Associating Inter-Image Salient Instances for Weakly Supervised Semantic Segmentation

Ruochen Fan[1][0000−0003−1991−0146], Qibin Hou[2][0000−0002−8388−8708], Ming-Ming Cheng[2][0000−0001−5550−8758] Gang Yu[3][0000−0001−5570−2710], Ralph R. Martin[4], and Shi-Min Hu[1][0000−0001−7507−6542]

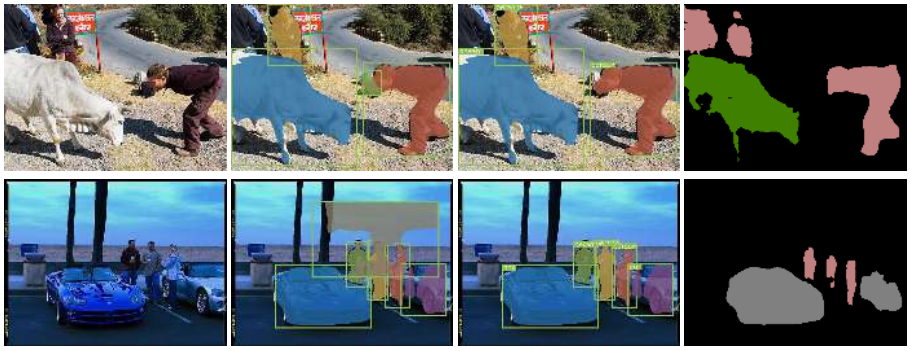
[1] Tsinghua University, Beijing, China {frc16@mails.,shimin@}tsinghua.edu.cn
[2] Nankai University, Tianjin, China cmm@nankai.edu.cn, andrewhoux@gmail.com
[3] Megvii Inc., Beijing, China yugang@megvii.com
[4] Cardiff University, Cardiff CF243AA, U.K. ralph@cs.cardiff.ac.uk

Abstract. Effectively bridging between image level keyword annotations and corresponding image pixels is one of the main challenges in weakly supervised semantic segmentation. In this paper, we use an instance-level salient object detector to automatically generate salient instances (candidate objects) for training images. Using similarity features extracted from each salient instance in the whole training set, we build a similarity graph, then use a graph partitioning algorithm to separate it into multiple subgraphs, each of which is associated with a single keyword (tag). Our graph-partitioning-based clustering algorithm allows us to consider the relationships between all salient instances in the training set as well as the information within them. We further show that with the help of attention information, our clustering algorithm is able to correct certain wrong assignments, leading to more accurate results. The proposed framework is general, and any state-of-the-art fully-supervised network structure can be incorporated to learn the segmentation network. When working with DeepLab for semantic segmentation, our method outperforms state-of-the-art weakly supervised alternatives by a large margin, achieving 65.6% mIoU on the PASCAL VOC 2012 dataset. We also combine our method with Mask R-CNN for instance segmentation, and demonstrated for the first time the ability of weakly supervised instance segmentation using only keyword annotations.

Keywords: Semantic segmentation, weak supervision, graph partitioning.

1 Introduction

Semantic segmentation, providing rich pixel level labeling of a scene, is one of the most important tasks in computer vision. The strong learning ability of convolutional neural networks (CNNs) has enabled significant progress in this field recently [5,27,29,46,47]. However, the performance of such CNN-based methods requires a large amount of training data annotated to pixel-level, *e.g.,* PASCAL VOC [11] and MS COCO [28]; such data are very expensive to collect. As an approach to alleviate the demand for pixel-accurate annotations, weakly supervised semantic segmentation has drawn great attention recently. Such methods merely require supervisions of one or more of the



(a) input images (b) salient instances (c) proxy GT (d) output results

Fig. 1: Input images (a) are fed into a salient instance detection method (*e.g.,* S^4Net [12]) giving instances shown in colour in (b). Our system automatically generates proxy ground-truth data (c) by assigning correct tags to salient instances and rejecting noisy instances. Traditional fully supervised semantic/instance segmentation methods learn from these proxy ground-truth data; final generated segmentation results are shown in (d).

following kinds: keywords [19, 22, 23, 42, 43], bounding boxes [36], scribbles [26], points [2], *etc.* , making the collection of annotated data much easier. In this paper, we consider weakly supervised semantic segmentation using only image-level keyword annotations.

In weakly supervised semantic segmentation, one of the main challenges is to effectively build a bridge between image-level keyword annotations and corresponding semantic objects. Most previous state-of-the-art methods focus on generating *proxy ground-truth* from the original images by utilizing low-level cue detectors to capture pixel-level information. This may be done using a saliency detector [4, 20, 22, 42] or attention models [4, 42], for example. Because these methods give only pixel-level saliency/attention information, it is difficult to distinguish different types of semantic objects from the heuristic cues produced. Thus, the ability to discriminate semantic instances is essential. With the rapid development of saliency detection algorithms, some saliency extractors, such as MSRNet [24] and S^4Net [12], are now not only able to predict gray-level salient objects but also instance-level masks. Inspired by the advantages of such instance-level salient object detectors, in this paper, we propose to carry out the instance distinguishing task in the early saliency detection stage, with the help of S^4Net, greatly simplifying the learning pipeline. Fig. 1(b) shows some instance-level saliency maps predicted by S^4Net.

