Simultaneous Semantic Segmentation of a Set of Partially Labeled Images

Qiongjie Tian
Computer Science and Engineering
Arizona State University
qtian5@asu.edu

Baoxin Li
Computer Science and Engineering
Arizona State University
baoxin.li@asu.edu

Abstract

Semantic segmentation, by which an image is decomposed into regions with their respective semantic labels, is often the first step towards image understanding. Existing research on this regard is mainly performed under two conditions: the fully-supervised setting that relies on a set of images with pixel-level labels and the weakly-supervised one that uses only image-level labels. In both cases, the labeling task is time-consuming and laborious, and thus training data are always limited. In practice, there are voluminous on-line images, which unfortunately often have only incomplete image-level labels (tags) but would otherwise be potentially useful for a learning-based algorithm. Only limited efforts have been attempted on using such coarsely and incompletely labelled data for semantic segmentation. This paper proposes a new approach to semantic segmentation of a set of partially-labelled images, using a formulation considering information from multiple visual similar images. Experiments on several popular datasets, with comparison with existing methods, demonstrate evident performance improvement of the proposed approach.

1. Introduction

In the era of Internet and social media, there are more and more images posted on-line. Often, such on-line data lack sufficient textual annotation desired by learning-based algorithms. To make such data more useful, efforts have been devoted towards tasks like image sentiment analysis [27], image tagging[5][7][28] and image classification[35][26], targeting at producing labels for the images. In the labeling effort the finest granularity one could achieve is to perform semantic segmentation [3], which may classify each pixel in one image into a proper class/label. Both fully-supervised and weakly-supervised approaches exist.

In the fully-supervised setting, a set of images with pixel-level labels are available. In [21], all pixels in one superpixel are assumed to have the same label and Markov Random Field (MRF) was used to capture the context information to help improve the local superpixel-level labeling. Limited availability of fully-labeled data is a practical constraint for such approach. In [20][21], region-based cues are used to build exemplar-SVMs to gain the final labeling. However, there is one obvious disadvantage: users have to label each pixel in the dataset, which is time-consuming and involves a lot of manual work. In the weakly-supervised setting, data with only image-level labels are assumed. Most existing work further assumes that the labels are "complete" in the sense that the image-level label set for a given image contains all possible labels we may assign to any pixel in that image. This setting has been used in [29][11][25].

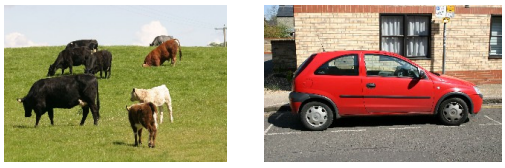

Figure 1. Two images with partially and fully image-level labels

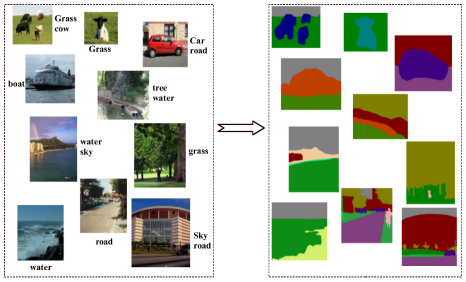

Figure 2. Illustrating the problem studied in this paper: the left panel represents the input to our algorithm, which are a set of images with partial image-level labels (one demo shown in Figure 1), and the right panel is the output of segmented images with labeled pixels. A formal problem definition is shown in Section 3.

The abundance of images with tags on social media platforms provides the opportunity for obtaining large-scale training sets without laborious manual labelling. However, in reality, even if we may be able to obtain a lot of images

with a desired set of semantic tags (and use the tags as semantic labels for simplicity), the majority of on-line images would still have only *incomplete* image-level labels, especially for user-generated images. That is, it is unrealistic to expect tags associated with an on-line image would happen to cover *all* semantic concepts we need to employ for segmentation. Therefore, in order to utilize the vast on-line images, we face the task of how to label each pixel in each image (i.e., semantic segmentation), given a set of images with partial image-level labels. Figure shows a demo of one image with partially image-level labels, while our task is illustrated in Figure 2. One similar work is [34], which only considers using information from one image only and does not consider the fact that visually similar superpixels across different images also are likely to have the same labels. In this paper, we work on this problem from one new aspect by proposing an approach based on conditional random fields (CRFs), which attempts to employ all possible sources of information in the dataset to deal with the challenge of incomplete labels.

