GraphNet: Learning Image Pseudo Annotations for Weakly-Supervised Semantic Segmentation

Mengyang Pu, Yaping Huang*, Qingji Guan, Qi Zou
Beijing Key Laboratory of Traffic Data Analysis and Mining, Beijing Jiaotong University, Beijing, 100044
{mengyangpu,yphuang,qingjiguan,qzou}@bjtu.edu.cn

ABSTRACT

Weakly-supervised semantic image segmentation suffers from lacking accurate pixel-level annotations. In this paper, we propose a novel graph convolutional network-based method, called GraphNet, to learn pixel-wise labels from weak annotations. Firstly, we construct a graph on the superpixels of a training image by combining the low-level spatial relation and high-level semantic content. Meanwhile, scribble or bounding box annotations are embedded into the graph, respectively. Then, GraphNet takes the graph as input and learns to predict high-confidence pseudo image masks by a convolutional network operating directly on graphs. At last, a segmentation network is trained supervised by these pseudo image masks. We comprehensively conduct experiments on the PASCAL VOC 2012 and PASCAL-CONTEXT segmentation benchmarks. Experimental results demonstrate that GraphNet is effective to predict the pixel labels with scribble or bounding box annotations. The proposed framework yields state-of-the-art results in the community.

CCS CONCEPTS

• **Computing methodologies → Image Segmentation**;

KEYWORDS

Weakly Supervised Learning; Semantic Segmentation

ACM Reference Format:

Mengyang Pu, Yaping Huang, Qingji Guan, Qi Zou. 2018. GraphNet: Learning Image Pseudo Annotations for Weakly-Supervised Semantic Segmentation. *In 2018 ACM Multimedia Conference (MM' 18), October 22-26, 2018, Seoul, Republic of Korea.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3240508.3240542

1 INTRODUCTION

Image semantic segmentation aims to assign each pixel a visual category label in one image. Benefiting from the deep convolutional neural networks (DCNNs) and large-scale pixel-level annotated training data, fully supervised image semantic segmentation has been developed and achieved relatively high performance [3, 4, 7, 8, 18, 29, 46, 47]. However, large-scale pixel-level annotation data is still needed to train a deep model [7, 31]. It is expensive and time-consuming to annotate pixel-level labels on large-scale image datasets, which greatly limits the application and development of semantic segmentation in practice. In contrast, weakly-supervised image semantic segmentation learns the masks with weak annotations, and thus has attracted more attention.

Recently, image semantic segmentation in a weakly-supervised manner has been widely developed [13, 20, 22, 24, 28, 31, 34, 35, 38, 40, 42–45]. The critical issue is to predict pixel-wise labels from image-level annotations or partially pixel-level annotations. Lin *et al.* [28] employ another form of the weak labels, *i.e.*scribble. Scribble can be obtained by users interacting with ordinary touch screen devices and machines in a friendly manner and is therefore widely used in practice. Though such scribble-supervised methods deliver more impressive results, compared with the corresponding fully-supervised counterparts, its optimization process is totally cumbersome. Other bounding box-based weakly-supervised method, such as [10], also suffers from optimizing the model with several dozens of iterations.

It is worthy to note that some graph-based image segmentation proceeds appreciable performance by dividing an image into "regions" or "blobs" with only generic cues of coherence or similarity among pixels [5]. Some supervised graph-based methods are also proposed to preserve the necessary structure for accurate segmentation. For example, based on the region boundaries, [15] performs a graph-based image representation to preserve details in low-variability region and ignore details in high-variability regions. Graph-based methods propagate the labeled information and capture the intrinsic relation in both local neighborhood and global image. Due to the excellent performs of DCNNs in feature learning, researchers attempt to construct a similar convolutional neural network on graph-structured data [6, 11, 16, 19, 23]. Although graph convolution networks surpass the traditional methods on some existing relational datasets, such as text citation dataset, it is still a challenging task to directly apply it on the images or video data in the field of computer vision.

In this paper, we consider employing graph convolutional networks to deal with weakly-supervised semantic segmentation. Motivated by the advantages of graph-structured data and efficient graph convolutional operations, we propose a novel graph convolutional network-based method, called GraphNet, to learn image pseudo annotations for weakly-supervised semantic segmentation. More specifically, we construct a graph on an image by considering a dual constraint between the image CNN features and image spatial partition. Such a graph not only combines image low-level local correlated cues and the high-level semantic contents but also characterizes a naturally structured representation of the original image. Then we establish a graphical neural network model, which

* Corresponding author.

MM '18, October 22–26, 2018, Seoul, Republic of Korea
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-5665-7/18/10…$15.00
https://doi.org/10.1145/3240508.3240542

allows us to perform the necessary operations of convolutional networks, such as convolution, pooling or non-linear transformation, directly on the graph. Consequently, in order to estimate a reliable segmentation mask for each training image, we propagate the category information from labeled pixels to unlabeled pixels by our proposed GraphNet.

Furthermore, GraphNet is apt to generalize to different kinds of annotated data. Accordingly, we integrate the GraphNet into a weakly-supervised semantic segmentation framework. We perform experiments on both scribble and bounding box annotations. Scribble can provide the category information of partial pixels directly, but bounding box annotations need to be processed so as to generate high-confidence labels within the box. The proposed pseudo annotations methods improve the performance of image semantic segmentation supervised by both scribble and bounding box annotations. Specifically, with one initial round of training, we can achieve 68.2% and 65.0% for the scribble and bounding box annotations, respectively. Moreover, our framework yields 68.9%(scribble) and 65.6%(bounding box) with an additional round of optimization.

Our main contribution is summarized as follows:

- We address the challenges of weakly-supervised semantic segmentation by proposing a graph-based GraphNet to generate accurate pseudo annotations.
- GraphNet applies convolutional neural networks to the graph-structured data, which allows the graphical model can be efficiently optimized for label propagation. Therefore, our method can achieve comparable segmentation results with a single round training, and its performance can be further improved by additional rounds.
- We conduct comprehensive experiments on the PASCAL VOC 2012 and PASCAL-CONTEXT dataset with scribble or bounding box annotations. Our proposed framework achieves superior performance compared with the state-of-the-art methods.

The rest of the paper is organized as follows. We briefly review the related work in Section 2. Section 3 describes the details of GraphNet and the pre-processing of different annotations. The details of segmentation and optimization strategies are introduced in Section 4. The configuration of the proposed method and experimental results are described in Section 5. We conclude our work in Section 6.

2 RELATED WORK

Weakly-Supervised semantic segmentation. The recently proposed weakly-supervised methods mainly use weak labels like image-level labels [24, 31–33], scribble [28, 38], and bounding box [10, 31]. For the given image-level or box-level annotations, most existing approaches commonly use models such as Multiple Instance Learning (MIL) [32, 33], Expectation Maximization (EM) [31], or decoupled network [24], which can locate the most distinctive object parts. However, it is still hard to capture the whole object regions.

