf{C}_1, \cdots, \mathbf{C}_m\}$ by the image labels $\mathbf{L}$. Then, for each group $\mathbf{C}_i$, we implement multiple group cosegmentation to obtain the foregrounds $\mathbf{F}_i = \{f_{i1}, \cdots, f_{i|\mathbf{C}_i|}\}$ of images in $\mathbf{C}_i$. Such a process can be formulated as

$$\mathbf{F}_i = H(\mathbf{C}_i|\mathbf{C}), i = 1, \cdots, m \tag{1}$$

where H is the multiple group cosegmentation, and $H(\mathbf{C}_i|\mathbf{C})$ means considering all image groups $\mathbf{C}$ for $\mathbf{C}_i$. We denote $\mathbf{F} = \{\mathbf{F}_1, \cdots, \mathbf{F}_m\}$ as the foreground set of all groups.

In the fusion step, we combine the cosegmentation foregrounds of each image I_i by their labels $\mathbf{L}_i$ and finally form the semantic segmentation results S_i. We represent such a process as

$$S_i = Fusion(\mathbf{F}|\mathbf{L}_i), i = 1, \cdots, n \tag{2}$$

B. Multiple Group Cosegmentation by Discriminative Prior

We formulate the multiple group cosegmentation as an energy minimization problem by considering the discriminative probability map and the local-to-global segmentation. The first one is used to consider the common cue and discriminative cue among image groups. Since the priors provided by the image-level labels are usually local rather than global, the second one is used to provide global priors.

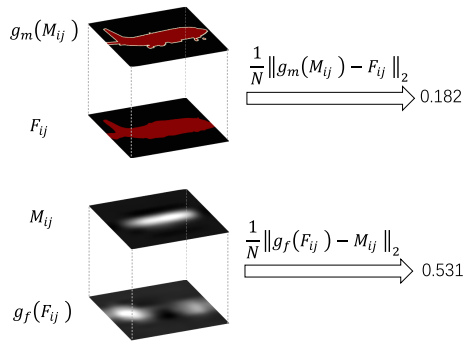


Fig. 2. Examples of the distance function. The L_2 norm is used.

1) Energy Formulation: Given multiple groups of images $\mathbf{C} = \{\mathbf{C}_1, \cdots, \mathbf{C}_m\}$ with m classes, we form the energy function as

$$E(\mathbf{F}, \mathbf{M}) = \sum_{i=1}^{m} \sum_{j=1}^{n_i} [d(F_{ij}, g_m(M_{ij})) + d(M_{ij}, g_f(F_{ij}))] \quad (3)$$

where M_{ij} is the discriminative probability map for C_{ij} (the jth image in the ith class $\mathbf{C}_i$), with the same size as the image. The value of a pixel in M_{ij} means the probability of the pixels to distinguish the image. $\mathbf{M}$ is the set of all M_{ij}. Since M_{ij} is implicitly expressed among the images and image groups, we use it as latent variable here.

F_{ij} is the foreground mask of image C_{ij}, which is our target. Functions g_m and g_f are functions that transfer the discriminative probability map to the expected segmentation region and the segmentation region to the expected discriminative probability map, respectively. The aim of introducing the transfer function is to bridge the two types of data for measuring their fitness. The idea is to capture the contextual cues from the local prior and image and generate object regions by the contextual cues. The transfer function can be trained to learn the mapping from the local prior to the object region. A CNN-based network is used to formulate the transfer function. Details can be found in Section III-B.1.b.

The function d calculates the mean Euclidean distance between the pixel values of a pair of segmentation masks or probability maps. Here, the distance between foregrounds and the expected foregrounds and the distance between the discriminative probability map and the expected discriminative probability map are considered. Fig. 2 shows examples of calculating distance d between a pair of foregrounds $(g_m(M_{ij}), F_{ij})$ and a pair of discriminative probability maps $(g_f(F_{ij}), M_{ij})$, where N is the total number of pixels of the input. $||\cdot||_2$ is the L_2 norm.

As seen from (3), our energy function measures the fitness between the foregrounds and the discriminative probability maps, and we aim at obtaining the foregrounds that have the best fitness to the discriminative probability map, i.e.,

$$\mathbf{F}^* = \arg\min_{\mathbf{F},\mathbf{M}} E(\mathbf{F}, \mathbf{M}) \quad (4)$$

The proposed energy function in (3) is designed as a sum of terms involving per image, per class and foreground segmentation and is a sum of independent terms. It is different from the traditional cosegmentation energy function that uses the foreground consistency term to form the cosegmentation task. The reasons for such a definition are twofold. First, it leads to simple optimization of the segmentation model. Second, the cosegmentation can be represented by the latent variable $\mathbf{M}$ that represents the discriminative region of the image. Since the discriminative region is captured by comparing the image with the images of all classes, it describes two essential cues of cosegmentation, such as the common cues within each image class and the discriminative cue between multiple classes. Therefore, the cosegmentation is implemented along with the searching of the latent variable $\mathbf{M}$.

We next detail our energy function and its minimization.

a) The discriminative probability map $\mathbf{M}$*:* The purpose of introducing the discriminative probability map is to capture segmentation cues from both inter-image and intra-image groups. In our method, the set of discriminative probability maps $\mathbf{M}$ is obtained by iteratively updating $\mathbf{M}$ from a set of discriminative maps. Here, the initial probability map is initialized by the class activation map of the image.

We use the method in [11] to generate the class activation map, of which the main idea is to first train the multiple-class based classification network, and then extract the region of each image by the classification network. The class activation map for class c_i is set by the weighted sum of the feature maps of the last convolution layer. The weights of the fully connected layer corresponding to the class c_i are used as the weight vector. The method [11] consists of two stages, namely, the training stage and inference stage. The training stage trains the classification model based on the image-level labels. The inference stage generates class activation maps of the new images. The probability maps are finally generated to initialize the discriminative probability maps of the test images.

However, the existing method considers each class separately while the mutual information among the classes is ignored. Here, we propose a new class activation map generation network that uses a refine-block to combine the mutual information of different classes. Our CAM generation network is shown in Fig. 3. The backbone network is first used to obtain high-level convolution features. Then, we use a 1×1 convolution layer to generate the initial class activation map. Afterward, the class activation maps are fed to the refine block to output the refined maps. Global average pooling is used to obtain the classification scores from the class activation maps for training.

The refine block consists of three branches. The first is composed of the initial class activation maps. The second and third branches consist of the batch normalization (BN) layer, ReLU layer and convolution layer followed by the BN layer. We select the size of the convolution filter as 3×3 and 1×1 for the second and third branches, respectively. The outputs of the three branches are summed to obtain the refined maps. After training the network with the classification task, the refined class activation map is used as our initial discriminative probability map.

b) The function g_m*:* The function g_m transfers the image to the segmentation region that is supported by the discriminative probability map. It is used to measure the fitness between

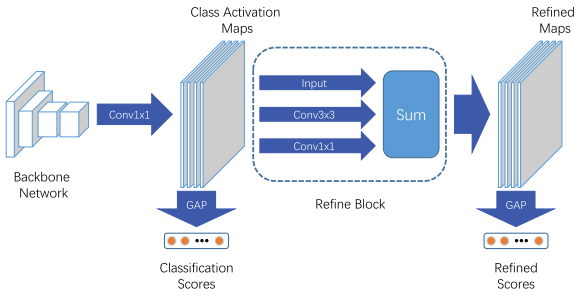


Fig. 3. The proposed class activation map generation network. The network is a special multilabel classification network, which aims to obtain better class activation maps. A coarse map is first obtained, and then, a refine subnetwork is used to obtain the final CAM.

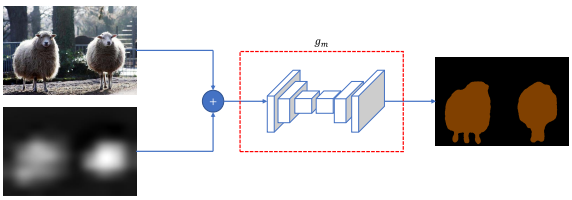


Fig. 4. The structure of the proposed function g_m. The convolutional neural network is used to transform the image and discriminative probability map to the expected segmentation mask.

the segmentation region and the discriminative probability map. Here, we also aim to use g_m to add the global priors to refine the local discriminative regions generated by the CAM method.