In order to make use of the salient instance masks with their bounding boxes, two main obstacles need to be overcome. **Firstly**, an image may be labeled with multiple keywords, so determining a correct keyword (tag) for each class-agnostic salient instance is essential. For example, see Fig. 1(b): the upper image is associated with two image-level labels: ‘sheep’ and ‘person’. Allocating the correct tag to each detected instance is difficult. **Secondly**, not all salient instances generated by the salient instance

detector are semantically meaningful; incorporating such noisy instances would degrade downstream operations. For example, in the lower image in Fig. 1(b), an obvious noisy instance occurs in the sky (shown in gray). Such instances and the associated noisy labels frequently arise using current algorithms. Therefore, recognizing and excluding such noisy salient instances is important in our approach. The two obstacles described above can be regarded as posing a tag-assignment problem, *i.e.,* , associating salient instances, including both semantically meaningful and noisy ones, with correct tags.

In this paper, we take into consideration both the intrinsic properties of a salient instance and the semantic relationships between all salient instances in the whole training set. Here we use the term *intrinsic properties* of a salient instance to refer to the appearance information within its (single) region of interest. In fact, it is possible to predict a correct tag for a salient instance using only its intrinsic properties: see [19, 22, 42]. However, as well as the appearance information within each region of interest, there are also strong semantic relationships between all salient instances: salient instances in the same category typically share similar semantic features. We will show that taking this property into account is important in the tag-assignment operation in Section 5.2.

More specifically, our proposed framework contains an attention module to predict the probability of a salient instance belonging to a certain category, based on its intrinsic properties. On the other hand, to assess semantic relationships, we use a semantic feature extractor which can predict a semantic feature for each salient instance; salient instances sharing similar semantic information have close semantic feature vectors. Based on the semantic features, a similarity graph is built, in which the vertices represent salient instances and the edge weights record the semantic similarity between a pair of salient instances. We use a graph partitioning algorithm to divide the graph into subgraphs, each of which represents a specific category. The graph partitioning process is modelled as a mixed integer quadratic program (MIQP) problem [3], for which a globally optimal solution can be found. The aim is to make the vertices in each subgraph as similar as possible, while taking into account the intrinsic properties of the salient instances.

Our approach provides high-quality proxy-ground-truth data, which can be used to train any state-of-the-art fully-supervised semantic segmentation methods. When working with DeepLab [5] for semantic segmentation, our method obtains mean intersection-over-union (mIoU) of 65.6% for PASCAL VOC 2012 test set, beating the current state-of-the-art. In addition to pixel-level semantic segmentation, this paper demonstrated for the first time the ability of weakly supervised instance segmentation using only keyword annotations, by fitting our instance level proxy ground-truth data into latest instance segmentation network, *i.e.,* Mask R-CNN [14]. In summary, the main contributions of this paper are:

- the first use of salient instances in a weakly supervised segmentation framework, significantly simplifying object discrimination, and performing instance-level segmentation under weak supervision.
- a weakly supervised segmentation framework exploiting not only the information inside salient instances but also the relationships between all objects in the whole dataset.

2 Related Work

While longstanding research has considered fully supervised semantic segmentation, *e.g.,* [5, 27, 29, 46, 47], more recently, weakly-supervised semantic segmentation has come to the fore. Early work such as [41] relied on hand-crafted features, such as color, texture, and histogram information to build a graphical model. However, with the advent of convolutional neural network (CNN) methods, this conventional approach has been gradually replaced because of its lower performance on challenging benchmarks [11]. We thus only discuss weakly supervised semantic segmentation work based on CNNs.

In [32], Papandreou *et al.* use the expectation-maximization algorithm [8] to perform weakly-supervised semantic segmentation based on annotated bounding boxes and image-level labels. Similarly, Qi *et al.* [36] used proposals generated by Multi-scale Combinatorial Grouping (MCG) [35] to help localize semantically meaningful objects. Scribbles and points are further used as additional supervision. In [26], Lin *et al.* made use of a region-based graphical model, with scribbles providing ground-truth annotations to train the segmentation network. Bearman *et al.* [2] likewise leveraged knowledge from human-annotated points as supervision.

Other works rely only on image-level labels. Pathak *et al.* [33] addressed the weakly-supervised semantic segmentation problem by introducing a series of constraints. Pinheiro *et al.* [34] treated this problem as a multiple instance learning problem. In [23], three loss functions are designed to gradually expand the areas located by an attention model [48]. Wei *et al.* [42] improved this approach using an adversarial erasing scheme to acquire more meaningful regions that provide more accurate heuristic cues for training. In [43], Wei *et al.* presented a simple-to-complex framework which used saliency maps produced by the methods in [6, 21] as initial guides. Hou *et al.* [19] advanced this approach by combining the saliency maps [18] with attention maps [45]. More recently, Oh *et al.* [31] and Chaudhry *et al.* [4] considered linking saliency and attention cues together, but they adopted different strategies to acquire semantic objects. Roy and Todorovic [38] leveraged both bottom-up and top-down attention cues and fused them via a conditional random field as a recurrent network. Very recent work [17, 22] tackles the weakly-supervised semantic segmentation problem using images or videos from the Internet. Nevertheless, the ideas used to obtain heuristic cues are similar to those in previous works.

In this paper, differently from all the aforementioned methods, we propose a weakly supervised segmentation framework using salient instances. We assign tags to salient instances to generate proxy ground-truth for fully supervised segmentation network. The tag-assignment problem is modeled as graph partitioning, in which both the relationships between all salient instances in the whole dataset, as well as the information within them are taken into consideration.