The contributions of the paper are as follows. First, we propose a novel formulation for a new problem of semantic segmentation with partial image-level labels. Second, under the proposed multi-image model, we propose an efficient solution and demonstrate with comparative experiments its effectiveness.

The organization of the remainder of the paper is as follows. We first give a brief literature review on related works in Section 2. Then, a detailed description of the problem and our proposed approach are provided in Section 3. To show the performance of our proposed method, experiments are reported in Section 4. We conclude our work and present our future work in Section 5.

2. Related Works

We briefly review below two classes of related research on semantic segmentation: those relying on fully-supervised learning and those utilizing only weakly-supervised learning. As is evident from the following discussion, the distinction between these two classes of approaches is mainly on the granularity of labelling for the training data.

2.1. Fully-supervised Semantic Segmentation

As described in Section 1, in fully-supervised semantic segmentation, labels of each pixel or superpixel in the training set are known. There are a lot of existing efforts on this regard. In [19], Jamie Shotton *et al.* proposed semantic texton forests to do semantic segmentation using a bag-of-semantic-textons model, where only simple features of superpixels were used. To improve the performance, some other approaches attempt to consider neighboring information of different superpixels. In [9], Pushmeet Kohli *et al.* proposed to use higher order CRFs to capture such information of a set of pixels. Since high-order CRF models do not model the relevance of semantic labels, in [13], Heesoo Myeong *et al.* proposed to use high-order semantic relations to capture the context information in images and then transfer semantic labels from a labeled image to another unlabeled image. Besides tree-structure algorithms and graphical models (like CRF, MRF), active learning and deep learning are also applied to semantic segmentation recently. In [16], Gemma Roig *et al.* proposed a MAP inference method based on active learning, which is in fact one semi-supervised method. In [18], to improve the Recursive Context Propagation Network (RCPN), two revisions were made: one is to solve the potential problem because of the special structure of RCPN, which can help reduce the complexity of the network structure; the other is to consider the context information by building a Markov Random Field on the modified structure. This is one recent work on applying deep network to capture the context information of different superpixels for semantic segmentation.

Obviously, one key limitation of the fully-supervised approaches is the requirement of a set of images with pixel-level (or superpixel-level) labels. Due to the cost associated labeling, generally speaking one cannot assume the availability of high-quality and large-scale training data.

2.2. Weakly-supervised Semantic Segmentation

Because of the strong requirement of fully-supervised semantic segmentation, research on finding new techniques to solve weakly semantic segmentation becomes popular. Liu *et al.* worked on dual clustering for semantic segmentation by constructing two clusterings on smoothness and also the relation between image features and superpixel-level labels [11]. Besides the dual clustering method, many other approaches are also proposed to solve weakly supervised semantic segmentation. For example, Vezhnevets et al proposed to use active learning in [23], and multiple instance multi-task learning to solve weakly semantic segmentation in [22]. It may be difficult to learn superpixel-level labels from only one image. In [24], a multi-image model was proposed, which builds a graphical model on the entire dataset. More recently, a graphical model was also proposed in [4], where multiple instance learning and CRF are combined. Besides CRF-based methods, structural information from different superpixels was also considered in [32][33][31], using the concept of graphlets. Recently, semantic relevance has also been studied in the weakly-supervised cases. For example, in [30], hypergraphs were used to capture the high-order semantic relevance, instead of only the second-order relevance in [29], and in [14], deep learning techniques are used to find the pixel-level labeling. In [34], Wei Zhang *et al.* studied one new practical case in which each image is assumed to have part of image-level labels and also

maybe some incorrect labels.