Image-level annotations are the easiest to obtain, but accuracy is still far behind supervised segmentation. Instead, the results of bounding box annotations are closer to supervised ones. Dai *et al.* [10] propose segmentation with BoxSup which performs iterative optimization between generating segmentation masks and training network. Though the performance is more advanced than other methods, it also suffers from optimizing the model with dozens of iterations. Additionally, pixel-level annotations are also extra needed in the step of Multiscale Combinatorial Grouping (MCG) [2] in BoxSup. Papandreou *et al.* [31] solve the optimization problem by adopting an automatic foreground/background segmentation strategy. Specifically, a fully connected CRF is leveraged to filter out the background pixels. However, the CRF parameters are learned from a small held-out set of fully-annotated images.

Scribble is proposed for image semantic segmentation as another form of weak supervision[28]. Lin *et al.* [28] solve the problem that assigns a category label for each superpixel by multi-label graph cuts. The label information is effectively propagated to all unlabeled pixels. Based on the estimated labels, the gap between trained segmentation model and fully supervised methods is reduced. However, the energy function of the graphical model is optimized with an alternating algorithm, and multiple rounds of optimization are also necessary to obtain good performance.

Graph Convolutional Network in Computer Vision. As a powerful data structure, graphs can express not only the intrinsic entities with nodes but also the complicated relationships between entities with edges, which is commonly adopted in social networks, knowledge graphs, protein-interaction networks *etc.*. Inspired by the advanced development of convolutional neural networks in the image (video)-based tasks, a number of researchers rise to study the problem of employing neural networks to arbitrarily structured graph [6, 11, 19, 21, 23, 25]. It is a challenging task to apply well-established neural models to structured graphs in the field of computer vision. Jain *et al.* [21] construct a graph by combining the temporal and spatial relationships between entities (objects, human *etc.*) and extend Recurrent Neural Networks to graphs for detecting human activity. In addition, many fully-supervised semantic segmentation problems are solved by utilizing neural network optimization graphical models. Liang *et al.* [27] propose a deep Local-Global Long Short-Term Memory (LG-LSTM) architecture, which is applied to a grid structure and incorporate short-distance and long-distance spatial dependencies into the feature learning over all pixel positions. In [26], a Graph Long Short-Term Memory (Graph LSTM) network is proposed, which is the generalization of LSTM from sequential or multidimensional data to general graph-structured data. Although the adaptive graph structure improves the consequence of the semantic segmentation models, training these models still requires pixel-level annotation data.

3 THE PROPOSED METHOD

This section describes how to generate pseudo labels by GraphNet from weak labels such as scribble and bounding box annotations. The overall framework is shown in Figure 1. Firstly, we convert an image with a regular grid structure into a structured graph with initial outline cues. Subsequently, we elaborate on the significant work in this paper that includes the theoretical definition and network architecture of GraphNet. Furthermore, we also illustrate the detailed process of generating the initially labeled nodes from scribble or bounding box annotations.

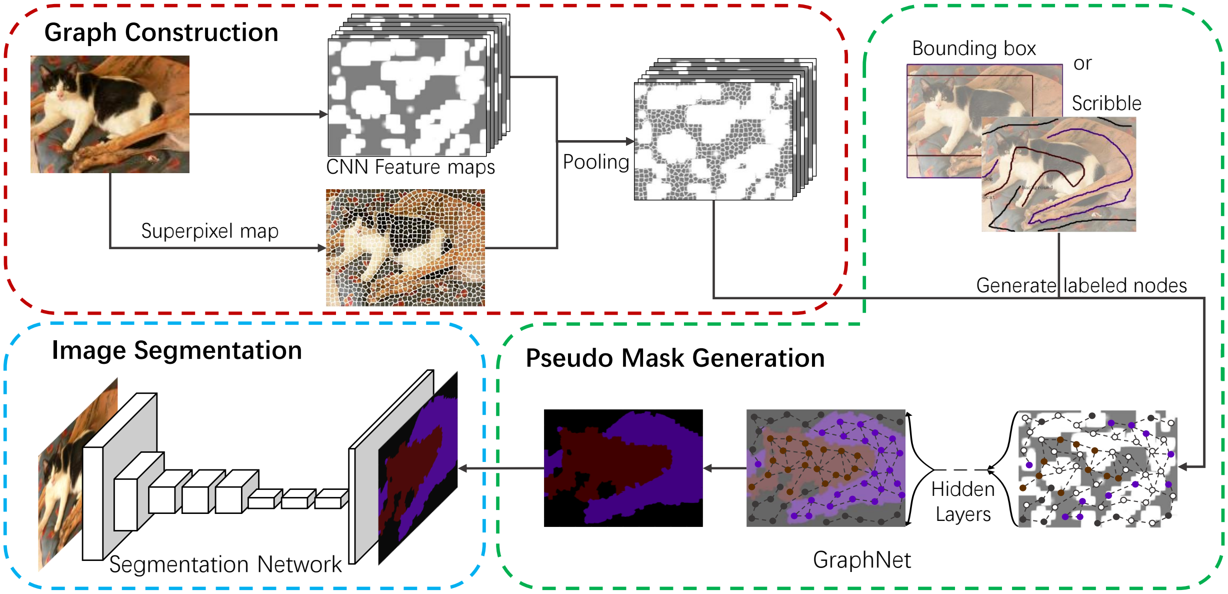


Figure 1: Overview of the proposed framework for weakly-supervised semantic segmentation. First, the CNN features are extracted from a VGG-16[36] pre-trained on ImageNet[12], and graph is constructed by combining feature similarity and the spatial location given by superpixel map and the scribble or bounding box annotations. Then, pseudo image masks are learned by the proposed GraphNet. At last, a segmentation network is trained supervised by the generated pseudo annotations. The process of generating labeled nodes by scribble and bounding box annotations is described in Figure 2 and 3, respectively.

3.1 Graph Construction

Learning pseudo labels can be regarded as a label propagation problem. Traditional DCNNs operate on images with regular grid structure. However, it is hard to work for label propagation problem. We consider transforming an image to a graph-structured data for label propagation. The graph representation is constructed by considering low-level cues (*e.g.*outline, shape) of individual images and the dual constraints of spatial location and semantic content.

Superpixels with object outline cues are extremely suitable as nodes of the graph structure which represents one image. As discussed in [39], superpixel provides a larger, locally homogeneous and coherent region that preserve most of the structure necessary for accurate segmentation. Therefore, we utilize the Simple Linear Iterative Clustering (SLIC) algorithm [1] to over-segment each image I and divide it into a superpixel set, denoted as $\{sp_i\}_{i=1}^{N}$, which contains N superpixels. Following, we describe how to extract features on each superpixel and construct a graph by dual constraints of spatial location and semantic content.