3 Overview and Network Structure

We now present an overview of our pipeline, then discuss our network structure and tag-assignment algorithm. Our proposed framework is shown in Fig. 2. Most previous work which relies on pixel level cues (such as saliency, edges and attention maps) regards instance discrimination as a key task. However, with the development of deep

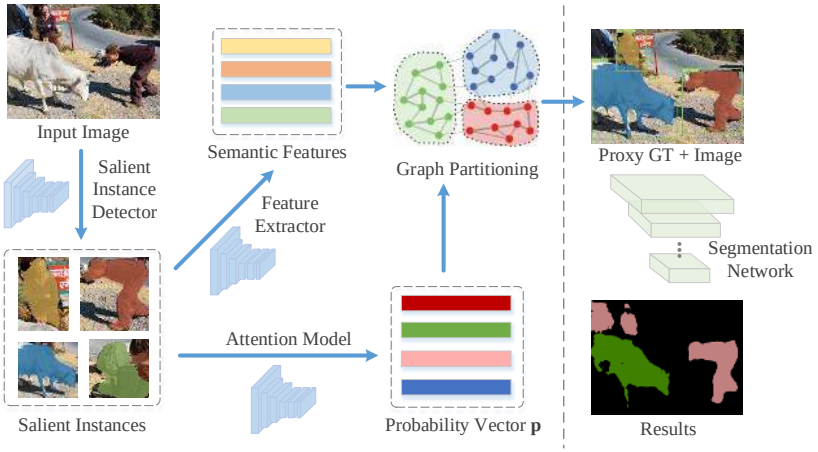


Fig. 2: Pipeline. Instances are extracted from the input images by a salient instance detector (*e.g.,* S^4Net [12]). An attention module predicts the probability of each salient instance belonging to a certain category using its intrinsic properties. Semantic features are obtained from the salient instances and used to build a similarity graph. Graph partitioning is used to determine the final tags of the salient instances. The fully supervised segmentation network (*e.g.,* DeepLab [5] or Mask R-CNN [14]) is trained using the proxy ground-truth generated.

learning, saliency detectors are now available that can predict saliency maps along with instance bounding boxes. Given training images labelled only with keywords, we use an instance-level saliency segmentation network, S^4Net [12], to extract salient instances from every image. Each salient instance has a bounding box and a mask indicating a visually noticeable foreground object in an image. These salient instances are class-agnostic, so the extractor S^4Net does not need to be trained for our training set. Although salient instances contain ground-truth masks for training a segmentation mask, there are two major limitations in the use of such salient instances to train a segmentation network. The first is that an image may be labelled by multiple keywords. For example, a common type of scene involves *pedestrians* walking near *cars*. Determining the correct keyword associated with each salient instance is necessary. The second is that instances detected by S^4Net may not fall into the categories in the training set. We refer to such salient instances as noisy instances. Eliminating such noisy instances is a necessary part of our complete pipeline. Both limitations can be removed by solving a tag-assignment problem, in which we associate salient instances with correct tags based on image keywords, and tag others as noisy instances.

Our pipeline takes into consideration both the intrinsic characteristics of a single region, and the relationships between all salient instances. A classification network responds strongly to discriminative areas (pixels) of an object in the score map for the correct category of the object. Therefore, inspired by class activation mapping (CAM) [48], we use an attention module to identify the tags of salient instances di-

rectly from their intrinsic characteristics. One weakness of existing weakly supervised segmentation work is that it treats the training set image by image, ignoring the relationships between salient instances across the entire training set. However, salient instances belonging to the same category share similar contextual information which is of use in tag-assignment. Our architecture extracts *semantic features* for each salient instance; regions with similar semantic information have similar semantic features. These are used to construct a similarity graph. The tag-assignment problem now becomes one of graph partitioning, making use not only of the intrinsic properties of a single salient instance, but the global relationships between all salient instances.

3.1 Attention Module

The attention module in our pipeline is used to determine the correct tag for each salient instance from its intrinsic characteristics. Formally, let C be the number of categories (excluding the background) in the training set. Given an image I, the attention module predicts C attention maps. Each pixel in a map indicates the probability that the pixel belongs to the corresponding object category. Following FCAN [4], we make use of a fully convolutional network as our classifier. After prediction of C score maps by the backbone model, *e.g.,* off the shelf VGG16 [40] or ResNet101 [15], the classification result $\mathbf{y}$ is output by a sigmoid layer fed with the average of the score maps using a global average pooling (GAP) layer. Notice that $\mathbf{y}$ is not a probability distribution, as the input image may have multiple keywords. An attention map denoted by A_i can be produced by feeding the i-th score map into a sigmoid layer. As images may be associated with multiple keywords, we treat network optimization as C independent binary classification problems. Thus, the loss function is:

$$L_a = -\frac{1}{C}\sum_{i}^{C}(\bar{\mathbf{y}}_\mathbf{i}\log \mathbf{y}_\mathbf{i} + (1-\bar{\mathbf{y}}_\mathbf{i})\log(1-\mathbf{y}_\mathbf{i})), \tag{1}$$

where $\bar{\mathbf{y}}_\mathbf{i}$ denotes the keyword ground-truth. The dataset for weakly supervised semantic segmentation is used to train the classifier, after which the attention maps for the images in this dataset can be obtained.

Assuming that a salient instance has a bounding box (x_0, y_0, x_1, y_1) in image I, the probability of this salient instance belonging to the i-th category $\mathbf{p}_\mathbf{i}$ is:

$$\mathbf{p}_\mathbf{i} = -\frac{1}{(x_1-x_0)(y_1-y_0)}\sum_{x=x_0}^{x_1}\sum_{y=y_0}^{y_1}A_i(x,y), \tag{2}$$

and the tag for this salient instance is given by $\arg\max(\mathbf{p})$.