While apparently less stringent than the fully-supervised cases, the image-level labels in existing methods of weakly-supervised semantic segmentation are still assumed to be complete, i.e., the set of labels of a given image captures all possible semantic labels that can be assigned to pixels of that image. As discussed previously, this limitation makes it difficult to utilize vast amount of on-line pictures that would otherwise be useful for the learning task. Our study in this paper is intended to address this issue by considering using information from the entire dataset instead of only one image. We will formally define the problem and present our solution in the next section.

3. Proposed Approach

Based on the previous discussion, we formally define the following problem of this study: Given a set of images with incomplete image-level labels, to predict all pixel-level labels for each image in the set. The image-level labels indicate possible objects in one image, while the pixel-level labels are the final desired segmentation and classification. The incompleteness of labels for an image means that this image may contain some objects/regions which cannot be assigned to any of the given classes in its label set. For example, an image with four objects, *car*, *street*, *sky*, and *grass*, may have only a set of image-level labels, say *car* and *sky*. Still, in the final segmentation, the correct results should properly label those regions corresponding to the missing labels (*street* and *grass*). Apparently, the missing information needs to be figured out by considering the entire set of images. This is schematically illustrated in Figure 2. In this work, we employ the concept of superpixel [15], and assume that pixels within the same superpixel share the same label. This helps simplify the problem to some extent for better tractability.

We use the following notations in the rest of the presentation. Denote one image set with N images by $\mathscr{A} = \{I_i, i \in \{1, \cdots, N\}\}$, which has corresponding partial image-level labels $\mathscr{L} = \{L_i, i \in \{1, \cdots, N\}\}$. Pixels are denoted by $p_{i,j}, j \in \{1, \cdots, M_i\}, i \in \{1, \cdots, N\}$ where $p_{i,j}$ is the j^{th} pixel in the image I_i which has M_i pixels in total. Similarly, superpixels of the image I_i are denoted by $x_i = \{x_{i,j}, j \in \{1, \cdots, n_i\}\}$ where $x_{i,j}$ is the j^{th} superpixel in the image I_i which has n_i superpixels in total. Also we use $L_{i,j}$ to denote the label of the j^{th} superpixel's label in the image I_i.

3.1. Formulating the Problem

In our problem, the input images do not have superpixel-level labels. Further, the images do not have a complete set of semantic labels. Evidently, in general the full information needed for labelling an image needs to be inferred from other images. The multi-image model introduced in [24] may be employed except that complete labelling was assumed therein. Our basic strategy in modeling the problem with incomplete labels is to construct a conditional random field (CRF) for capturing these types of probabilistic associations: visually-similar superpixels are likely to have the same labels (but two similar superpixels may have different likelihoods belonging to the same label, depending on if they are from the same image or from different images), nearby superpixels tend to share labels, and the final label set of an image is a superset of the given (incomplete) label set. Graphically, a basic component of the overall CRF model may be illustrated by Figure 3.

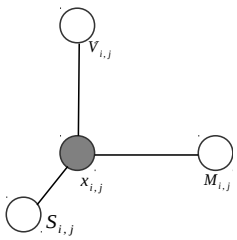

Figure 3. Illustrating the basic component of the proposed CRF model. Each superpixel is related to others via the shown connections. See text for definitions of the symbols. The *entire set of image* forms an overall CRF by combining all the basic components corresponding all superpixels.

In Figure 3, $x_{i,j}$ is the j^{th} superpixel of the image I_i in the dataset. $S_{i,j}$ is the set of spatial neighbors of $x_{i,j}$, defined as the superpixels which are located next to $x_{i,j}$ in the image I_i. $M_{i,j}$ is the set of visually-similar neighbors of $x_{i,j}$, defined as superpixels which are located in those images sharing common image-level labels as I_i. $V_{i,j}$ is the set of visually-similar neighbors of $x_{i,j}$, defined as superpixels which are located in the images without common image-level labels with I_i. To help illustrate how the nodes and connections on the final CRF link the entire image set together, we depict in Figure 4 a visual example with exemplar images and their superpixels explicitly shown.