Feature Extraction on superpixels. Above all, we extract the features from the whole image. Earlier layers in convolutional neural networks are prone to learn low-level features (*e.g.*edges), while later layers capture more semantic information (*e.g.*class labels). We require assigning a semantic label to each pixel, so semantic features are extracted from a deep, coarse layer. Specifically, we employ the VGG-16 [36] model which is pre-trained on the ImageNet Visual Recognition Challenge [12] to extract the high-level semantic representations.

The VGG-16 model is designed for image classification tasks and limits the size of the input image. In our work, we require extracting feature maps from the ReLU-5 layer, instead of the final result of the classification. Therefore, we input the original image without cropping into the pre-trained VGG-16 model to extract high-resolution feature maps. Note that, the feature maps of each image are downsampled by the operation of several convolutional layers. In order to obtain dense feature maps with high-resolution, we resize the feature activations to the same size with original images by a bilinear interpolation. Finally, an average pooling is performed on a superpixel along the channels, and a 512-dimensional CNN feature vector x_i for each superpixel sp_i is obtained.

The dual constraints. Two spatially adjacent nodes with similar characteristics commonly tend to belong to the same category. However, only relying on the spatial constraints may ignore the contextual semantic interaction. Here, we consider the dual constraints of spatial information and semantic content to construct a graph for each image.

First, we consider the spatial adjacency constraint. The superpixel-based graph focuses on constructing a structured representation of image, defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$. Each node v_i in $\mathcal{V}$ corresponds to a superpixel and the edge ε_{ij} in $\mathcal{E}$ only connects two spatially neighboring superpixel nodes. A is the graph adjacency matrix.

We denote the *initial weight matrix* as $W_l = [w_l^{ij}]_{n \times n} \in \mathbb{R}^{N \times N}$ which measures the spatial relationship among all superpixel nodes. If two nodes v_i and v_j are spatially adjacent, then the weight w_l^{ij} between them is defined as:

$$w_l^{ij} = \begin{cases} 1, & \text{if } v_j \in \mathcal{N}_{\mathcal{G}}(i) \\ 0, & \text{otherwise} \end{cases}, \tag{1}$$

where $\mathcal{N}_{\mathcal{G}}(i)$ represents a neighboring nodes set of node v_i.

Then, we use a *semantic weight matrix* $W_s = [w_s^{ij}]$ to measure the semantic similarity between all spatially neighboring nodes.

Given the superpixel features $\{x_i\}_{i=1}^{N}$, the value w_s^{ij} is calculated as:

$$w_s^{ij} = \begin{cases} \exp(-\frac{\|x_i - x_j\|}{2h}), & \text{if } w_l^{ij} = 1 \\ 0, & \text{otherwise} \end{cases}, \tag{2}$$

where h is the dimension of the feature vector.

Subsequently, according to the *semantic weight matrix* W_s, we remove the edges with low similarity from the edge set $\mathcal{E}$ to calculate the final adjacency matrix A. We observe that the above operation may produce isolated nodes in a graph. Therefore, we utilize the following strategies to remove edges with low semantic similarity while ensuring connectivity of the graph. At first, we set a threshold $\gamma = u(W_s) - \sigma(W_s)$ to filter out the edges with low semantic similarity, where $u(\cdot)$ is the mean value, $\sigma(\cdot)$ is the standard deviation, and W_s is the *semantic weight matrix*. For a node v_i, we calculate its maximum similarity value $\tilde{w}_{s,max}^{i}$ between node v_i and its adjacent nodes, denoted as $\tilde{w}_{s,max}^{i} = \max(w_s^{ij}, v_j \in \mathcal{N}_{\mathcal{G}}(i))$. Then we specify that if the semantic similarity w_s^{ij} between v_i and $v_j \in \mathcal{N}_{\mathcal{G}}(i)$ is below the threshold γ, at the same time, w_s^{ij} is lower than $\tilde{w}_{s,max}^{i}$, then the edge ε_{ij} will be removed. The element a_{ij} in adjacency matrix A is calculated as follows:

$$a_{ij} = \begin{cases} 0, & \text{if } w_s^{ij} < \gamma \text{ and } w_s^{ij} < \tilde{w}_{s,max}^{i} \\ 1, & \text{otherwise} \end{cases}. \tag{3}$$

3.2 GraphNet

GraphNet propagates the labels of a small number of nodes to the unlabeled nodes in a graph. It is directly achieved by graph convolutional networks. For the dense annotation generation of an image, the image represented by a graph is particularly effective for modeling both local appearance and global spatial interaction. Moreover, powerful features can be further obtained from convolutional operations. The GraphNet is based on spectral graph convolutional neural networks, introduced in [6, 23]. To perform the convolution directly on the constructed graph, we introduce an essential operator for spectral graph analysis [9], graph Laplacian, whose combinatorial definition is given by

$$L = D - A \in \mathbb{R}^{N \times N}, \tag{4}$$

where $D \in \mathbb{R}^{N \times N}$ is the diagonal degree matrix with $D_{ii} = \sum_j a_{ij}$, and the normalized definition is

$$L = I_N + D^{-\frac{1}{2}} A D^{-\frac{1}{2}} = U \Lambda U^T, \tag{5}$$

where I_N is the identity matrix, and U is the matrix of eigenvectors of the normalized graph Laplacian L with a diagonal matrix of its eigenvalues Λ.

GraphNet Architecture. Let P and Q be the number of labeled and unlabeled nodes, where $P + Q = N$. The corresponding feature vector is denoted as $X_{1:P}$ and $X_{P+1:P+Q}$, respectively. Given the labels $Y_{1:P}$, we train a neural network model $f(X, A)$ with a supervised loss for all labeled nodes. Training the parameters of $f(\cdot)$ will allow the model to distribute gradient information from the supervised loss. Meanwhile, It will also enable nodes without labels to learn representations. According to the propagation rules of classical neural networks, each layer can be written as a nonlinear function. We adopt the following simple form of a layer-wise propagation rule [23]:

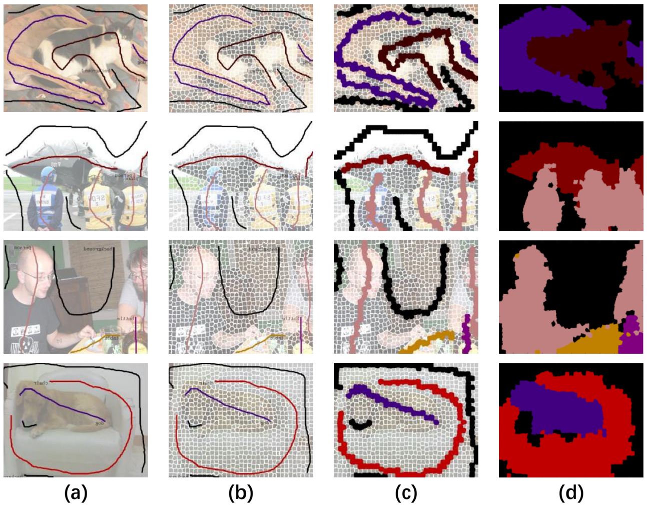


Figure 2: Labeled nodes from scribble annotations. (a) Scribble annotations. (b) The superpixel map with scribble. (c) Labeled nodes. (d) Pseudo annotations.

$$H^{(l+1)} = \sigma(A H^{(l)} W^{(l)}), \tag{6}$$

where A is the graph adjacency matrix, $W^{(l)}$ is a parameter matrix for the l-th neural network layer and $\sigma(\cdot)$ is a non-linear activation function like ReLU.