The function g_m is achieved by a convolutional neural network named $CNN_{local-global}$. The input $(I_i, \mathbf{M})$ is the combination of the image and its discriminative probability map. The output is the binary segmentation mask F'. We use VGG-16 as the backbone network for feature extraction. Since we lack pixel-level annotations of the given images at the test stage, we train the network $CNN_{local-global}$ offline. The implementation of function g_m based on the convolutional neural network is illustrated in Fig. 4, where the input is the image and the corresponding discriminative probability map, and the output is the foreground segmentation mask. The convolutional neural network is used to accomplish the transformation. In the training of the network, the discriminative maps are generated from the groundtruth by the g_f function.

We can use the function g_m to achieve segmentation of the object region from a local part prior, with the idea of first capturing the contextual cues from the image and then mapping local priors to object regions. To this end, we use various local priors and segmentation masks to learn the network sufficiently. In addition, since it is hard to segment object region successfully by running the network only once, an iteration process is used to refine the prior and the segmentation mask gradually. Fig. 5 displays some results of our local-to-global segmentation network, where the original images, the groundtruth, the discriminative probability maps and the corresponding binary masks in three iterations (iter0, iter1 and iter2) are displayed. It can be seen that the discriminative probability maps are improved by the iteration process, and the object regions are finally obtained.

c) The function g_f: The function g_f is used to transfer the image segmentation mask to the discriminative probability map. We propose a downsample method for such a transformation. Given a segmentation mask F_0 with the same size as the image, we first downsample the mask F_0 to a small size of mask F_s (with a size of 7×7). Then, we upsample F_s to the size of F_0 by a bilinear interpolation method and obtain the new map F_1'. Afterward, we set the value of F_1' smaller than a threshold T_f to zero and obtain the new map F_2', where $T_f = \delta \cdot \upsilon_m$ with υ_m as the maximum value of F_1' and $\delta = 0.9$ as the deletion factor. Finally, we downsample F_2' to the size of 7×7 first and then upsample to the size of F and obtain the discriminative map F'. The process of the function g_f is shown in Fig. 6.

Note that the mask sizes after employing the downsampling and upsampling operators are set to 7×7 and the original size, respectively, which are set empirically, with the aim of making the discriminative probability maps similar to the class activation maps generated by the existing CAM generation method [11].

Various discriminative probability maps are required to learn the transferring function g_m. Here, we use function g_f with different settings of δ to generate the training samples. Two steps are used. In the first step, an image and its groundtruth mask are selected randomly. Then, the discriminative probability map is generated from the mask by the function g_f with δ generated randomly. The image, the groundtruth mask and the probability map form a training sample.

2) Energy Minimization: We next introduce the energy minimization. The energy function in (3) consists of two variables $\mathbf{M}$ and $\mathbf{F}$. We therefore minimize the energy with the EM framework, i.e., iteratively updating $\mathbf{F}$ and $\mathbf{M}$ until convergence of the two variables. Two steps are used. In the first step, given an image C_{ij} and its corresponding discriminative probability map M_{ij}, we feed them to the local-to-global network $CNN_{local-global}$ and obtain the expected regions F'. Then, we update F_{ij} by $F_{ij} = F'$. In the second step, we update M_{ij} by the new foreground F_{ij} using the transformation function g_f, i.e., $M_{ij} = g_f(F_{ij})$. The two steps are implemented iteratively until the iteration number is reached. Here, we set the iteration number as three empirically.

The total energy minimization process can be represented by a convolutional neural network, as shown in Fig. 7. Specifically, given an image C_{ij}, we first obtain the initial discriminative probability map M_{ij} by our CAM generation method. Then, C_{ij} and M_{ij} are forwarded to the local-to-global segmentation network. A foreground mask $F_{ij} = CNN_{local-global}(C_{ij}, M_{ij})$ is obtained. Next, the new class activation map M_{ij}' is updated by function g_f and the foreground mask F_{ij}, i.e., $M_{ij}' = g_f(F_{ij})$, and the segmentation process is implemented again to obtain the new segmentation. These two processes are iteratively implemented until the iteration number is reached.

C. The Fusion of the Cosegmentation Results

The processes are displayed in Fig. 8. Given an image I_i, the labels $\mathbf{L}_i = \{l_{i1}, \cdots, l_{i|\mathbf{L}_i|}\}$, and the corresponding

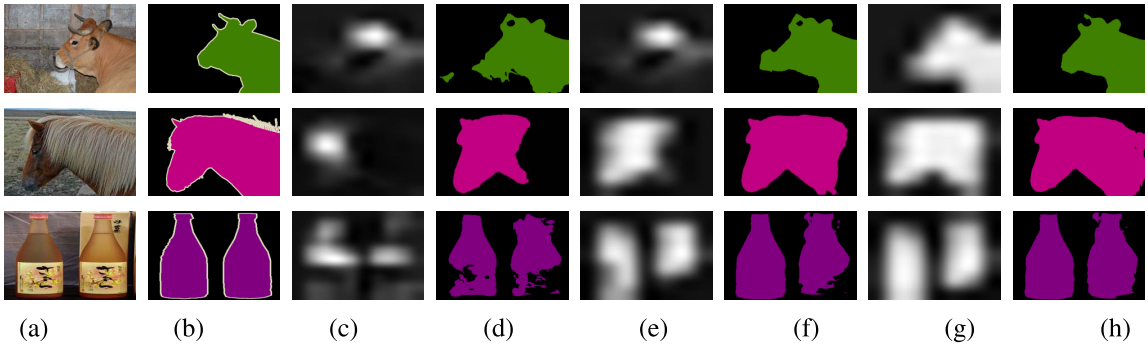


Fig. 5. Examples of binary masks generated by g_m. The original images, the groundtruths, the discriminative probability maps and the corresponding masks in three iterations are displayed from (a) to (h).

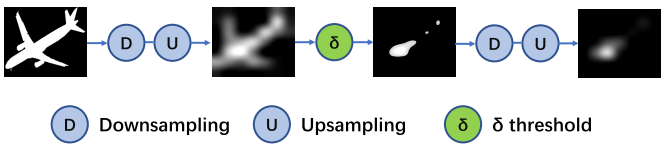


Fig. 6. An example of transferring the segmentation mask to the expected probability map by the proposed function g_f. Downsampling and upsampling are used to generate the probability map from the binary mask, while δ is a threshold that binarizes the probability map to preserve important regions.

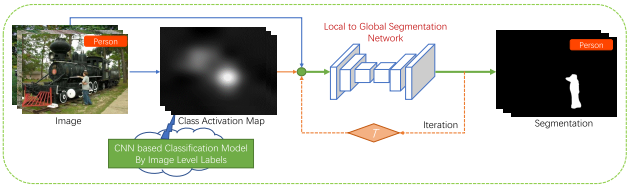


Fig. 7. The network for the proposed multiple group cosegmentation method. The class activation map is first obtained by our proposed CAM method. Then, a local-to-global segmentation network is implemented to segment the regions by the concatenation of the RGB channels and the activation map channel. In the train stage, an EM-based iteration is used to obtain the final segmentation results.

foregrounds $\hat{\mathbf{F}}_i = \{\hat{f}_{i1}, \cdots, \hat{f}_{i|\mathbf{L}_i|}\}$, the fusion step first obtains the nonoverlapping and overlapping foregrounds by comparing the results of different classes. The nonoverlapping regions have unique labels, while the overlapping regions have more than two labels so that we need to determine the label further. Here, we reuse the discriminative probability maps as used in the cosegmentation step to determine the labels by comparing the probability values, as shown in Fig. 8.