Based on the structure described above, we can have the complete energy function for our CRF-based model as given in Eqn.1:

$$
\begin{aligned}
E(\{L_{i,j}, & j \in \{1, \cdots, M_i\}, i \in \{1, \cdots, N\}\}, \theta, \alpha) = \\
& \sum_{x_{i,j}, \forall i,j} (\phi(x_{i,j}, L_{i,j}, \theta) + \lambda(L_{i,j}, I_i)) + \\
& \alpha_1 \sum_{(x_{i,j}, x'_{i,j}) \in S_{i,j}, \forall i,j} \varphi(L_{i,j}, L'_{i,j}) + \\
& \alpha_2 \sum_{(x_{i,j}, x'_{i,j}) \in M_{i,j}, \forall i,j} \varphi(L_{i,j}, L'_{i,j}) + \\
& \alpha_3 \sum_{(x_{i,j}, x'_{i,j}) \in V_{i,j}, \forall i,j} \varphi(L_{i,j}, L'_{i,j}) \qquad (1)
\end{aligned}
$$

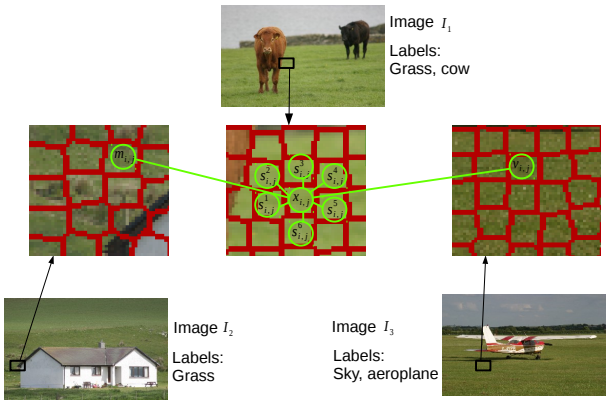

Figure 4. Illustrating basic components of the proposed CRF model with sample images. Shown are some superpixels of three images I_1, I_2, I_3. These superpixels are separated by red boundaries and their positions in their corresponding images are marked by the black rectangles. I_1 and I_2 have one common image-level label, while I_1 and I_3 have no common image-level labels. A basic CRF component is shown in light green color and is built on $x_{i,j}$. Each circle represents one node in CRF. In this example, we only set $M_{i,j} = \{m_{i,j}\}$ and $V_{i,j} = \{v_{i,j}\}$ and their size is one. It is easy to see there are six elements in $S_{i,j}$, which is $\{s^k_{i,j}, k \in \{1, 2, \cdots, 6\}\}$.

where $\alpha = [\alpha_1, \alpha_2, \alpha_3]$ controls the contributions of each potential terms, $\phi(x_{i,j}, L_{i,j}, \theta)$ is the unary potential which gives the energy caused by the fact that the label $L_{i,j}$ is assigned to the superpixel $x_{i,j}$. $\lambda(L_{i,j}, I_i)$ relates to how likely I_i has the label $L_{i,j}$. It can be the negative of the possibility that the image I_i has the label $L_{i,j}$, computed by [5]. For the pairwise potential, we use the Potts model, where the function $\varphi(\cdot)$ is given as Eqn.2.

$$\varphi(L_{i,j}, L^{'}_{i,j}) = \begin{cases} 1 & \text{if } L_{i,j} \neq L^{'}_{i,j} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

3.2. An Inference Algorithm

Exact solutions for achieving the extrema of Eqn.1 would require exponential complexity and thus cannot be obtained unless it is for datasets of trivial complexity. Approximate approaches to inference under similar graphical models have been developed over the years. Examples include Loopy Belief Propagation [12], Graph cut [6], Simulated Annealing [1], and etc. In this work, we adopt Iterated Conditional Modes (ICM) [8] in developing an inference algorithm, owing to its simplicity and in turn efficiency in dealing with a large model like ours. The key idea of the ICM-based algorithm is based on the iterative update: when computing the label of one superpixel, labels of the others are assumed to be fixed. For each superpixel $x_{i,j}$, its label $L_{i,j}$ is computed by (Eqn.3):

$$\begin{aligned} L_{i,j} = \arg\min_{l} \phi(x_{i,j}, l, \theta) + \lambda(l, I_i) + \\ \alpha_1 \sum_{(x_{i,j}, x^{'}_{i,j}) \in S_{i,j}} \varphi(l, L^{'}_{i,j}) + \\ \alpha_2 \sum_{(x_{i,j}, x^{'}_{i,j}) \in M_{i,j}} \varphi(l, L^{'}_{i,j}) + \\ \alpha_3 \sum_{(x_{i,j}, x^{'}_{i,j}) \in V_{i,j}} \varphi(l, L^{'}_{i,j}) \end{aligned} \tag{3}$$

The entire algorithm based on the above core ICM iteration is given in Algorithm 1.