In addition, to alleviate the problem of numerical instabilities and exploding or vanishing gradients in the deep neural network model, we introduce the following *renormalization trick*: $I_N + D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \rightarrow \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}$, with $\tilde{A} = A + I_N$ and $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$. What is more, layer-wise propagation rules can be defined as:

$$H^{(l+1)} = \mathrm{ReLU}(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)}). \tag{7}$$

In our scenario, a two-layer graphical convolution model $f(X, A)$ is introduced for the label propagation. We define the rule of model forwarding:

$$Z = f(X, A) = \mathrm{softmax}\,(\hat{A}\ \mathrm{ReLU}\,(\hat{A} X W^{(0)}) W^{(1)}), \tag{8}$$

where $\hat{A} = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}$, and X is a matrix of node feature vector x_i. $W^{(0)}$ and $W^{(1)}$ are the model parameters from input layer to hidden layer and hidden layer to output layer, respectively.

Loss Function. We optimize the GraphNet by minimizing the cross-entropy loss over all labeled nodes:

$$\mathcal{L} = -\sum_{i \in \mathcal{Y}_P} \sum_{c=1}^{C} y_i^c \ln z_i^c, \tag{9}$$

where C is the number of category label, and z_i^c means that the prediction result of the i-th node belongs to category c.

The above loss $\mathcal{L}$ is continuously reduced by the gradient descent. The weight parameters of the neural network, W_0 and W_1, are optimized, thereby assigning a category label to unlabeled nodes and generating a dense predicted mask for each training image.

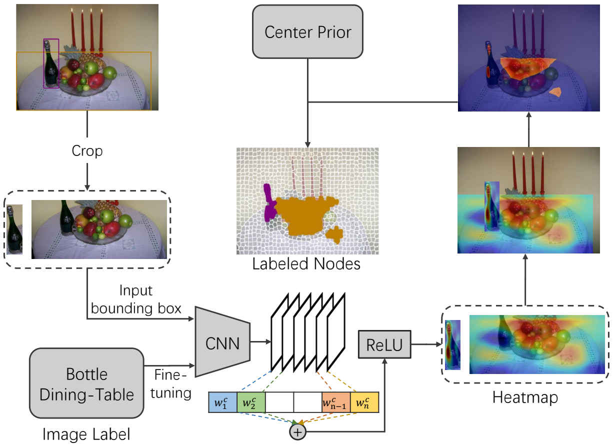


Figure 3: Overview of generating labeled nodes from bounding box annotations.

3.3 Scribble Annotations

As shown in Figure 2 (a), scribble annotations provide a set of sparse pixels with category labels. Firstly, we take the superpixels covered by scribbles as the labeled data. Specifically, following the work of Lin *et al.* [28], we denote the scribble annotations of an image as $S = \{s_j, c_j\}$, where s_j is the pixel set of the j-th scribble and c_j is the category label of this scribble. If a superpixel sp_i overlaps with a scribble s_j, then we assign the category label c_j to this superpixel sp_i, as:

$$y_i = \begin{cases} c_j, & \text{if } sp_i \bigcap s_j \neq \emptyset \\ ø, & \text{otherwise} \end{cases} . \tag{10}$$

We show some examples of the labeled nodes with scribble in Figure 2 (c). GraphNet assigns a category label to each superpixel after training on a supervised loss $\mathcal{L}$ for all nodes with labels. Some estimated masks are shown in the figure 2 (d).

3.4 Bounding Box Annotations

Obviously, scribbles can provide accurate labels for specific pixels in the image so that we can obtain nodes with labels directly. However, the bounding box provides only the location of the object in the image. Therefore, it is a crucial step to accurately locate the pixels on the object within bounding boxes to generate labeled nodes. As discussed in [41], classification-based networks, such as CAM model[48], can only produce small and sparse object regions. Here, we can utilize the propagation performance of GraphNet to further locate dense and complete object regions.

The process of generating initially labeled nodes from bounding box annotations is described in Figure 3. The bounding box annotations provide category labels for single or multiple objects per image, so we first fine-tune the original CAM model [48] on the dataset. Then, the cropped bounding box image patch is input into the fine-tuned model in turn, and the corresponding category heatmap is extracted. We set a threshold β to select the highlight regions in the heatmap. These selected regions are discriminative for image classification and they can accurately locate the local regions of the object in the bounding box. For images with multiple objects, we require paying special attention to the overlap between boxes. This problem can cause overlapping of the selected discriminating regions, then the label of the pixel on the overlap regions would be confused. We solve this problem by assigning the label of ambiguities pixels (that belong to multiple bounding boxes) to the one with the highest prediction values in the heatmap. Subsequently, we merge the extracted regions with the range box of center prior, then we assign labels to the superpixel nodes covered by the merged regions. Note that the range box of center prior is controlled by a threshold α. That is, the height and width of the range box is $\alpha\%$ of the bounding box. Ultimately, we employ those labeled nodes as training data and utilize GraphNet (Section 3.2) to predict a final dense pseudo annotation for each image.

4 SEGMENTATION AND OPTIMIZATION

4.1 Segmentation with Weak Annotations

We train a DeepLabv2-VGG16 [7] based segmentation network with the pseudo masks generated by the proposed GraphNet. VGG-16 is utilized as our backbone. In our framework, there is a weak coupling between generating pseudo labels and training segmentation models. Therefore, we can replace the segmentation model with any state-of-the-art models.

4.2 Further Optimization

With the pseudo image masks generated by GraphNet, the segmentation model with initial round can achieve satisfying performance. In order to further improve the segmentation performance of the model, we provide two following optimization strategies.