Let an image and its labels be I_i and $\mathbf{L}_i = \{l_{i1}, \cdots, l_{i|\mathbf{L}_i|}\}$; the discriminative probability maps and cosegmentation foregrounds for each label are denoted as $\hat{\mathbf{M}}_i = \{\hat{M}_{i1}, \cdots, \hat{M}_{i|\mathbf{L}_i|}\}$ and $\hat{\mathbf{F}}_i = \{\hat{F}_{i1}, \cdots, \hat{F}_{i|\mathbf{L}_i|}\}$, respectively. The foregrounds are first classified into certain (nonoverlapping) R_c and uncertain regions (overlapping regions) R_u (as shown in Fig. 8). The pixels in R_c are assigned their unique label, and the label result is denoted as L_c. For each pixel p_k in the overlapping regions, we assign the labels with the larger probability value. In our network, we first obtain the label map M^l for all pixels by the max operator. i.e.,

$$M^l(p_k) = l_{ij^*}, if \quad j^* = \arg\max_j \hat{M}_{ij}(p_k) \tag{5}$$

Then, we use the labels in M^l as the pixel labels in region R_u and denote the result as L_u. Finally, we combine L_c and L_u to form the final semantic segmentation result.

IV. Experimental Results

We next verify the proposed method. The experiments include the experimental setup, subjective results and objective results on the PASCAL VOC 2012 [57] and MS COCO 2017 [58] datasets. Then, the performance with different δ parameters, and the comparisons with cosegmentation methods, co-saliency detection methods, and CAM generation methods are presented. The inference speeds are also discussed.

A. PASCAL VOC 2012 Dataset

1) Experimental Setup: We first introduce the experimental setup of our experiments, including the images for verification, and the settings for training the classification network and the local-to-global segmentation network. The proposed method is verified on the validation and test dataset of the PASCAL VOC 2012 dataset that contains images of 20 classes. Both the images and the corresponding image-level labels are used as inputs. Three steps are implemented to segment the regions from the test images semantically. The first step classifies the test images into 20 image groups according to their image-level labels and trains the classification network to obtain the classification model using the images of 20 classes. Then, the class activation map is extracted from the classification network, and the local-to-global network is used to obtain the foregrounds in an iteration manner. The third step combines the foregrounds of cosegmentation to form the final semantic segmentation results. For the image containing multiple classes of objects such as c_i and c_j, we put it into both image groups of $\mathbf{C}_i$ and $\mathbf{C}_j$ and therefore can obtain the foreground of each class separately.

We train two networks such as the classification network CNN_c and the local-to-global segmentation network $CNN_{local-global}$. For the first network, the backbone is set to ResNet-101 initialized by the ImageNet dataset. The size of the image is set to 320×320 for input. In the training, the batch size is set to 32, and the number of epochs is set to 40. The learning rate is dynamically decreased by the number of epochs using the logspace function.

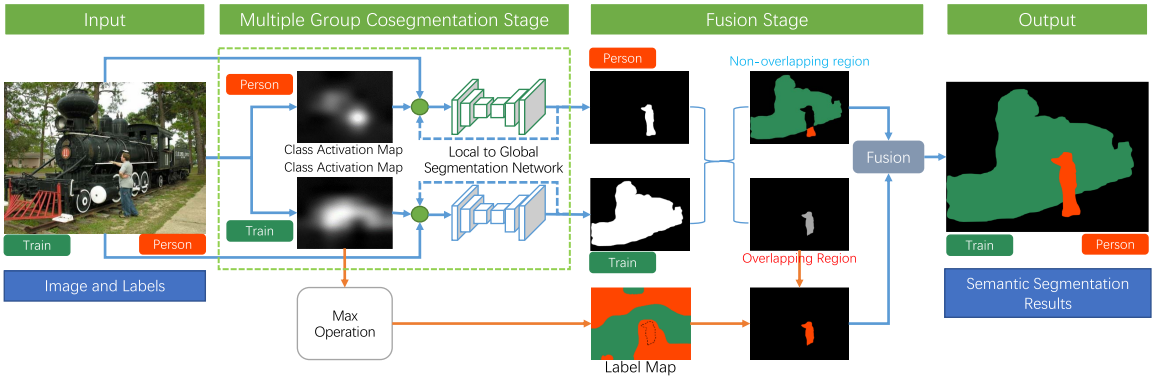


Fig. 8. The detailed flowchart of the proposed method on a specific image. In the multi-group cosegmentation stage, the cosegmentation results are obtained according to the labels "train" and "person". In the fusion stage, the foreground segmentation results are divided into the nonoverlapping region and the overlapping region. The image-level labels are assigned directly to the nonoverlapping region. The labels of overlapping regions are determined via the value of the class activation maps. Finally, the multiple foreground segmentation regions are combined to form the semantic segmentation result.

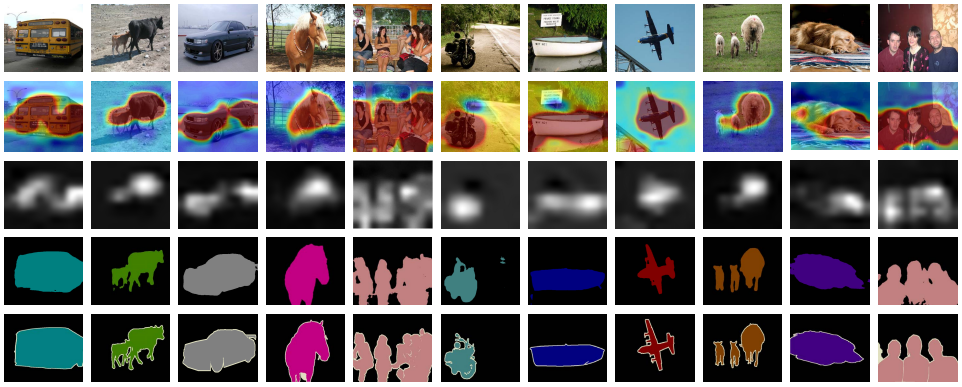


Fig. 9. The segmentation results achieved with our method. The original images containing a single class of an object, the discriminative probability map from the proposed class activation extraction network, our semantic segmentation results, and the groundtruth are displayed.

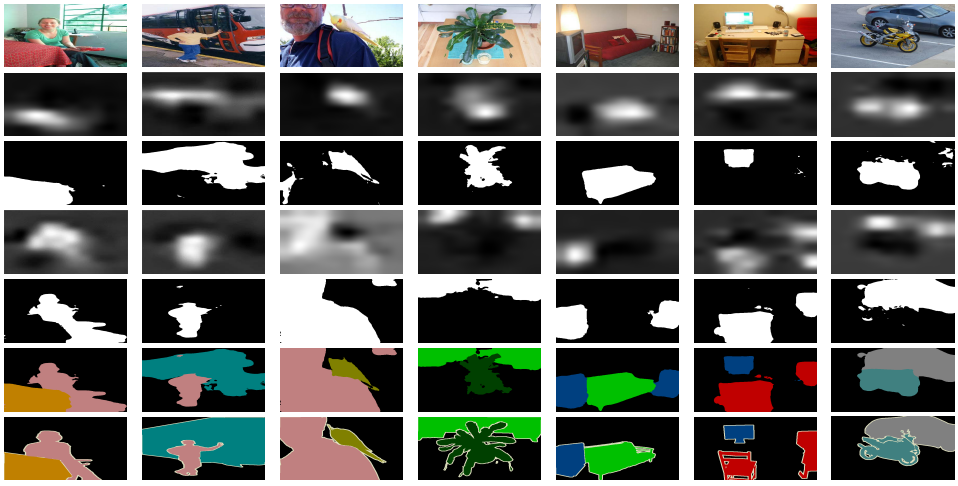


Fig. 10. Our semantic segmentation results on images containing two class of objects. The activation maps of the two classes, the corresponding segmentation results from local-to-global segmentation, the semantic segmentation results, and the groundtruths are displayed.