Algorithm 1 An Algorithm Based On ICM

1: Input: Energy function (Eqn.1), one potential label set $\tilde{L}$ of each superpixel $x_{i,j}$
2: Output: the label $L_{i,j}$ of each superpixel $x_{i,j}$, $j \in \{1, \cdots, M_i\}, i \in \{1, \cdots, N\}$
3: BEGIN:
4: initialize each $x_{i,j}$ using random element from $\tilde{L}$ and store initialized labels of each superpixel in Y_1, Y_2.
5: **while** check the stop-condition **do**
6: **for** each superpixel $x_{i,j}$, $j \in \{1, \cdots, M_i\}, i \in \{1, \cdots, N\}$ **do**
7: tmp = $\emptyset$ and Consider $S_{i,j}$, $M_{i,j}$ and $V_{i,j}$ of $x_{i,j}$.
8: **for** each l in L **do**
9: compute the local energy (denoted as e) by assuming each superpixel has the label as that in Y_1 except that $x_{i,j}$ has the label $L_{i,j} = l$
10: tmp = tmp $\cup e$.
11: **end for**
12: Set the label of $x_{i,j}$ in Y_2 as $l^{'}$ which has the smallest local energy.
13: **end for**
14: $Y_1 = Y_2$.
15: **end while**

3.3. Key Implementation Details

We now present a few key technical details that are necessary to fully implement the proposed solution. We use the SLIC algorithm proposed in [2] to obtain superpixels for images in our experiments and also compute the histogram-based features for superpixels and images, following the method of [21]. Before constructing the entire energy function of Eqn.1, we first train one SVM classifier using a very small image set. In this small image set, there are about two images per label and full pixel-level labels of each image are provided. Labeling this subset requires less manual work. More details are shown in Section 4. This pre-trained SVM classifier supplies a measurement for the unary potential in

the proposed model, i.e., the function $\phi(\cdot)$ given in Eqn. 4.

$$\phi(x_{i,j}, L_{i,j}, \theta) = \begin{cases} \rho & \text{if } L_{i,j} \neq L^{'}_{i,j}(\theta) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where $L^{'}_{i,j}(\theta)$ is the predicted label of $x_{i,j}$ by the pre-trained SVM with model parameters θ, and ρ is the penalty.

For the term $\lambda(L_{i,j}, I_i)$, we compute it using the method proposed in [5], which does image-tagging and can provide a ranked list of all possible image-level labels which are likely to be shown in the corresponding image. $\lambda(L_{i,j}, I_i)$ is the negative value of the likelihood that the image I_i has the label $L_{i,j}$.

For pairwise potentials, we need to consider different neighboring relations. For one superpixel $x_{i,j}$, there are three sets of neighbors we need to compute: $S_{i,j}$, $M_{i,j}$ and $V_{i,j}$. For one given superpixel $x_{i,j}$, the spatial neighbor set $S_{i,j}$ can be estimated using image erosion/dilation (note that typically superpixels are irregular in shape). This is illustrated in Figure 5. For the other two sets of neighbors, we can obtain them by Algorithm 2, in which the normalized Euclidean distance is used to compute the similarity between different images and superpixels, based on the image/superpixel features defined above. We emphasize that such neighboring relations are defined based on the proposed CRF model and thus they reflect physical constraints imposed by the given labels (and their interaction) and geometrical proximity, in addition to visual similarity.