3.2 Semantic Feature Extractor

The attention module introduced above assigns tags to salient instances from their intrinsic properties, but fails to take relationships between all salient instances into consideration. To discover such relationships, we use a semantic feature extractor to produce

feature vectors for each input region of interest, such that regions of interest with similar semantic content share similar features. To avoid the need for additional data, we use ImageNet [9] to train this model.

The network architecture of the semantic feature extractor is very similar to that of a standard classifier. ResNet [16] is used as the backbone model. We add a GAP layer after the last layer of ResNet to obtain a 2048-channel semantic feature vector $\mathbf{f}$. During the training phase, a 1000-dimensional auxiliary classification vector $\mathbf{y}$ is predicted by feeding $\mathbf{f}$ into a 1×1 convolutional layer.

Our training objective is to maximize the distance between features from regions of interest with different semantic content and minimize the distance between features from the same category. To this end, in addition to the standard softmax-cross entropy classification loss, we employ center loss [44] to directly concentrate features on similar semantic content. For a specific category of ImageNet, the standard classification loss trains $\mathbf{y}$ to be the correct probabilistic distribution, and the center loss simultaneously learns a center $\mathbf{c}$ for the semantic features and penalizes the distance between $\mathbf{f}$ and $\mathbf{c}$. The overall loss function is formulated as:

$$L = L_{cls} + \lambda L_c, \qquad L_c = 1 - \frac{\mathbf{f} \cdot \mathbf{c}_{\bar{\mathbf{y}}}}{\|\mathbf{f}\| \, \|\mathbf{c}_{\bar{\mathbf{y}}}\|}, \tag{3}$$

where L_{cls} is the softmax-crossentropy loss, $\bar{y}$ is the ground-truth label of a training sample and $\mathbf{c}_{\bar{\mathbf{y}}}$ is the center of the $\bar{y}$-th category.

In every training iteration, the center for the category of the input sample is updated using:

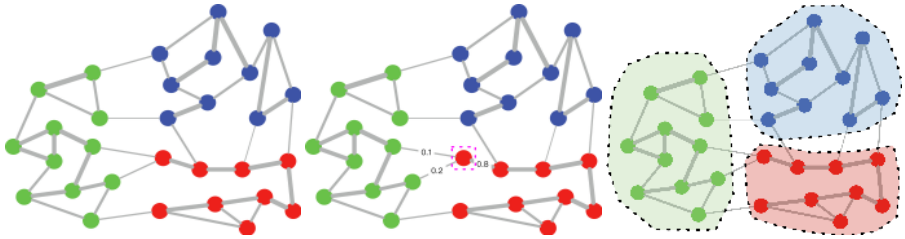
$$\mathbf{c}_{\bar{y}}^{t+1} = \mathbf{c}_{\bar{y}}^{t} + \alpha \cdot (\mathbf{f} - \mathbf{c}_{\bar{y}}^{t}), \tag{4}$$

4 Tag-Assignment Algorithm

In order to assign a correct keyword to every salient instance with or identify it as a noisy instance, we use a tag-assignment algorithm, exploiting both the intrinsic properties of a single salient instance, and the relationships between all salient instances in the whole dataset. The tag-assignment process is modeled as a graph partitioning problem . Although the purpose of graph partitioning can be considered as clustering, traditional clustering algorithms using a hierarchical approach [37], k-means [30], DBSCAN [10] or OPTICS [1], are unsuited to our task as they only consider relationships between input data points, and ignore the intrinsic properties of each data point.

In detail, assume that n salient instances have been produced from the training set by S^{4}Net, and n semantic features extracted for each salient instance, denoted as $\mathbf{f}_j$, $j = 1, \ldots, n$. As Sec. 3.1 described, we predict the probability of every salient instance j belonging to category i, written as $\mathbf{p}_{ij}$, $i = 0, \ldots, C, j = 1, \ldots, n$, where category 0 means the salient instance is a noisy one.

Let the image keywords for a salient instance j be the set K_j. The purpose of the tag-assignment algorithm is to predict the final tags of the salient instances $\mathbf{x}_{ij}, i = 0, \ldots, C, j = 1, \ldots, n$, such that $\mathbf{x}_{ij} \in \{0, 1\}$ if $i \in K_j$ and otherwise $\mathbf{x}_{ij} \in \{0\}$, and $\sum_i \mathbf{x}_{ij} = \mathbf{1}$, where $\mathbf{x}_{0j} = 1$ means that instance j is considered noisy.



(a) Similarity graph (b) A vertex in detail (c) Subgraphs

Fig. 3: Graph partitioning. (a): similarity graph, thickness of edges indicating edge weights; color shows the correct tags of the vertices. (b): consider the vertex bounded by a dotted square— only by including it in the red subgraph can the objective be optimized. (c): subgraphs after partitioning.

We associate semantic similarity with the edges of a weighted undirected similarity graph having a vertex for each salient instance, and an edge for each pair of salient instances which are strongly similar. Edge weights give the similarity of a salient instance pair. Tag-assignment thus becomes a graph partitioning process. The vertices are partitioned into C subsets, each representing a specific category; their vertices are tagged accordingly. As salient instances in the same category have similar semantic content and semantic features, a graph partitioning algorithm should ensure the vertices inside a subset are strongly related while the vertices in different subsets should be as weakly related as possible. We define the cohesiveness of a specific subgraph as the sum of edge weights linking vertices inside the subgraph; the optimization target is to maximize the sum of cohesiveness over all categories. This graph partitioning problem can be modeled as a mixed integer quadratic program (MIQP) problem as described later.