Since the classification network trained on the other image datasets can also be used to extract CAM of the test images, different types of modes are established according to the images used for training. We name them as $\mathbf{D}_c \rightarrow \mathbf{D}_t$, where $\mathbf{D}_c$ represents the set of images used for training the classification network, and $\mathbf{D}_t$ represents the set of test images. For example, $\{val, training\} \rightarrow \{val\}$ is the mode that extracts the CAMs of the images in the validation dataset based on the classification network trained on the images of the validation and training dataset. In our method, six types of

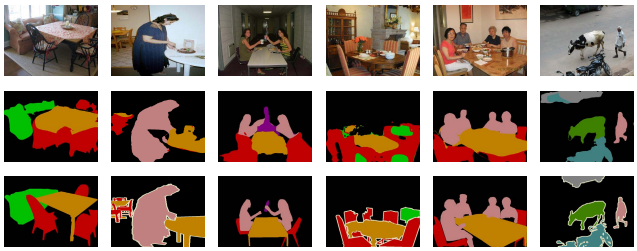


Fig. 11. The segmentation results of multiple classes on the PASCAL VOC 2012 dataset by our method. The first row, second row and third row show the original images, the segmentation results and the groundtruths, respectively.

modes, namely, $\{test\} \rightarrow \{test\}$, $\{test, training\} \rightarrow \{test\}$, $\{training\} \rightarrow \{test\}$, $\{val\} \rightarrow \{val\}$, $\{val, training\} \rightarrow \{val\}$ and $\{training\} \rightarrow \{val\}$, are considered for the PASCAL VOC 2012 dataset.

The local-to-global segmentation network is pretrained by the images in the MS COCO 2017 dataset. The images of all the classes except the 20 classes in the PASCAL VOC 2012 dataset are used for training. VGG-16 is used as the backbone network. The training data are composed of the training images and the discriminative probability maps generated from the segmentation groundtruth using function g_f, where the deletion factor δ is set randomly from $[0.6, 0.95]$. The batch size and the learning rate are set to 10 and 0.0001, respectively. Since the regions with sizes that are extremely small and large reduce the segmentation performance, we filter the extreme training samples by setting $v_t < 0.1$ or $v_t > 0.95$, where $v_t = \frac{A_b}{A_a}$ with A_b the area of the object region and A_a the image region.

2) Subjective Results: We next show the subjective results of our method based on the number of classes contained in the images. The subjective results are displayed in Fig. 9, Fig. 10 and Fig. 11, where the images with the single class, two classes and more than two classes are displayed, respectively. In Fig. 9, the original images, the corresponding activation maps, the segmentation results by our local-to-global segmentation network and the groundtruths are displayed. It is seen that the activation maps usually contain a local part of the objects. Meanwhile, our method successfully segments the objects from these images, which demonstrates the effectiveness of our local-to-global segmentation network. It is also seen that our method successfully segments the object with multiple instances from the images, which further demonstrates the effectiveness of the proposed method.

In Fig. 10, the results of images containing two classes are displayed. It is seen that the object regions of two classes are highlighted by the class activation maps successfully. It is also seen that the semantic segmentation results are obtained from these images successfully.

In Fig. 11, the results of images with more than two classes are displayed, including the original images, the semantic segmentation results and the groundtruths. It is seen that the regions of multiple classes are successfully segmented from these images.

3) Objective Results: We next display the objective results. The intersection-over-union (IoU) value is used for the verification, which is defined as

$$IoU = \frac{Seg \cap GT}{Seg \cup GT} \tag{6}$$

where Seg and GT are the segmentation region and the annotation region, respectively. The larger the IoU value is, the better the segmentation performance is. For multiple images, the mean IoU value (mIoU) is used for verification.

The mIoU values of the test and validation dataset are displayed in Table I. Six types of modes are considered. For each mode, the mIoU values of each class and their mIoU value are shown. "iter$i$" means the results under the ith iteration.

As seen from Table I, the best mIoU values are 60.3 % and 56.4 % for the test dataset and validation dataset, respectively. It is also seen that the mIoU values with the training dataset are obviously better than those without the training dataset (60.3 vs 57.1 and 56.4 vs 52.9 for the test and validation dataset, respectively). This result indicates that better activation maps can be extracted by more images. Moreover, by comparing the results under different iterations, the segmentation can be obviously improved by the second iteration, which indicates that a small number of iteration can be used. Hence, we set the iteration number to three.

We also show the mIoU values by the local-to-global segmentation network trained from the training dataset of PASCAL VOC 2012 (Ours+PASCAL) rather than MS COCO 2017 dataset (Ours+COCO) for comparison. The mIoU values of Ours+PASCAL are shown in Table II. It is seen that the segmentation performances are improved (63.7 vs 60.3 and 58.6 vs 56.4 for the test and validation dataset, respectively). Therefore, domain adaptation can be used to improve our method.

We next display the mIoU values of the existing weakly supervised semantic segmentation methods for comparison. The mIoU values are displayed in Table III. It is seen that our method obtains 56.4 and 60.3 for the validation set and test set, respectively, which outperforms these existing one-stage-based and two-stage-based weakly supervised semantic segmentation methods. Note that the results are obtained without CRF post-processing. This result further demonstrates the effectiveness of the proposed method.

B. MS COCO 2017 Dataset

1) Experimental Setup: We next evaluate the proposed method on the MS COCO 2017 dataset. The validation dataset that contains 5k images is used for verification. The ResNet-101 initialized by the ImageNet dataset is used to train the classification network. The image size, batch size and training epochs are set to 320×320, 32 and 40, respectively. The learning rate is dynamically decreased by the number of epochs using the logspace function. We set δ to 0.9. The mode of the experiment is $\{val\} \rightarrow \{val\}$.

2) Subjective Results: The subjective results are displayed in Fig. 12, including the images, the segmentation results and the groundtruths. The images containing different numbers of classes are shown. It is seen that the object regions with a large size can be successfully segmented, such as "Aeroplane" and

TABLE I

THE MIOU VALUES BY THE PROPOSED METHOD. THE MODE MEANS THE DATASET USED FOR TRAINING THE CLASSIFICATION MODEL FOR ACTIVATION MAP GENERATION. THE LOCAL-TO-GLOBAL NETWORK IS TRAINED BY THE MS COCO 2017 DATASET EXCEPT FOR THE 20 CLASSES IN THE PASCAL VOC 2012 DATASET