In the beginning, we replace the VGG-16 model with the initial segmentation model to extract the features. Subsequently, we utilize the output results of GraphNet and segmentation model to extend the new labeled nodes. Obviously, the initial segmentation model can output a dense label for each training image and predict confidence maps with regard to each semantic label. GraphNet, on the other hand, assigns a category label and a corresponding prediction score for each superpixel. We select the regions with high-confidence value and same category labels from the predictions of two networks. Some initially labeled nodes generated by Section 3.3 or Section 3.4 may already exist in the selected regions. Departing from these nodes, we regard the others as the extended labeled nodes. Let P, EP and Q' be the number of initially labeled nodes and extended labeled nodes and unlabeled nodes, where $P + EP + Q' = N$. Let $X_{1:P}$, $X_{P+1:P+EP}$, and $X_{P+EP+1:N}$ denote the feature vectors of the above three types of nodes, respectively. Certainly, the labels $Y_{1:P}$ and $Y_{P+1:P+EP}$ are known.

In the first strategy, *Joint Training*, initial and expanded nodes are collectively referred to as labeled nodes. We optimize the joint loss with labels $Y_{1:P}$ and $Y_{P+1:P+EP}$ as described with Equation 9. The labels of the extended nodes are decided by both the initial segmentation model and GraphNet, however, which could lead to ambiguous label assignment. Therefore, in the *Joint Training*, a small number of nodes propagate the false labels to nearby nodes with higher probability. To reduce the influence of the nodes with the false labels on the overall label propagation, we propose another optimization strategy: *Alternate Training*. Training GraphNet is divided into two stages: 1) optimizing the supervised loss on initial nodes with labels $Y_{1:P}$, and 2) fine-tune the parameters of GraphNet on initial and extended nodes.

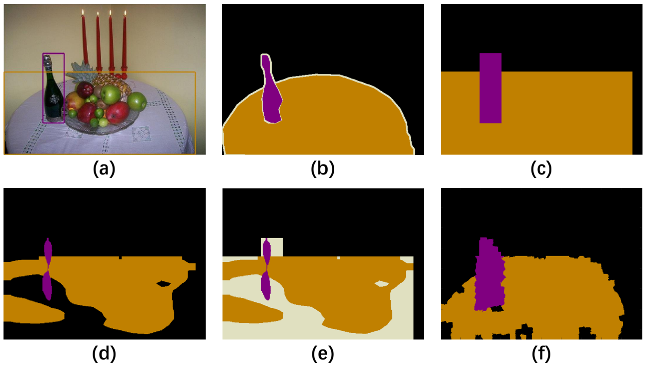


Figure 4: The estimated mask from box annotation. (a) Image with bounding box. (b) Ground-Truth. (c) Boundingbox Rectangles. (d) The estimated mask of *BboxCAM-S1*. (e) The estimated mask of *BboxCAM-S2*. (f) The estimated mask of *BboxGraphNet*.

5 EXPERIMENT

Dataset. We evaluate our method on the PASCAL VOC 2012 [14] and PASCAL-CONTEXT [30] dataset. **PASCAL VOC 2012** involves 20 object categories and one background category. The original dataset for segmentation contains 1,464 training images, 1,449 validation images, and 1,456 test images. The dataset is augmented by the extra annotations provided by [17], resulting in 10,582 (*train-aug*) training images. Each image pixel is elaborately labeled as one of the 21 categories. However, instead of using precise pixel-level annotations, we use scribble annotations provided by [28], and bounding box annotations for object detection tasks [14]. **PASCAL-CONTEXT** involves 59 categories of objects and stuff. The dataset has 4,998 training images and 5,105 images for validation and is annotated with scribble annotations.

Implementation Details. We adopt the DeepLabv2-VGG16[7] to evaluate the accuracy of the generated pseudo masks. We train the model on the 10,582 training images with pixel-level annotations and take it as our strongly-supervised baseline. In addition, the mean Intersection-over-Union (mIoU) is evaluated. For all networks, we report results before CRF (w/o CRF) and after CRF (w/ CRF). The parameters in CRF is the same as the DeepLabv2-VGG16 code. The strongly-supervised baseline result we implemented is 68.8% (w/o CRF) and 71.5% (w/ CRF) respectively. The network architecture of DeepLabv2-VGG16 serves as our network architecture of scribble and bounding box annotations experiments.

5.1 Scribble Annotations

We perform experiments with scribble annotations on the PASCAL VOC 2012 dataset and PASCAL-CONTEXT dataset. The GraphNet is evaluated by training the DeepLabv2-VGG16 with the estimated pseudo masks from scribble annotations.

5.1.1 Experiments on PASCAL VOC 2012. Our method is marked as *ScrGraphNet*. We report the quantitative results in Table 1. More specifically, in Table 1, we present the mIoU of *ScrGraphNet* with the initialized masks generated by GraphNet (Initial) and masks which are further optimized with an additional round training (1-Round). ScribbleSup [28] and RAWKS [38] have mIoU of 63.1% and 61.4%, with 5.6% and 7.3% lower than its corresponding strongly-supervised results, respectively. In the most recent results, NormalizedCutLoss [37] achieves state-of-the-art performance. With DeepLabv2-VGG16, our proposed ScrGraphNet obtains mIoU of 68.2% with only initial training which is higher 3% than NormalizedCutLoss [37]. Through another 1-Round training, ScrGraphNet leads to 68.9% and reduces the gap to 2.6% compared with the strong supervised result (71.5%).

Table 1: PASCAL VOC 2012 *val* performance for scribble annotations.