4.1 Construction of the Similarity Graph

Let the similarity graph of vertices, edges and weights be $G = (V, E, W)$. Initially, we calculate the cosine similarity between every pair of features to determine W:

$$\begin{cases} W_{ij} = \frac{\mathbf{f_i} \cdot \mathbf{f_j}}{\|\mathbf{f_i}\| \|\mathbf{f_j}\|} + 1, & i \neq j, \\ W_{ij} = 0, & i = j, \end{cases} \tag{5}$$

If every pair of vertices is related by an edge, G would be a dense graph, the number of edges growing quadratically with the number of vertices, and in turn, cohesiveness would be dominated by the number of vertices in the subset. In order to eliminate the effect of the size of the subgraph, we turn G into a sparse graph by edge reduction, so that each vertex retains only those k linked edges with the largest weights. In our experiments, we set $k = 3$.

4.2 The Primary Graph Partitioning Algorithm

As described above, the cohesiveness of a subset i can be written in matrix form as $\mathbf{x}_i^T W \mathbf{x}_i$. As x_i is a binary vector with length n, this formula simply sums the weights of

edges between all vertices in subgraph i. To maximize cohesiveness over all categories, we formulate the following optimization problem:

$$\begin{aligned} &\max_{\mathbf{x}} \sum_{i=1}^{C} \mathbf{x}_i^T W \mathbf{x}_i, \qquad \text{such that} \\ &\text{s.t.} \sum_{i=1}^{C} \mathbf{x}_i = \mathbf{1}, \\ &\mathbf{x}_{ij} \in \begin{cases} \{0,1\} & \text{if } i \in K_j \\ \{0\} & \text{otherwise.} \end{cases} \end{aligned} \tag{6}$$

To further explain this formulation, consider a salient instance, such as the vertex bounded by dotted square in Figure 3(b), which belongs to category i_a. Sharing similar semantic content, the vertex representing this salient instance has strong similarity with the vertices in subset i_a. So the weights of edges between this vertex and subset i_a are larger than between it and any other subset, such as i_b. The objective of the optimization problem reaches a maximum if and only if this vertex is partitioned into subset i_a, meaning that the salient instance is assigned a correct tag.

This optimization problem can easily be transformed into a standard mixed integer quadratic programing (MIQP) problem. Although this MIQP is nonconvex because of its zero diagonal and nonnegative elements, it can easily be reformulated as a convex MIQP, since all the variables are constrained to be 0 or 1. It can be solved by a branch-and-bound method using IBM-CPLEX [3].

4.3 The Graph Partitioning with Attention and Noisy Vertices

The tag assignment problem in Section 4.2 identifies keywords for salient instances using semantic relationships between the salient instances. However, the intrinsic properties of a salient instance are also important in tag assignment. As explained in Section 3.1, the attention module predicts the probability $\mathbf{p}_{ij}$ that a salient instance j belongs to category i. In order to make use of the intrinsic characteristics of the salient instances, we reformulate the optimization problem as:

$$\begin{aligned} &\max_{\mathbf{x}} \sum_{i=1}^{C} \mathbf{x}_i^T W \mathbf{x}_i + \beta \mathbf{p}_i \mathbf{x}_i, \qquad \text{such that} \\ &\sum_{i=1}^{C} \mathbf{x}_i = \mathbf{1}, \\ &\mathbf{x}_{ij} \in \begin{cases} \{0,1\} & \text{if } i \in K_j \\ \{0\} & \text{otherwise,} \end{cases} \end{aligned} \tag{7}$$

where the hyper-parameter β balances intrinsic instance information and global object relationship information.

As the salient instances are obtained by the class-agnostic S^4Net, some salient instances may fall outside the categories of the training set. We should thus further adjust the optimization problem to reject such noisy vertices:

$$\begin{aligned} &\max_{\mathbf{x}} \sum_{i=1}^{C} \mathbf{x}_i^T W \mathbf{x}_i + \beta \mathbf{p}_i \mathbf{x}_i, \qquad \text{such that} \\ &\sum_{i=1}^{C} \mathbf{x}_i \leq \mathbf{1}, \\ &\sum_{i=1j} \mathbf{x}_{ij} = \lfloor rn \rfloor, \\ &\mathbf{x}_{ij} \in \begin{cases} \{0,1\} & \text{if } i \in K_j \\ \{0\} & \text{otherwise,} \end{cases} \end{aligned} \tag{8}$$

where the retention ratio r determines the number of vertices recognized as non-noisy.

5 Experiments

In this section, we show the efficacy of our method on the challenging PASCAL VOC 2012 semantic segmentation benchmark and at the same time conduct comparisons with state-of-the-art methods. The results show that our proposed framework greatly outperforms all existing weakly-supervised methods. We also perform a series of experiments to analyze the importance of each component in our method and discuss limitations highlighted by the experiments. We furthermore present the first results of instance-level segmentation for MS COCO.

5.1 Methodology

Datasets. We consider two training sets widely used in other work, the PASCAL VOC 2012 semantic segmentation dataset [11] plus an augmented version of this set [13]. As it has been widely used as a main training set [4, 23, 42], we also do so. We also consider a simple dataset [19], all of whose images were automatically selected from the ImageNet dataset [39]. We show the results of training on both sets individually, as well as in combination. Details concerning the datasets can be found in Tab. 1b. We have tested our method on both the PASCAL VOC 2012 validation set and test set. For instance-level segmentation, the training process is performed on the standard COCO trainval set; all pixel-level masks in the ground-truth are removed. We evaluate the performance using the standard COCO evaluation metric. We use ImageNet as an auxiliary dataset to pretrain all backbone models and the feature extractor.