mode: $\{test\} \rightarrow \{test\}$																						
Iteration	bg	aeroplane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mIoU
Iter0	82.1	45.3	33.3	55.2	29.4	51.4	71.1	64.2	71.7	21.4	78.7	43.0	70.9	79.9	75.6	48.4	30.7	77.2	55.3	47.5	41.9	55.9
Iter1	84.1	49.8	32.3	59.0	31.6	51.9	72.3	66.1	72.4	22.0	79.4	38.7	72.8	82.5	74.3	51.6	31.2	76.3	56.2	51.8	41.9	**57.1**
Iter2	84.2	50.1	31.8	58.4	32.3	50.8	72.4	66.0	72.2	21.4	79.4	37.4	72.6	83.0	72.8	51.3	31.4	75.6	56.0	53.5	41.1	56.8
mode: $\{train\} \rightarrow \{test\}$																						
Iter0	83.0	55.9	33.1	58.4	32.8	47.4	77.2	63.8	70.8	21.0	76.8	39.5	73.5	78.5	69.8	51.4	26.4	76.0	49.5	52.0	39.0	56.0
Iter1	86.8	63.1	33.1	65.7	40.4	54.2	74.9	68.6	74.6	23.1	79.5	34.4	76.4	82.1	67.8	54.4	27.1	77.9	55.1	58.2	38.9	**58.9**
Iter2	86.6	62.3	32.1	65.0	40.4	54.4	74.7	68.4	74.7	22.5	79.8	33.1	75.8	82.1	67.1	53.5	26.8	77.7	54.9	58.3	38.7	58.5
mode: $\{test, train\} \rightarrow \{test\}$																						
Iter0	83.5	57.0	32.7	54.8	35.7	53.1	80.3	69.5	71.7	25.6	78.9	48.8	73.0	80.8	72.6	56.6	32.7	75.2	55.7	49.7	45.8	58.7
Iter1	85.4	62.8	32.5	59.3	39.3	54.0	80.9	69.9	72.9	26.9	80.7	46.9	75.6	83.1	72.9	57.1	32.1	72.3	57.2	53.2	47.8	**60.3**
Iter2	85.4	64.6	32.0	59.6	40.4	53.3	80.9	69.8	72.7	26.7	81.0	45.5	75.0	82.5	71.0	56.4	30.7	75.7	56.3	54.9	46.8	60.1
mode: $\{val\} \rightarrow \{val\}$																						
Iter0	79.3	49.2	23.6	48.4	32.2	48.6	66.0	53.8	68.7	20.3	73.7	40.2	64.8	61.2	60.9	48.0	29.4	68.6	46.9	51.9	43.2	51.4
Iter1	81.5	53.0	23.5	51.1	34.8	48.3	66.7	56.8	71.2	20.4	73.4	38.0	66.7	64.6	62.1	50.0	29.8	69.4	49.3	54.0	46.0	**52.9**
Iter2	81.7	52.8	23.3	50.7	35.8	49.0	66.7	57.7	71.4	20.1	72.8	37.9	66.3	64.9	61.9	49.6	29.0	69.5	49.6	54.9	45.6	52.9
mode: $\{train\} \rightarrow \{val\}$																						
Iter0	78.9	52.2	24.0	49.0	31.7	45.5	69.6	56.2	68.3	19.6	72.5	32.7	65.3	61.7	60.9	48.1	24.6	66.1	46.5	58.5	41.9	51.1
Iter1	83.8	58.9	25.1	56.3	42.8	48.1	72.1	59.3	73.0	23.1	74.8	32.4	69.2	65.3	62.5	51.8	26.3	69.2	51.2	62.2	42.4	**54.8**
Iter2	83.8	58.8	25.0	54.9	43.4	47.9	72.2	59.1	73.1	22.9	74.7	31.6	69.0	65.9	62.3	50.9	26.1	68.8	51.1	62.3	41.9	54.6
mode: $\{val, train\} \rightarrow \{val\}$																						
Iter0	81.5	52.6	22.8	49.9	34.5	47.8	70.8	60.6	67.9	26.3	74.2	46.1	68.0	64.5	61.1	54.6	31.0	67.9	55.2	60.1	45.5	54.4
Iter1	83.6	57.1	23.2	52.8	38.7	48.2	72.8	64.1	70.6	26.4	73.9	46.1	70.5	67.2	62.7	55.7	31.6	69.9	57.8	62.8	47.1	56.3
Iter2	83.8	57.2	23.2	52.6	40.1	48.2	72.6	64.4	71.2	26.3	73.8	45.3	70.1	67.2	62.5	54.9	31.2	70.0	58.1	63.8	47.2	**56.4**

TABLE II

THE MIOU VALUES THAT ARE SIMILAR TO TABLE I EXCEPT FOR THE LOCAL-TO-GLOBAL NETWORK. THE LOCAL-TO-GLOBAL NETWORK IS TRAINED BY THE PASCAL VOC 2012 TRAINING DATASET

mode:$\{test\} \rightarrow \{test\}$																						
Iteration	bg	aeroplane	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mIoU
Iter0	83.6	65.9	29.7	55.9	33.0	39.5	71.7	62.2	68.3	17.9	76.7	49.3	68.4	81.1	75.8	48.0	34.2	73.3	49.8	59.9	43.8	56.6
Iter1	87.1	71.4	34.0	63.3	38.2	42.5	72.5	65.7	70.9	19.2	80.8	48.7	72.3	84.1	78.2	52.8	38.9	78.0	52.0	68.3	46.1	60.2
Iter2	87.7	72.5	35.2	65.3	41.6	42.7	72.4	66.7	71.1	19.6	81.8	47.5	72.5	84.4	78.6	53.7	40.1	78.8	51.6	69.6	46.7	**61.0**
mode: $\{train\} \rightarrow \{test\}$																						
Iter0	86.9	68.5	37.3	64.5	44.2	44.9	80.1	69.0	69.6	21.0	79.2	45.3	73.7	79.8	78.2	54.1	32.5	81.1	49.2	63.7	41.3	60.2
Iter1	89.8	71.7	36.9	68.6	51.6	51.2	80.2	71.6	73.2	23.4	80.9	47.5	75.2	81.9	79.4	60.5	38.7	82.6	52.7	69.3	46.3	**63.5**
Iter2	89.6	66.3	37.1	68.3	49.8	50.7	80.1	71.1	73.2	23.5	81.1	47.3	74.1	82.4	79.0	60.6	38.7	80.8	52.7	79.0	46.3	63.0
mode: $\{test, train\} \rightarrow \{test\}$																						
Iter0	86.1	69.9	32.0	57.4	41.5	42.6	78.9	70.9	68.6	22.3	77.6	52.1	70.8	77.9	76.6	55.4	37.1	75.0	51.4	62.3	46.9	59.7
Iter1	88.8	75.2	36.3	64.3	49.4	45.6	80.1	73.0	71.0	23.9	81.0	51.9	73.3	81.4	78.0	59.3	40.9	79.3	53.6	70.8	49.3	63.2
Iter2	89.2	76.2	37.2	65.6	51.7	46.2	80.0	73.2	71.3	24.5	81.5	51.1	73.3	81.7	78.2	60.2	41.9	80.3	53.3	72.6	48.8	**63.7**
mode: $\{val\} \rightarrow \{val\}$																						
Iter0	80.4	57.8	22.1	50.1	35.4	36.7	69.6	55.5	69.6	23.9	76.7	42.0	61.1	66.4	63.5	49.7	24.1	72.0	41.7	63.4	37.5	52.4
Iter1	84.5	63.7	24.7	56.9	43.4	37.9	71.2	59.6	73.0	25.3	79.2	40.4	65.3	70.7	67.0	54.2	25.9	76.0	45.0	68.7	42.6	56.0
Iter2	85.3	65.1	25.3	57.2	46.3	38.8	70.9	60.8	73.5	25.5	79.7	39.1	66.1	71.5	67.5	55.2	25.9	76.8	45.7	70.1	43.6	**56.7**
mode: $\{train\} \rightarrow \{val\}$																						
Iter0	83.4	64.5	27.2	55.3	35.6	43.0	72.0	62.3	68.9	23.2	78.0	36.2	68.3	67.3	63.3	50.5	24.3	75.6	43.7	66.4	46.2	55.0
Iter1	86.8	67.9	27.6	54.9	49.5	47.0	74.5	66.7	74.4	25.6	79.2	35.1	71.7	69.4	65.8	54.6	25.9	77.8	47.1	69.1	51.0	**58.2**
Iter2	86.8	67.5	27.5	54.5	49.9	46.6	74.4	66.7	74.4	25.6	79.4	34.3	71.8	69.8	66.0	54.7	26.0	77.4	47.1	69.5	50.0	58.1
mode: $\{val, train\} \rightarrow \{val\}$																						
Iter0	83.0	59.3	24.3	52.3	40.4	36.3	73.0	60.7	67.2	23.3	75.7	46.3	66.2	67.6	63.5	52.9	29.8	74.8	49.9	64.2	46.0	55.1
Iter1	85.9	63.7	26.3	58.6	47.3	38.3	74.8	65.0	71.0	25.0	78.0	44.4	69.8	70.2	66.2	56.1	31.0	77.3	52.4	70.4	49.0	58.1
Iter2	86.4	64.4	27.0	59.1	50.1	38.8	74.7	65.7	71.4	25.2	78.4	43.0	69.9	70.6	66.5	56.5	30.4	77.7	52.2	72.3	49.3	**58.6**

"Sheep" in the first and second image. However, the proposed method fails to segment the regions of small objects, such as "Person" in the sixth image. This occurs because small objects may be dropped by the pooling operation. We will further study segmenting small objects in the future.