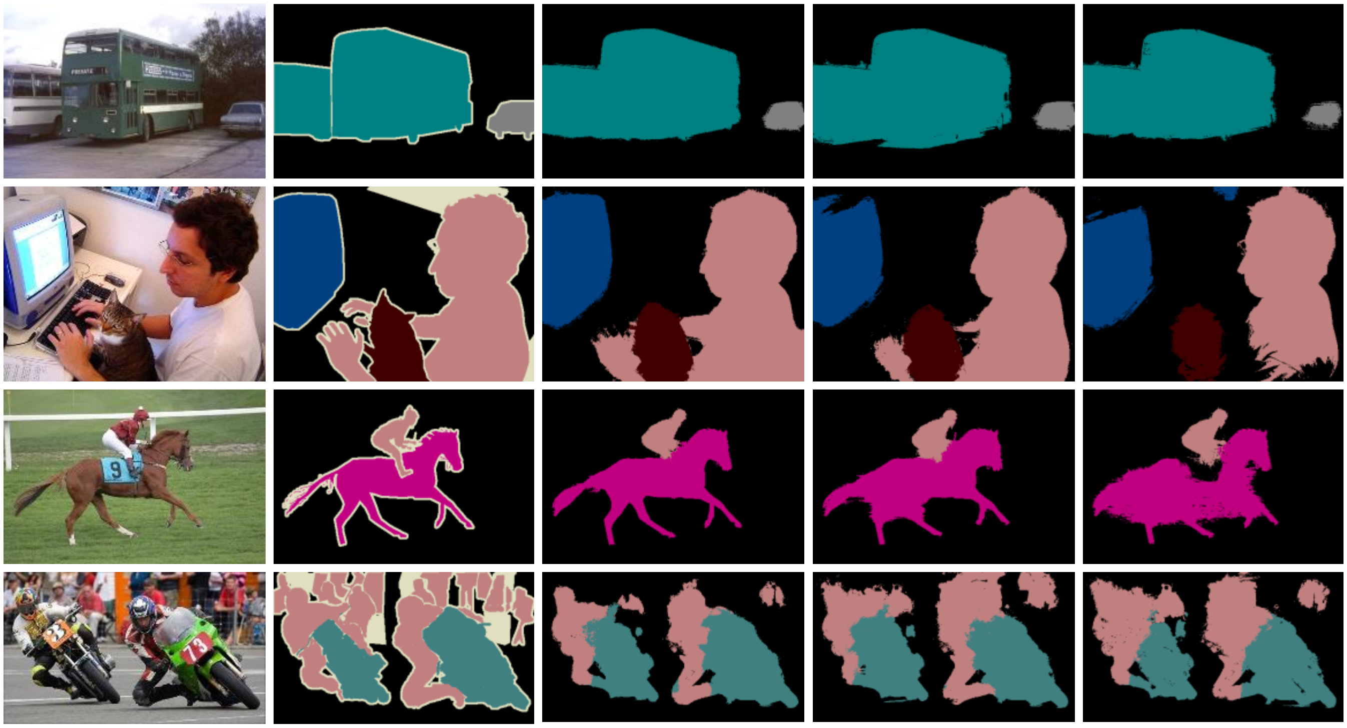
method	strong	w/o CRF	w/ CRF
ScribbleSup [28]	(1)	-	63.1
RAWKS [38]	(1)	-	61.4
NormalizedCutLoss [37]	(1)	60.5	65.1
NormalizedCutLoss [37]	(2)	62.4	65.2
Ours: ScrGraphNet, Initial			
w/o semantic	(2)	62.8	68.0
w/ semantic	(2)	**63.3**	**68.2**
w/ semantic (max pooling)	(2)	62.7	67.9
Ours: ScrGraphNet, 1-Round			
Replaced Features	(2)	63.7	68.3
Joint Training	(2)	64.1	68.7
Alternate Training	(2)	**64.5**	**68.9**
strong			
(1) DeepLab-MSc-largeFOV		64.1	68.7
(2) DeepLabv2-VGG16		68.8	71.5

In summary, GraphNet can sufficiently employ the representation of the graph structure, and it can achieve more effective optimization results by simple propagation rules.

Comparisons of Graph Structure Construction. We evaluate the impact of high-level semantic cues in graph construction for segmentation results. To prove the effectiveness of semantic information, we establish the following settings: ***w/ semantic*** refers to combining semantic similarity and spatial relation constraints in graph construction, while the ***w/o semantic*** represents that graph is only constrained by spatial relationship. As shown in Table 1, with high-level semantic cues, the mIoU (w/o CRF) improves from 62.8% to 63.3%. The comparison indicates that dual constraints of spatial location and semantic content can lead to a better representation of the image. However, mIoU (w/ CRF) only increases from 68.0% to 68.2% after the segmentation results which are refined by CRF. The reason may be that CRF considers semantic interaction on ***w/o semantic*** and leads to similar segmentation results compared with ***w/ semantic***.

Comparisons of Superpixel Pooling Strategy. We investigate the influence of pooling strategy when extracting features on superpixels. Max pooling is used instead of average pooling, denoted as ***w/ semantic (max pooling)*** and the other settings are as same as the ***w/ semantic***. *ScrGraphNet* under condition of ***w/ semantic (max pooling)*** has mIoU of 62.7% (w/o CRF) and 67.9% (w/ CRF), which is inferior to it with average pooling.

Comparisons of Optimization Strategy for GraphNet. As described in section 4.2, we propose two optimization strategies, which are divided into the following two steps: 1) replacing the classification model with the initial segmentation model to extract



(a) image (b) ground-truth (c) strong (d) ScrGraphNet (e) BboxGraphNet

Figure 5: Our results on the PASCAL VOC 2012 *val* set.

Table 2: Training different networks with ScrGraphNet on PASCAL VOC 2012 *val*.

	strong		weak	
method	w/o CRF	w/ CRF	w/o CRF	w/ CRF
DeepLab-largeFOV	62.3	67.6	59.5	66.3
DeepLabv2-VGG16	68.8	71.5	64.5	68.9
DeepLabv2-ResNet101	75.6	76.5	70.3	73.0

features, and 2) training the model with extended labels nodes by *Joint Training* and *Alternate Training* strategies, respectively. In order to verify the impact of features on the optimization strategy, we set a comparative experiment, *Replaced Features*. Specifically, *Replaced Features* means that we only replace the VGG-16 model with the initial segmentation model and then learn new pseudo labels without extended labels nodes. Through the CRF post-processing, mIoU of *Joint Training* reaches to 68.7%. Especially, *Alternate Training* obtains the highest mIoU of 68.9%.

The experimental results with different settings show that GraphNet can obtain the relatively accurate masks and stably boost weakly-supervised semantic segmentation performance. Moreover, we provide the qualitative results of *SrcGraphNet* with an additional round of alternate optimizations, as shown in the Figure 5(d).

Comparisons of Different Segmentation Networks. We test the generated pseudo annotations on different segmentation networks on the PASCAL VOC 2012 dataset. The results are summarized in Table 2. The generated pseudo annotations reduce the performance gap between the strong-supervised and weak-supervised methods. It also illustrates the effectiveness of ScrGraphNet.

Table 3: The result of ScrGraphNet on CONTEXT *val*.

method	strong	w/o CRF	w/ CRF
ScribbleSup [28]	(1)	-	36.1
RAWKS [38]	(1)	-	37.4
Ours: ScrGraphNet			
Initial	(2)	33.1	39.7
1-Round, *Joint Training*	(2)	34.2	40.1
1-Round, *Alternate Training*	(2)	33.9	**40.2**
strong			
(1) DeepLab-MSc-LargeFOV		-	37.7
(2) DeepLabv2-VGG16		36.0	41.7

5.1.2 Experiments on PASCAL-CONTEXT. We further evaluate the ScrGraphNet on the PASCAL-CONTEXT dataset. The quantitative and qualitative results are shown in Table 3 and Figure 6, respectively. The mIOU of strongly-supervised is 36.0%(w/o CRF) and 41.7%(w/ CRF). ScrGraphNet achieves the mIoU of 40.2% with *Alternate Training*. The results show that our method also preserves excellent performance on the dataset which contains more categories.