Hyper-Parameters and Model Settings. In order to concentrate feature vectors for salient instances in the same category, we use center loss. As suggested in [44], we set $\lambda = 10^{-3}$ and $\alpha = 0.5$ to train center loss. However, unlike in the original version, center loss is calculated by cosine distance instead of Euclidean distance for consistency

Table 1: Ablation study for our proposed framework on three datasets. The best result in each column is highlighted in **bold**. Subscripts represent growth relative to the value above. Numbers of samples in the three datasets are also given.

	mIoU (%)		
Methods	VOC	SI	VOC+SI
random	56.4	–	61.3
attention	$62.0_{+5.6}$	–	$62.7_{+1.4}$
GP w/o filtering	$64.0_{+2.0}$	62.8	$64.9_{+2.2}$
GP + filtering	$\mathbf{64.5}_{+0.5}$	$\mathbf{63.9}_{+1.1}$	$\mathbf{65.6}_{+0.7}$

(a) **Ablation results** 'Random' refers to keywords of an image being assigned randomly to the salient instances. 'Attention' stands for the framework using only the attention module. The results of the whole pipeline with or without noisy salient instance filtering are also given.

dataset	size
VOC	10,582
SI	24,000
VOC + SI	34,582

(b) **Size of each dataset** In the experiments, we use 10,582 images from the augmented PASCAL VOC 2012 dataset, and 24,000 from the simple ImageNet dataset.

Table 2: Influence of the hyper-parameters β and r on graph partitioning. The best result for each hyper-parameter is highlighted in **bold**. This experiment is conducted on the PASCAL VOC dataset.

β	0	3	10	30	90	300
mIoU (%)	63.2	63.9	64.1	**64.5**	63.6	62.9

(a) **Influence of** β The hyper-parameter β balances instance intrinsic information and global object relationship information in the optimization model. $\beta = 0$ means the graph is partitioned solely using global relationship information.

r	1.00	0.95	0.90	0.85	0.80	0.75
mIoU (%)	63.8	**64.5**	64.1	63.4	62.3	60.9

(b) **Influence of** r The retention ratio r determines the proportion of salient instances labeled as valid during graph partitioning. $r = 0$ means a tag-assignment algorithm without noisy instances filtering.

with the distance measure used in similarity graph construction. The semantic feature extractor is trained on ImageNet using input images cropped and resized to 224×224 pixels. The attention module is implemented as a standard classifier and ResNet-50 is used as the backbone model. We use all the training data (PASCAL VOC 2012 or simple ImageNet) to train this module. For the traditional fully supervised segmentation CNNs in our framework, we train DeepLab using the following hyper-parameters: initial learning rate = 2.5×10^{-4}), divided by a factor of 10 after 20k iterations, weight decay = 5×10^{-4}, and momentum = 0.9. The mask-RCNN for instance-level segmentation is trained using: initial learning rate = 2×10^{-3}, divided by a factor of 10 after 5 epochs, weight decay = 10^{-4}, and momentum = 0.9.

5.2 Sensitivity Analysis

To analyze the importance of each component of our proposed framework, we perform a series of ablation experiments using three datasets. Tab. 1a shows the results of the

ablation study. As for existing works, the PASCAL VOC 2012 training set (VOC) [11] is used in our experiments. Also, the simple ImageNet (SI) used important dataset in our experiments. Unlike in PASCAL VOC 2012, in the simple ImageNet dataset every image has only one keyword. The results in Tab. 1a are evaluated on PASCAL VOC test set and the results in Tab. 2 are evaluated on PASCAL VOC val set.

Importance of each component of the framework Figure 1a shows that it is impossible to obtain reasonable results by assign the image keywords to instances randomly, indicating the necessity of tag assignment. One can observe from Tab. 1a that the proposed graph partitioning operation brings 2.2% improvement compared to the single attention module for the combined PASCAL VOC and simple ImageNet dataset. These results indicate that global object relationship information across the whole dataset is useful in tag-assignment. and clearly contributes to the final segmentation performance. The results on the three datasets, especially for the simple ImageNet set which contains more noisy salient instances, show that the noise filtering mechanism further improves segmentation performance.

Balancing ratio β Graph partitioning depends on two key hyper-parameters: balancing ratio β and retention ratio r, and they have great impact on the final performance of the whole framework. The balancing ratio β balances information within salient instances to global object relationship information across the whole dataset. If β is set to 0, graph partitioning depends solely on the global relationship information; as β increases, the influence of the intrinsic properties of the salient instances also increases. Tab. 2a shows the influence of β. Even using only global relationship information ($\beta = 0$), reasonable results can still be obtained. This verifies the effectiveness and importance of the global relationship information. When $\beta = 30$, 1.3% performance gain is obtained as intrinsic properties of the salient instances are also taken into consideration during graph partitioning. Too large a value of β decreases use of global relationship information and may impair the final performance.

Retention ratio r The other key hyper-parameter, the retention ratio r, determines the proportion of salient instances to be regarded as valid in graph partitioning, as a proportion $(1 - r)$ of the instances are rejected as noise. Tab. 2b shows the influence of r on PASCAL VOC val set. Eliminating a proper number of salient instances having low confidence improves the quality of the proxy-ground-truth and benefits the final segmentation results, but too small a retention ratio leads to a performance decline.

5.3 Comparison with Existing Work

We compare our proposed method with existing state-of-the-art weakly supervised semantic segmentation approaches. Tab. 3 shows results based on the PASCAL VOC 2012 'val' and 'test' sets. We can see that our framework achieves the best results for both 'val' and 'test' sets. Specifically, our approach improves on the baseline result presented in Mining Pixels [19] by 6.0% points for the 'test' set and 5.8% for the 'val' set. It is

Table 3: Pixel-level segmentation results on the PASCAL VOC 2012 'val' and 'test' sets compared to those from existing state-of-the-art approaches. The default training dataset is VOC 2012 for our proposed framework, while '†' indicates experiments using both VOC 2012 and the simple ImageNet dataset. The best keyword-based result in each column is highlighted in **bold**.