3) Objective Results: The mIoU values of the proposed method on the MS COCO 2017 dataset are shown in Table IV, where the mIoU value of our method is 28.1. The results of the existing methods such as SEC [14], BFBP [59] and DSRG [60] are also displayed for comparison. The method of SEC [14] learns the segmentation network from weak image-level labels by multiple losses correlated to the seed, region expansion and boundary fitness. The method of BFBP [59] proposes a weakly supervised semantic segmentation method based on prior generation, region binarization and CRF processing. The method of DSRG [60] expands regions from the region seeds

TABLE III

THE MIOU VALUES ON THE PASCAL VOC 2012 DATASET. OURS AND OURS+TRAINING INDICATE THE SEGMENTATION WITHOUT AND WITH THE TRAINING DATASET, RESPECTIVELY

Method	CRF	Validation	Test
SEC [14]	Yes	50.7	51.7
TransferNet [15]	Yes	52.1	51.2
C-BT-S [23]	Yes	52.8	53.7
Built-in [34]	Yes	44.8	45.8
Two-Phase [13]	Yes	53.1	53.8
Multi-Evidence [35]	Yes	–	55.6
Adversarial Erasing [7]	Yes	55.0	55.7
TMWL [16]	Yes	55.3	56.8
DSCM [17]	Yes	44.1	45.1
Ours+COCO	No	56.4	60.3
Ours+PASCAL	No	**58.6**	**63.7**

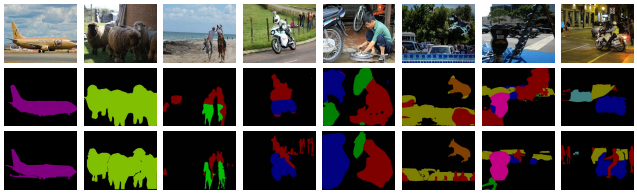


Fig. 12. The subjective results of the proposed method on the MS COCO 2017 dataset. The input images, the segmentation results, and the groundtruths are displayed. The images containing different numbers of classes are shown.

TABLE IV

THE MIOU VALUES OF THE PROPOSED METHOD AND EXISTING METHODS ON THE MS COCO 2017 VAL DATASET

Method	mIoU
SEC [14]	22.4
BFBP [59]	20.4
DSRG [60]	26.0
Ours	**28.1**

iteratively to obtain the object regions. It is seen that the proposed method outperforms these comparison methods.

C. Discussion

We display the results of our method under different settings of the parameter δ, which is used in the iteration process to generate discriminative probability maps. The results are displayed in Table V, with mode $\{train, val\} \rightarrow \{val\}$. The values of δ are shown in the first column. The column of iter$i$ displays the results of the ith iteration. It is seen that the best mIoU value is obtained by $\delta = 0.9$ as used in our experiment. Meanwhile, it is seen that comparable results can be obtained by setting $\delta > 0.8$, and the mIoU values are very close for $\delta > 0.8$, which demonstrates the robustness of our method to the setting of δ.

We next compare our method with cosegmentation and saliency detection methods. To compare fairly with these methods, we re-calculate the mIoU based on the criteria of cosegmentation which calculates the value based on foregrounds of each class. The results are displayed in Table VI, where five methods such as DCSM [61], STC [62], ICSC [63], ICSG [64] and DOCS [65] are considered. The methods

TABLE V

THE RESULTS OF THE PROPOSED METHOD WITH DIFFERENT SETTINGS OF δ. THE PASCAL VOC 2012 VALIDATION DATASET AND THE MODE OF $\{train, val\} \rightarrow \{val\}$ ARE CONSIDERED

δ	Iter0	Iter1	Iter2
0.60	54.4	49.3	46.2
0.65	54.4	51.8	50.0
0.70	54.4	53.4	52.4
0.75	54.4	54.4	54.0
0.80	54.4	55.4	55.2
0.85	54.4	56.1	56.1
0.90	54.4	56.3	**56.4**
0.95	54.4	55.6	55.2

TABLE VI

THE MIOU VALUES OF THE PROPOSED METHOD AND EXISTING COSEGMENTATION, SALIENCY DETECTION METHODS ON PASCAL VOC 2012 VALIDATION DATASET

Method	Annotations level	mIoU
DCSM [61]	Image-level	44.1
STC [62]	Image-level	49.8
ICSC [63]	Non	45.0
ICSG [64]	Pixel-level	56.0
DOCS [65]	Pixel-level	**64.5**
Ours	Image-level	57.2

TABLE VII

THE MIOU VALUES AND LOCALIZATION ERROR OF CAM BY OUR METHOD AND THE COMPARISON METHODS

Method	mIoU	loc error
Grad-cam++ [66]	22.9	71.9
Aerial-cam [67]	27.0	64.1
Ours	**35.5**	**55.0**

of DCSM [61] and STC [62] are weakly-supervised single image based saliency detection methods. The method of ICSC [63] is unsupervised co-saliency method, and the methods of ICSG [64] and DOCS [65] are cosegmentation methods aided by pixel-level annotations. The mIoU values presented in these papers are used for comparison. The mIoU value of our method is 57.2, which outperforms these comparison methods except DOCS that uses pixel-level annotations. This demonstrates the effectiveness of the proposed method.

We also compare the proposed CAM generation method with other CAM-like methods under two evaluation metrics such as mIoU and mean localization error (loc error). The larger mIoU value and lower loc error mean better performance. To compute the mIoU value between CAM map and segmentation groundtruth, we normalize the CAM map to binary mask by threshold 0.15 as usually used in CAM verification. As can be seen from the Table VII, the mIoU value of the proposed CAM network is 35.5, which is better than the methods of Grad-cam++ (22.9) [66] and Aerial-cam (27.0) [67]. Meanwhile, the loc error of the proposed CAM network is 55.0, which is lower than Grad-cam++ (71.9) [66] and Aerial-cam (64.1) [67]. This further demonstrates the effectiveness of the proposed CAM generation method.

We next evaluate the algorithmic complexity of the proposed method. Let T_{vgg} and $\bar{n}$ be the running time of the VGG-16

network and the average number of classes in each image, respectively. The algorithmic complexity of the proposed method is $T_{vgg} \cdot N + T_{vgg} \cdot \bar{n} \cdot 3N = (1 + 3\bar{n})T_{vgg}N$, where N is the number of images, $T_{vgg} \cdot N$ is the cost of extracting the class activation map, and $T_{vgg} \cdot \bar{n} \cdot 3N$ is the cost of the local-to-global segmentation network with three as the iteration number. By treating $\bar{n}$ as a constant value, the algorithmic complexity of the proposed method is $O(T_{vgg} \cdot N)$, which is similar to many existing supervised semantic segmentation methods such as FCN and DeepLabv3 when using VGG-16 as the backbone network for a fair comparison. In practical applications, the running time of the proposed method is 0.21 s per image (0.09 s for CAM generation and 0.04 s per segmentation iteration) based on a 1080ti GPU, while the inference time of the FCN network is 0.09 s per image and is faster than the proposed method. This result occurs because the VGG-16 network is implemented multiple times in our method. Meanwhile, the inference time of local-to-global segmentation is 0.04 s per image and per iteration, which is smaller than that of FCN. The reason is that the binary segmentation task considered in our method has a fewer parameters for convolution than that of FCN, which considers the classification of 20 classes.

V. Conclusion

This paper proposes a new cosegmentation and fusion-based strategy for weakly supervised semantic segmentation, which can sufficiently use the labels of both the training and testing images and avoid the drawbacks of the local priors and rough pseudo-annotations that appear in the traditional weakly supervised semantic segmentation method. A new multiple group cosegmentation by considering discriminative region extraction and local-to-global segmentation is proposed. Two subnetworks that include the discriminative region extraction network and the local-to-global segmentation network are proposed to form the multiple group cosegmentation network. A simple fusion method that considers the class activation map is finally proposed to form the semantic segmentation. We verify the proposed method on the PASCAL VOC 2012 and MS COCO 2017 datasets. The experimental results demonstrate that our method can obtain larger mIoU values than those of the existing weakly supervised semantic segmentation methods.

References

[1] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[2] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 3431–3440.

[3] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Conput. Vis. Pattern Recognit. (CVPR)*, Jun. 2017, pp. 2881–2890.