5.2 Bounding Box Annotations

In this experiment, we evaluate the proposed GraphNet by training the DeepLabv2-VGG16 and DeepLab-LargeFOV models with bounding box annotations. DeepLab-LargeFOV and DeepLab are served as network architecture in WSSL [31] and BoxSup [10], respectively. We report results of corresponding strongly-supervised

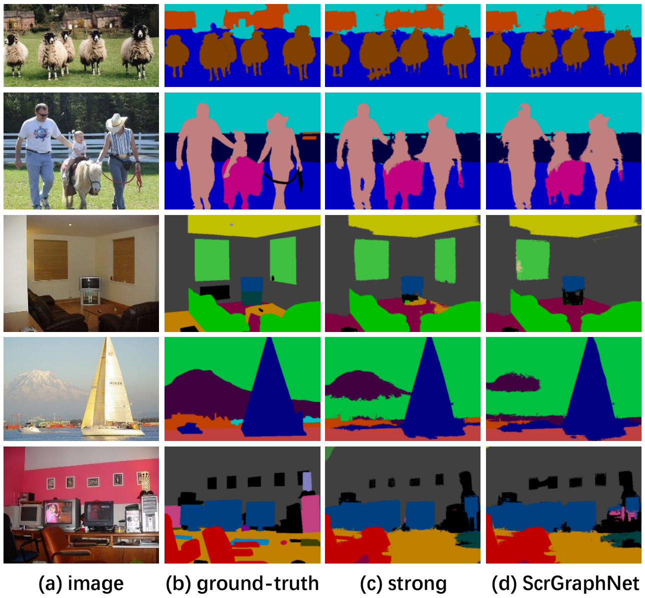


Figure 6: Our results on the PASCAL-CONTEXT *val* set.

results in Table 4. In our *BoxGraphNet*, the response value of each pixel in heatmap is normalized to $[0, 1]$. To obtain regions with high-confidence, we set the threshold $\beta = 0.75$. Also, we define $\alpha = 10\%$. The comparative results are summarized in Table 4.

With DeepLab-LargeFOV network, our method improves over WSSL [31] by 2.8%, and only 4.2% worse than the strongly-supervised results. Additionally, with the DeepLabv2-VGG16 network, our implementation with initial round (Ours: Initial) has mIoU of 65.0%. The performance can be further improved to 65.6% with an additional round optimization (Ours:1-round), that is only 5.9% less than the strongly-supervised result. Especially BoxSup [10] yields only 1.8% worse than the strong pixel-level supervision result. However, BoxSup [10] uses MCG algorithm [2], which requires training with pixels annotations.

Comparative Experiments. To prove the benefits of refining object regions in bounding boxes, we attach three comparative experiments. In the first *BboxRectangle*, we consider each bounding box as an object region of the corresponding category, as shown in Figure 4 (c). Then we train the segmentation model with these masks. The score is 53.2%. However, in the remaining two comparative experiments, we set the threshold $\beta = 0.4$ to directly select the regions with high response value from heatmap as the object regions. For the rest pixels in the box, we perform an interesting setting. *BboxCAM-S1* means that the rest pixels are regarded as the background. On the contrary, we set the label of rest pixels to 255 in the *BboxCAM-S2*, which means that these regions are ignored when training the segmentation model. Examples of estimated masks with *BboxCAM-S1* and *BboxCAM-S2* method as shown in Figure 4 (d) and (e), respectively.

As shown in Table 4, the results show that the *Box-GraphNet* method are the most superior ones in our proposed methods, which demonstrate the effectiveness of the refinement of the objects in the bounding boxes. In addition, we find an attractive result: without CRF, *BboxCAM-S2* improves over *BboxCAM-S1* by 9.7%, and especially *BboxCAM-S2* can yields mIoU of 57.2%. The analysis results show that compared with the *BboxRectangles* and *BboxCAM-S1* methods, the labeled regions in *Bbox-CAM-S2* become much smaller, while the labeling accuracy of these regions is higher, so the segmentation result of *Bbox-CAM-S2* is more accurate.

Table 4: PASCAL VOC 2012 *val* performance for bounding box annotations.

method	strong	w/o CRF	w/ CRF
WSSL [31]	(1)	-	60.6
BoxSup* [10]	(2)	-	62.0
Ours: BboxRectangles	(3)	49.6	53.2
Ours: BboxCAM-S1	(3)	47.5	48.6
Ours: BboxCAM-S2	(3)	57.2	60.7
Ours: BboxGraphNet			
Initial	(1)	56.3	62.7
1-Round, *Joint Training*	(1)	56.9	63.2
1-Round, *Alternate Training*	(1)	**57.1**	**63.4**
Initial	(3)	60.5	65.0
1-Round, *Joint Training*	(3)	60.8	65.2
1-Round, *Alternate Training*	(3)	**61.3**	**65.6**
strong			
(1) DeepLab-LargeFOV		-	67.6
(2) DeepLab		-	63.8
(3) DeepLabv2-VGG16		68.8	71.5

* BoxSup[10] uses MCG[2], which requires training from pixel-level annotations.

We also verify the effect of optimization strategies for GraphNet with bounding box annotations. As shown in Table 4, *Alternate Training* is slightly more effective than *Joint Training* (65.6% vs. 65.2%), which is consistent with the results of *ScrGraphNet*. Furthermore, in Figure 5, we qualitatively compare the visual results of proposed training methods. Note that, Figure 5(e) refers to the qualitative results of *BboxGraphNet* with an additional round of alternate optimizations.

6 CONCLUSIONS

In this paper, we propose GraphNet, a graph convolutional neural network-based method, for learning pseudo labels from weak annotations. We explore GraphNet for different kinds of weak annotations, such as scribble and bounding box annotations. The experimental results show that GraphNet can achieve superior performance compared to other weakly-supervised methods. Although in this paper we only implement GraphNet to generate pseudo labels from scribble and bounding box annotations, GraphNet can also be applied to image-level annotations with appropriate transformations on image data. Additionally, we believe that GraphNet can be improved in network architecture or graph construction. We intend to investigate these issues in future works.

ACKNOWLEDGMENTS

This work is supported by National Natural Science Foundation of China (No.61273364, No.61473031, No.61472029), the Fundamental Research Funds for the Central Universities (2018YJS045, 2016JBZ005, 2016YJS041).

REFERENCES

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. 2012. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence* 34, 11 (2012), 2274–2282.

[2] Pablo Arbeláez, Jordi Pont-Tuset, Jonathan T Barron, Ferran Marques, and Jitendra Malik. 2014. Multiscale combinatorial grouping. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 328–335.

[3] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence* 39, 12 (2017), 2481–2495.

[4] Aayush Bansal, Xinlei Chen, Bryan Russell, Abhinav Gupta Ramanan, et al. 2017. Pixelnet: Representation of the pixels, by the pixels, and for the pixels. *arXiv preprint arXiv:1702.06506* (2017).