Method	Publication	Supervision			Dataset	
		keywords	scribbles	points	val	test
CCNN [33]	ICCV'15	✓			35.3%	-
EM-Adapt [32]	ICCV'15	✓			38.2%	39.6%
MIL [34]	CVPR'15	✓			42.0%	-
SEC [23]	ECCV'16	✓			50.7%	51.7%
AugFeed [36]	ECCV'16	✓			54.3%	55.5%
STC [43]	PAMI'17	✓			49.8%	51.2%
Roy et al. [38]	CVPR'17	✓			52.8%	53.7%
Oh et al. [31]	CVPR'17	✓			55.7%	56.7%
AS-PSL [42]	CVPR'17	✓			55.0%	55.7%
WebS-i2 [22]	CVPR'17	✓			53.4%	55.3%
DCSP-VGG16 [4]	BMVC'17	✓			58.6%	59.2%
Mining Pixels [19]	EMMCVPR'17	✓			58.7%	59.6%
ours-VGG16 (Ours)	-	✓			61.3%	62.1%
ours-ResNet101	-	✓			63.6%	64.5%
ours-VGG16† (Ours)	-	✓			61.9%	63.1%
ours-ResNet101†	-	✓			**64.5**%	**65.6**%
ScribbleSup [26]	CVPR'16	✓	✓		63.1%	-
Bearman et al. [2]	ECCV'16	✓		✓	49.1%	-

Table 4: Instance segmentation results on the COCO test-dev set compared to those of existing approaches. The training set for our weakly supervised framework is the COCO training set without pixel level annotations (masks).

Method	weakly	fully	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
FCIS [25]		✓	29.2%	49.5%	-	7.1%	31.3%	50.0%
MNC [7]		✓	24.6%	44.3%	24.8%	4.7%	25.9%	43.6%
Mask-RCNN [14]		✓	37.1%	60.0%	39.6%	35.3%	35.3%	35.3%
ours	✓		13.7%	25.5%	13.5%	00.7%	15.7%	26.1%

further worth noting that our framework even outperforms the methods with additional supervision in the form of scribbles and points.

In addition to the semantic segmentation results, we present results for instance-level segmentation under weak supervision using only keyword annotations. Tab. 4

compares our results to those from state-of-the-art fully supervised methods. Using only original RGB images with keywords, our method achieves results within 36.9% of the best fully supervised method.

5.4 Efficiency Analysis

We use IBM-CPLEX [3] to solve the MIQP in graph partitioning process. Because our academic version CPLEX restricts the maximum number of variables to be optimized, we use batches of 400 salient instances in implementation. To assign tags for 18878 salient instances extracted from VOC dataset, $\lceil 18878/400 \rceil = 48$ batches are processed sequentially, which takes 226M memory and 22.14s on an i7 4770HQ CPU.

6 Conclusions

We have proposed a novel weakly supervised segmentation framework, focusing on generating accurate proxy-ground-truth based on salient instances extracted from the training images and tags assigned to them. In this paper, we introduce salient instances to weakly supervised segmentation, significantly simplifying the object discrimination operation in existing work and enabling our framework to conduct instance-level segmentation. We regard the tag-assignment task as a network partitioning problem which can be solved by a standard approach. In order to improve the accuracy of tag-assignment, both the information from individual salient instances, and from the relationships between all objects in the whole dataset are taken into consideration. Experiments show that our method achieves new state-of-the art results on the PASCAL VOC 2012 semantic segmentation benchmark and demonstrated for the first time weakly supervised results on the MS COCO instance-level segmentation task using only keyword annotations.

Acknowledgments

This research was supported by the Natural Science Foundation of China (Project Number 61521002, 61620106008, 61572264) and the Joint NSFC-ISF Research Program (project number 61561146393), the national youth talent support program, Tianjin Natural Science Foundation for Distinguished Young Scholars (NO. 17JCJQJC43700), Huawei Innovation Research Program.

References

1. Ankerst, M., Breunig, M.M., Kriegel, H.P., Sander, J.: Optics: ordering points to identify the clustering structure. In: ACM Sigmod record. vol. 28, pp. 49–60. ACM (1999) 7
2. Bearman, A., Russakovsky, O., Ferrari, V., Fei-Fei, L.: Whats the point: Semantic segmentation with point supervision. In: ECCV. pp. 549–565 (2016) 2, 4, 13
3. Bliek1ú, C., Bonami, P., Lodi, A.: Solving mixed-integer quadratic programming problems with ibm-cplex: a progress report. In: Proceedings of the twenty-sixth RAMP symposium. pp. 16–17 (2014) 3, 9, 14