[4] S. Hong, H. Noh, and B. Han, "Decoupled deep neural network for semi-supervised semantic segmentation," 2015, *arXiv:1506.04924*. [Online]. Available: http://arxiv.org/abs/1506.04924

[5] N. Souly, C. Spampinato, and M. Shah, "Semi supervised semantic segmentation using generative adversarial network," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Jun. 2017, pp. 5688–5696.

[6] W. Wang, J. Shen, H. Sun, and L. Shao, "Video co-saliency guided co-segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 8, pp. 1727–1736, Aug. 2017.

[7] 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 *Proc. CVPR*, Jul. 2017, pp. 6488–6496.

[8] F. Meng, H. Li, Q. Wu, B. Luo, C. Huang, and K. N. Ngan, "Globally measuring the similarity of superpixels by binary edge maps for superpixel clustering," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 4, pp. 906–919, Apr. 2018.

[9] Z. Li, Y. Gan, X. Liang, Y. Yu, H. Cheng, and L. Lin, "RGB-D scene labeling with long short-term memorized fusion model," 2016, *arXiv:1604.05000*. [Online]. Available: http://arxiv.org/abs/1604.05000

[10] Y. Wei *et al.*, "STC: A simple to complex framework for weakly-supervised semantic segmentation," 2015, *arXiv:1509.03150*. [Online]. Available: http://arxiv.org/abs/1509.03150

[11] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," 2015, *arXiv:1512.04150*. [Online]. Available: http://arxiv.org/abs/1512.04150

[12] K. K. Singh and Y. J. Lee, "Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization," 2017, *arXiv:1704.04232*. [Online]. Available: http://arxiv.org/abs/1704.04232

[13] D. Kim, D. Cho, D. Yoo, and I. S. Kweon, "Two-phase learning for weakly supervised object localization," 2017, *arXiv:1708.02108*. [Online]. Available: http://arxiv.org/abs/1708.02108

[14] A. Kolesnikov and C. H. Lampert, "Seed, expand and constrain: Three principles for weakly-supervised image segmentation," 2016, *arXiv:1603.06098*. [Online]. Available: http://arxiv.org/abs/1603.06098

[15] S. Hong, J. Oh, H. Lee, and B. Han, "Learning transferrable knowledge for semantic segmentation with deep convolutional neural network," in *Proc. IEEE Conf. Conput. Vis. Pattern Recognit. (CVPR)*, Jul. 2016, pp. 3204–3212.

[16] K. Li, Z. Wu, K. Peng, J. Ernst, and Y. Fu, "Tell me where to look: Guided attention inference network," 2018, *arXiv:1802.10171*. [Online]. Available: http://arxiv.org/abs/1802.10171

[17] W. Shimoda and K. Yanai, "Distinct class-specific saliency maps for weakly supervised semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 218–234.

[18] J. Dai, K. He, and J. Sun, "Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation," 2015, *arXiv:1503.01640*. [Online]. Available: http://arxiv.org/abs/1503.01640

[19] Q. Hou, P. K. Dokania, D. Massiceti, Y. Wei, M. Cheng, and P. H. S. Torr, "Mining pixels: Weakly supervised semantic segmentation using image labels," 2016, *arXiv:1612.02101*. [Online]. Available: http://arxiv.org/abs/1612.02101

[20] M. Tang, A. Djelouah, F. Perazzi, Y. Boykov, and C. Schroers, "Normalized cut loss for weakly-supervised CNN segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Apr. 2018, pp. 1818–1827.

[21] S. Hong, D. Yeo, S. Kwak, H. Lee, and B. Han, "Weakly supervised semantic segmentation using web-crawled videos," 2017, *arXiv:1701.00352*. [Online]. Available: http://arxiv.org/abs/1701.00352

[22] S. Kwak *et al.*, "Weakly supervised semantic segmentation using superpixel pooling network," in *Proc. 31th AAAI Conf. Artif. Intell.*, Feb. 2017, pp. 4111–4117.

[23] A. Roy and S. Todorovic, "Combining bottom-up, top-down, and smoothness cues for weakly supervised image segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 3529–3538.

[24] T. Durand, T. Mordan, N. Thome, and M. Cord, "WILDCAT: Weakly Supervised Learning of Deep ConvNets for Image Classification, Pointwise Localization and Segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 642–651.

[25] D. Pathak, P. Krahenbuhl, and T. Darrell, "Constrained convolutional neural networks for weakly supervised segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1796–1804.

[26] A. Vezhnevets, V. Ferrari, and J. M. Buhmann, "Weakly supervised semantic segmentation with a multi-image model," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 3–11.

[27] O. Russakovsky, A. L. Bearman, V. Ferrari, and F. Li, "What's the point: Semantic segmentation with point supervision," 2015, *arXiv:1506.02106*. [Online]. Available: http://arxiv.org/abs/1506.02106

[28] P. Arbelaez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 328–335.

[29] G. Papandreou, L. C. Chen, K. P. Murphy, and A. L. Yuille, "Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2016, pp. 1742–1750.

[30] D. Lin, J. Dai, J. Jia, K. He, and J. Sun, "Scribblesup: Scribble-supervised convolutional networks for semantic segmentation," 2016, *arXiv:1604.05144*. [Online]. Available: http://arxiv.org/abs/1604.05144

[31] A. Khoreva, R. Benenson, J. H. Hosang, M. Hein, and B. Schiele, "Weakly supervised semantic labelling and instance segmentation," 2016, *arXiv:1603.07485*. [Online]. Available: http://arxiv.org/abs/1603.07485

[32] X. Qi, Z. Liu, J. Shi, H. Zhao, and J. Jia, "Augmented feedback in semantic segmentation under image level supervision," in *Proc. IEEE Conf. Eur Conf. Comput. Vis.*, Sep. 2016, pp. 90–105.

[33] P. Tokmakov, K. Alahari, and C. Schmid, "Weakly-supervised semantic segmentation using motion cues," 2016, *arXiv:1603.07188*. [Online]. Available: http://arxiv.org/abs/1603.07188

[34] F. Saleh, M. S. A. Akbarian, M. Salzmann, L. Petersson, S. Gould, and J. M. Alvarez, "Built-in foreground/background prior for weakly-supervised semantic segmentation," 2016, *arXiv:1609.00446*. [Online]. Available: http://arxiv.org/abs/1609.00446

[35] W. Ge, S. Yang, and Y. Yu, "Multi-evidence filtering and fusion for multi-label classification, object detection and semantic segmentation based on weakly supervised learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 1278–1286.

[36] S. J. Oh, R. Benenson, A. Khoreva, Z. Akata, M. Fritz, and B. Schiele, "Exploiting saliency for object segmentation from image level labels," 2017, *arXiv:1701.08261*. [Online]. Available: http://arxiv.org/abs/1701.08261

[37] A. Jiwoon and K. Suha, "Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Apr. 2018, pp. 4981–4990.

[38] H. Yu *et al.*, "Methods and datasets on semantic segmentation: A review," *Neurocomputing*, vol. 304, pp. 82–103, Aug. 2018.

[39] F. Meng, J. Cai, and H. Li, "Cosegmentation of multiple image groups," *Comput. Vis. Image Understand.*, vol. 146, pp. 67–76, May 2016.

[40] F. Meng *et al.*, "Constrained directed graph clustering and segmentation propagation for multiple foregrounds cosegmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 11, pp. 1735–1748, Nov. 2015.

[41] S. Vicente, V. Kolmogorov, and C. Rother, "Cosegmentation revisited: Models and optimization," in *Proc. Eur. Conf. Comput. Vis.*, Heraklion, Greece, Sep. 2010, pp. 465–479.

[42] B. Luo, H. Li, F. Meng, Q. Wu, and K. Ngan, "An unsupervised method to extract video object via complexity awareness and object local parts," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 7, pp. 1580–1594, Jul. 2018.

[43] C. Rother, V. Kolmogorov, T. Minka, and A. Blake, "Cosegmentation of image pairs by histogram matching-incorporating a global constraint into MRFs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2006, pp. 993–1000.