[5] Yuri Boykov and Gareth Funka-Lea. 2006. Graph cuts and efficient ND image segmentation. *International journal of computer vision* 70, 2 (2006), 109–131.

[6] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. 2013. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203* (2013).

[7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. 2018. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence* 40, 4 (2018), 834–848.

[8] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. 2018. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *arXiv preprint arXiv:1802.02611* (2018).

[9] F. R. K. Chung. 1997. Spectral Graph Theory CBMS Series. *American Mathematical Society* 9, 6 (1997), 55.

[10] Jifeng Dai, Kaiming He, and Jian Sun. 2015. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision.* 1635–1643.

[11] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems.* 3844–3852.

[12] Jia Deng, Wei Dong, Richard Socher, Li Jia Li, Kai Li, and Fei Fei Li. 2009. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 248–255.

[13] T. Durand, T. Mordan, N. Thome, and M. Cord. 2017. WILDCAT: Weakly Supervised Learning of Deep ConvNets for Image Classification, Pointwise Localization and Segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 5957–5966.

[14] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. 2015. The pascal visual object classes challenge: A retrospective. *International journal of computer vision* 111, 1 (2015), 98–136.

[15] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. 2004. Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision* 59, 2 (2004), 167–181.

[16] David K Hammond, Pierre Vandergheynst, and Rémi Gribonval. 2011. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis* 30, 2 (2011), 129–150.

[17] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. 2011. Semantic contours from inverse detectors. In *Proceedings of the IEEE conference on Computer Vision.* 991–998.

[18] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *proceeding of the IEEE conference on Computer Vision.* IEEE, 2980–2988.

[19] Mikael Henaff, Joan Bruna, and Yann Lecun. 2015. Deep Convolutional Networks on Graph-Structured Data. *Computer Science* (2015).

[20] Seunghoon Hong, Donghun Yeo, Suha Kwak, Honglak Lee, and Bohyung Han. 2017. Weakly supervised semantic segmentation using web-crawled videos. *arXiv preprint arXiv:1701.00352.*

[21] Ashesh Jain, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. 2016. Structural-RNN: Deep learning on spatio-temporal graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 5308–5317.

[22] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele. 2017. Simple Does It: Weakly Supervised Instance and Semantic Segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 1665–1674.

[23] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations (ICLR).*

[24] Suha Kwak, Seunghoon Hong, Bohyung Han, et al. 2017. Weakly Supervised Semantic Segmentation Using Superpixel Pooling Network.. In *AAAI.* 4111–4117.

[25] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. 2015. Gated Graph Sequence Neural Networks. *Computer Science* (2015).

[26] Xiaodan Liang, Xiaohui Shen, Jiashi Feng, Liang Lin, and Shuicheng Yan. 2016. Semantic object parsing with graph lstm. In *European Conference on Computer Vision.* Springer, 125–143.

[27] Xiaodan Liang, Xiaohui Shen, Donglai Xiang, Jiashi Feng, Liang Lin, and Shuicheng Yan. 2016. Semantic object parsing with local-global long short-term memory. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 3185–3193.

[28] Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. 2016. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 3159–3167.

[29] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 3431–3440.

[30] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. 2014. The Role of Context for Object Detection and Semantic Segmentation in the Wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.*

[31] George Papandreou, Liang-Chieh Chen, Kevin P Murphy, and Alan L Yuille. 2015. Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In *Proceedings of the IEEE international conference on computer vision.* 1742–1750.

[32] Deepak Pathak, Philipp Krahenbuhl, and Trevor Darrell. 2015. Constrained convolutional neural networks for weakly supervised segmentation. In *Proceedings of the IEEE international conference on computer vision.* 1796–1804.

[33] Pedro O Pinheiro and Ronan Collobert. 2015. From image-level to pixel-level labeling with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 1713–1721.

[34] Niloufar Pourian, S Karthikeyan, and BS Manjunath. 2015. Weakly supervised graph based semantic segmentation by learning communities of image-parts. In *Proceedings of the IEEE conference on computer vision.* IEEE, 1359–1367.

[35] Rong Quan, Junwei Han, Dingwen Zhang, and Feiping Nie. 2016. Object co-segmentation via graph optimized-flexible manifold ranking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 687–695.

[36] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *Computer Science* (2014).

[37] Meng Tang, Abdelaziz Djelouah, Federico Perazzi, Yuri Boykov, and Christopher Schroers. 2018. Normalized Cut Loss for Weakly-supervised CNN Segmentation. (2018). https://arxiv.org/abs/1804.01346

[38] Paul Vernaza and Manmohan Chandraker. 2017. Learning random-walk label propagation for weakly-supervised semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Vol. 3. 2953–2961.

[39] Remi Vieux, Jenny Benois-Pineau, Jean Philippe Domenger, and Achille Braquelaire. 2012. Segmentation-based multi-class semantic object detection. *Multimedia Tools and Applications* 60, 2 (2012), 305–326.

[40] Yuhang Wang, Jing Liu, Yong Li, and Hanqing Lu. 2015. Semi-and Weakly-Supervised Semantic Segmentation with Deep Convolutional Neural Networks. In *Proceedings of the 23rd ACM international conference on Multimedia.* ACM, 1223–1226.

[41] Y. Wei, J. Feng, X. Liang, M. M. Cheng, Y. Zhao, and S. Yan. 2017. Object Region Mining with Adversarial Erasing: A Simple Classification to Semantic Segmentation Approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 6488–6496.

[42] Yunchao Wei, Xiaodan Liang, Yunpeng Chen, Zequn Jie, Yanhui Xiao, Yao Zhao, and Shuicheng Yan. 2016. Learning to segment with image-level annotations. *Pattern Recognition* 59 (2016), 234–244.

[43] Yunchao Wei, Xiaodan Liang, Yunpeng Chen, Xiaohui Shen, Ming-Ming Cheng, Jiashi Feng, Yao Zhao, and Shuicheng Yan. 2017. Stc: A simple to complex framework for weakly-supervised semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence* 39, 11 (2017), 2314–2320.

[44] Xiwen Yao, Junwei Han, Gong Cheng, and Lei Guo. 2015. Semantic segmentation based on stacked discriminative autoencoders and context-constrained weakly supervised learning. In *Proceedings of the 23rd ACM international conference on Multimedia.* ACM, 1211–1214.

[45] Peng Ying, Jin Liu, Hanqing Lu, and Songde Ma. 2015. Exclusive Constrained Discriminative Learning for Weakly-Supervised Semantic Segmentation. In *Proceedings of the 23rd ACM international conference on Multimedia.* ACM, 1251–1254.

[46] F. Yu, V. Koltun, and T. Funkhouser. 2017. Dilated Residual Networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Vol. 1. 636–644.

[47] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. 2017. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2881–2890.

[48] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition.* IEEE, 2921–2929.