4. Chaudhry, A., Dokania, P.K., Torr, P.H.: Discovering class-specific pixels for weakly-supervised semantic segmentation. BMVC (2017) 2, 4, 6, 10, 13
5. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE TPAMI (2017) 1, 3, 4, 5
6. Cheng, M., Mitra, N.J., Huang, X., Torr, P.H., Hu, S.: Global contrast based salient region detection. IEEE TPAMI (2015) 4
7. Dai, J., He, K., Sun, J.: Instance-aware semantic segmentation via multi-task network cascades. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3150–3158 (2016) 13
8. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. Journal of the royal statistical society. Series B (methodological) pp. 1–38 (1977) 4
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 248–255. IEEE (2009) 7
10. Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Kdd. vol. 96, pp. 226–231 (1996) 7
11. Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. IJCV (2015) 1, 4, 10, 12
12. Fan, R., Hou, Q., Cheng, M.M., Mu, T.J., Hu, S.M.: s^4: Single stage salient-instance segmentation. arXiv preprint arXiv:1711.07618 (2017) 2, 5
13. Hariharan, B., Arbeláez, P., Bourdev, L., Maji, S., Malik, J.: Semantic contours from inverse detectors. In: ICCV (2011) 10
14. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Computer Vision (ICCV), 2017 IEEE International Conference on. pp. 2980–2988. IEEE (2017) 3, 5, 13
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016) 6
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) 7
17. Hong, S., Yeo, D., Kwak, S., Lee, H., Han, B.: Weakly supervised semantic segmentation using web-crawled videos. In: CVPR (2017) 4
18. Hou, Q., Cheng, M.M., Hu, X., Borji, A., Tu, Z., Torr, P.: Deeply supervised salient object detection with short connections. In: CVPR (2017) 4
19. Hou, Q., Dokania, P.K., Massiceti, D., Wei, Y., Cheng, M.M., Torr, P.: Bottom-up top-down cues for weakly-supervised semantic segmentation. EMMCVPR (2017) 2, 3, 4, 10, 12, 13
20. Hou, Q., Dokania, P.K., Massiceti, D., Wei, Y., Cheng, M.M., Torr, P.: Bottom-up top-down cues for weakly-supervised semantic segmentation. arXiv preprint arXiv:1612.02101 (2016) 2
21. Jiang, H., Wang, J., Yuan, Z., Wu, Y., Zheng, N., Li, S.: Salient object detection: A discriminative regional feature integration approach. In: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. pp. 2083–2090. IEEE (2013) 4
22. Jin, B., Ortiz Segovia, M.V., Susstrunk, S.: Webly supervised semantic segmentation. In: CVPR. pp. 3626–3635 (2017) 2, 3, 4, 13
23. Kolesnikov, A., Lampert, C.H.: Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In: ECCV (2016) 2, 4, 10, 13
24. Li, G., Xie, Y., Lin, L., Yu, Y.: Instance-level salient object segmentation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 247–256. IEEE (2017) 2

25. Li, Y., Qi, H., Dai, J., Ji, X., Wei, Y.: Fully convolutional instance-aware semantic segmentation. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 2359–2367 (2017) 13
26. Lin, D., Dai, J., Jia, J., He, K., Sun, J.: Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In: CVPR (2016) 2, 4, 13
27. Lin, G., Milan, A., Shen, C., Reid, I.: Refinenet: Multi-path refinement networks with identity mappings for high-resolution semantic segmentation. In: CVPR (2017) 1, 4
28. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014) 1
29. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR (2015) 1, 4
30. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. vol. 1, pp. 281–297. Oakland, CA, USA (1967) 7
31. Oh, S.J., Benenson, R., Khoreva, A., Akata, Z., Fritz, M., Schiele, B.: Exploiting saliency for object segmentation from image level labels. In: CVPR (2017) 4, 13
32. Papandreou, G., Chen, L.C., Murphy, K., Yuille, A.L.: Weakly-and semi-supervised learning of a dcnn for semantic image segmentation. arXiv preprint arXiv:1502.02734 (2015) 4, 13
33. Pathak, D., Krahenbuhl, P., Darrell, T.: Constrained convolutional neural networks for weakly supervised segmentation. In: ICCV (2015) 4, 13
34. Pinheiro, P.O., Collobert, R.: From image-level to pixel-level labeling with convolutional networks. In: CVPR (2015) 4, 13
35. Pont-Tuset, J., Arbelaez, P., Barron, J.T., Marques, F., Malik, J.: Multiscale combinatorial grouping for image segmentation and object proposal generation. IEEE TPAMI (2017) 4
36. Qi, X., Liu, Z., Shi, J., Zhao, H., Jia, J.: Augmented feedback in semantic segmentation under image level supervision. In: ECCV (2016) 2, 4, 13
37. Rokach, L., Maimon, O.: Clustering methods. In: Data mining and knowledge discovery handbook, pp. 321–352. Springer (2005) 7
38. Roy, A., Todorovic, S.: Combining bottom-up, top-down, and smoothness cues for weakly supervised image segmentation. In: CVPR (2017) 4, 13
39. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. IJCV (2015) 10
40. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015) 6
41. Vezhnevets, A., Ferrari, V., Buhmann, J.M.: Weakly supervised structured output learning for semantic segmentation. In: CVPR. pp. 845–852. IEEE (2012) 4
42. Wei, Y., Feng, J., Liang, X., Cheng, M.M., Zhao, Y., Yan, S.: Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In: CVPR (2017) 2, 3, 4, 10, 13
43. Wei, Y., Liang, X., Chen, Y., Shen, X., Cheng, M.M., Feng, J., Zhao, Y., Yan, S.: Stc: A simple to complex framework for weakly-supervised semantic segmentation. IEEE TPAMI (2016) 2, 4, 13
44. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A discriminative feature learning approach for deep face recognition. In: European Conference on Computer Vision. pp. 499–515. Springer (2016) 7, 10
45. Zhang, J., Lin, Z., Brandt, J., Shen, X., Sclaroff, S.: Top-down neural attention by excitation backprop. In: ECCV (2016) 4
46. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR (2017) 1, 4

47. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., Torr, P.H.: Conditional random fields as recurrent neural networks. In: ICCV (2015) 1, 4
48. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR (2016) 4, 5