[44] L. Mukherjee, V. Singh, and C. R. Dyer, "Half-integrality based algorithms for cosegmentation of images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 2028–2035.

[45] F. Meng, H. Li, G. Liu, and K. N. Ngan, "Object co-segmentation based on shortest path algorithm and saliency model," *IEEE Trans. Multimedia*, vol. 14, no. 5, pp. 1429–1441, Oct. 2012.

[46] S. Vicente, C. Rother, and V. Kolmogorov, "Object cosegmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 2217–2224.

[47] R. Cong, J. Lei, H. Fu, Q. Huang, X. Cao, and C. Hou, "Co-saliency detection for RGBD images based on multi-constraint feature matching and cross label propagation," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 568–579, Feb. 2018.

[48] C. Wang, Z.-J. Zha, D. Liu, and H. Xie, "Robust deep co-saliency detection with group semantic," in *Proc. 33th AAAI Conf. Artif. Intell.*, Jun. 2019, pp. 8917–8924.

[49] D. Zhang, D. Meng, and J. Han, "Co-saliency detection via a self-paced multiple-instance learning framework," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 5, pp. 865–878, May 2017.

[50] X. Cao, Z. Tao, B. Zhang, H. Fu, and W. Feng, "Self-adaptively weighted co-saliency detection via rank constraint," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 4175–4186, Sep. 2014.

[51] H. Fu, D. Xu, S. Lin, and J. Li, "Object-based RGBD image co-segmentation with mutex constraint," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4428–4436.

[52] H. Fu, D. Xu, and S. Lin, "Object-based multiple foreground segmentation in RGBD video," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1418–1427, Mar. 2017.

[53] K.-J. Hsu, Y.-Y. Lin, and Y.-Y. Chuang, "Deepco3: Deep instance co-segmentation by co-peak search and co-saliency detection," in *Proc. IEEE Conf. Conput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8846–8855.

[54] A. Kompella and R. V. Kulkarni, "Weakly supervised multi-scale recurrent convolutional neural network for co-saliency detection and co-segmentation," *Neural Comput. Appl.*, to be published.

[55] D. Zhang, H. Fu, J. Han, A. Borji, and X. Li, "A review of co-saliency detection algorithms: Fundamentals, applications, and challenges," *ACM Trans. Intell. Syst. Technol.*, vol. 9, no. 4, p. 38, Feb. 2018.

[56] R. Cong, J. Lei, H. Fu, M.-M. Cheng, W. Lin, and Q. Huang, "Review of visual saliency detection with comprehensive information," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 10, pp. 2941–2959, Oct. 2019.

[57] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes challenge: A retrospective," *Int. J. Comput. Vis.*, vol. 111, no. 1, pp. 98–136, Jan. 2014.

[58] T. Y. Lin *et al.*, "Microsoft coco: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2014, pp. 740–755.

[59] F. Saleh, M. S. Aliakbarian, M. Salzmann, L. Petersson, S. Gould, and J. M. Alvarez, "Built-in foreground/background prior for weakly-supervised semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 413–432.

[60] Z. Huang, X. Wang, J. Wang, W. Liu, and J. Wang, "Weakly-supervised semantic segmentation network with deep seeded region growing," in *Proc. IEEE Conf. Conput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7014–7023.

[61] W. Shimoda and K. Yanai, "Distinct class-specific saliency maps for weakly supervised semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 218–234.

[62] Y. Wei *et al.*, "STC: A simple to complex framework for weakly-supervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2314–2320, Nov. 2017.

[63] K. R. Jerripothula, J. Cai, and J. Yuan, "Image co-segmentation via saliency co-fusion," *IEEE Trans. Multimedia*, vol. 18, no. 9, pp. 1896–1909, Sep. 2016.

[64] S. Banerjee, A. Hati, S. Chaudhuri, and R. Velmurugan, "Image co-segmentation using graph convolution neural network," in *Proc. Indian Conf. Comput. Vis., Graph. Image Process. (ICVGIP)*, Dec. 2018.

[65] W. Li, O. H. Jafari, and C. Rother, "Deep object co-segmentation," in *Proc. IEEE Conf. Asi. Conf. Comput. Vis. (ACCV)*, Dec. 2018, pp. 638–653.

[66] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," in *Proc. IEEE Winter Conf. App. Comput. Vis. (WACV)*, Mar. 2018, pp. 839–847.

[67] B. Vasu, F. U. Rahman, and A. Savakis, "Aerial-cam: Salient structures and textures in network class activation maps of aerial imagery," in *Proc. IEEE 13th Image, Video, Multidim. Signal Process. (IVMSP) Workshop*, Jun. 2018, pp. 1–5.

Fanman Meng (Member, IEEE) received the Ph.D. degree in signal and information processing from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2014. From July 2013 to July 2014, he joined the Division of Visual and Interactive Computing, Nanyang Technological University, Singapore, as a Research Assistant. He is currently an Associate Professor with the School of Electronic Engineering, University of Electronic Science and Technology of China. He has authored or coauthored numerous technical articles in well-known international journals and conferences. His research interests include image segmentation and object detection.

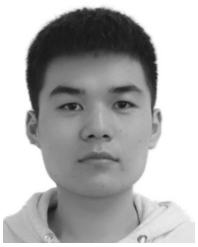
He is now a member of IEEE CAS society. He received the Best Student Paper Honorable Mention Award in the 12th Asian Conference on Computer Vision (ACCV 2014) at Singapore and the Top 10% Paper Award in the IEEE International Conference on Image Processing (ICIP 2014) at Paris, France.

Kunming Luo received the B.Eng. and M.S. degrees from the University of Electronic Science and Technology of China, Chengdu, China, in 2016 and 2019, respectively. His current research interests include computer vision and deep learning.

Hongliang Li (Senior Member, IEEE) received the Ph.D. degree in electronics and information engineering from Xi'an Jiaotong University, China, in 2005. From 2005 to 2006, he joined the Visual Signal Processing and Communication Laboratory (VSPC), The Chinese University of Hong Kong (CUHK), as a Research Associate, where he was a Post-Doctoral Fellow from 2006 to 2008. He is currently a Professor with the School of Electronic Engineering, University of Electronic Science and Technology of China. He has authored or coauthored numerous technical articles in well-known international journals and conferences. He was involved in many professional activities. His research interests include image segmentation, object detection, image and video coding, visual attention, and multimedia communication systems.

He is a member of the Editorial Board of *Journal on Visual Communications and Image Representation*. He served as a Technical Program Co-chair for VCIP2016 and ISPACS 2009, a General Co-Chair for the ISPACS 2010, a Publicity Co-Chair for IEEE VCIP 2013, a Local Chair for the IEEE ICME 2014, and a TPC member in a number of international conferences, such as ICME 2013, ICME 2012, ISCAS 2013, PCM 2007, PCM 2009, and VCIP 2010. He is an Area Editor of *Signal Processing: Image Communication*, Elsevier Science. He is a Co-Editor of a Springer book titled *Video Segmentation and Its Applications*.

Qingbo Wu (Member, IEEE) received the B.E. degree in education of applied electronic technology from Hebei Normal University in 2009 and the Ph.D. degree in signal and information processing from the University of Electronic Science and Technology of China in 2015. From February 2014 to May 2014, he was a Research Assistant with the Image and Video Processing (IVP) Laboratory, Chinese University of Hong Kong. From October 2014 to October 2015, he served as a Visiting Scholar with the Image and Vision Computing (IVC) Laboratory, University of Waterloo. He is currently a Lecturer with the School of Electronic Engineering, University of Electronic Science and Technology of China. His research interests include image/video coding, quality evaluation, and perceptual modeling and processing.



Xiaolong Xu received the B.Eng. degree in electronic and information engineering from the Hefei University of Technology, Hefei, China, in 2018. He is currently pursuing the M.S. degree in signal and information processing with the University of Electronic Science and Technology of China. His current research interests include computer vision and machine learning.