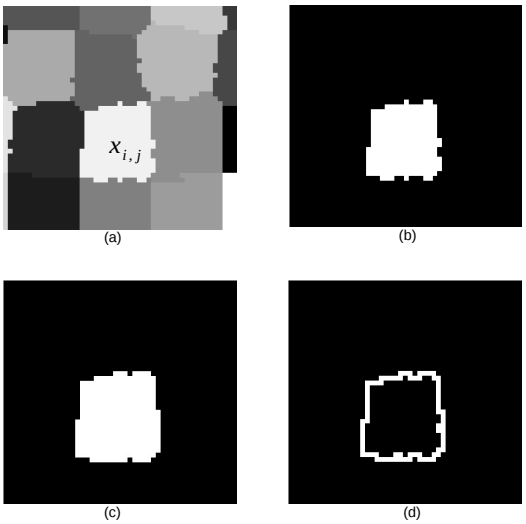

Figure 5. Illustrating how to find the spatial neighbors of one given superpixel $x_{i,j}$ shown in (a). First we need to get the image (b) which is the mask of $x_{i,j}$. Then we can apply the image dilation to (b) to get the image (c). By computing the difference of images (b) and (c), the final mask (d) is obtained. Comparing (d) and the original image (a), we can easily get $S_{i,j}$ which consists of super-pixels which overlap with the final mask (d).

Algorithm 2 Algorithm to compute $M_{i,j}$ and $V_{i,j}$

Input: $\{I_i, L_i\}$, $\{x_{i,j}\}$, $j \in \{1, \cdots, M_i\}$, $i \in \{1, \cdots, N\}$, $D_1(\cdot)$ which is the function to compute the distance between two images and $D_2(\cdot)$ which is to compute the distance between two superpixels.
Output: $M_{i,j}, V_{i,j}, j \in \{1, \cdots, M_i\}, i \in \{1, \cdots, N\}$

BEGIN:
// To compute SM_i, SV_i.
for i = 1,$\cdots$, N **do**
  for j = 1, $\cdots$, N, $i \neq j$ and $L_i \cap L_j \neq \emptyset$ **do**
    Compute the similarity $D_1(I_i, I_j)$.
  end for
  Find the top q most similar images, denoted as SM_i.
  for j = 1, $\cdots$, N, $i \neq j$ and $L_i \cap L_j == \emptyset$ **do**
    Compute the similarity $D_1(I_i, I_j)$.
  end for
  Find the top q most similar images to I_i, denoted as SV_i.
end for

for each superpixel $x_{i,j}$, $j \in \{1, \cdots, M_i\}$, $i \in \{1, \cdots, N\}$ **do**
  // we have SM_i and SV_i of I_i
  // and will construct $\text{SPM}_{i,j}$ and $\text{SPV}_{i,j}$
  $\text{SPM}_{i,j} = \emptyset$, $\text{MSS}_{i,j} = \emptyset, \forall i, j$.
  for each superpixel $x^{'}_{i,j}$ in each image $I' \in \text{SM}_i$ **do**
    Find the top p most similar superpixels to $x_{i,j}$ based on $D_2(x_{i,j}, x^{'}_{i,j})$
    Denote these p superpixels as $\text{MSS}_{i,j}$ and also we set $\text{SPM}_{i,j} = \text{SPM}_{i,j} \cup \text{MSS}_{i,j}$
  end for
  Find top k most similar superpixels to $x_{i,j}$ from $\text{SPM}_{i,j}$, which are $M_{i,j}$ of $x_{i,j}$.

  $\text{SPV}_{i,j} = \emptyset$, $\text{MSS}_{i,j} = \emptyset, \forall i, j$.
  for each superpixel $x^{'}_{i,j}$ in each image $I' \in \text{SV}_i$ **do**
    Find the top p most similar superpixels to $x_{i,j}$ based on $D_2(x_{i,j}, x^{'}_{i,j})$
    Denote these p superpixels as $\text{MSS}_{i,j}$ and $\text{SPV}_{i,j} = \text{SPV}_{i,j} \cup \text{MSS}_{i,j}$
  end for
  Find top k most similar superpixels to $x_{i,j}$ from $\text{SPV}_{i,j}$, which are $V_{i,j}$ of $x_{i,j}$
end for

3.4. Comparison With MIM

The proposed method bears some similarity to the Multi-Image Model (MIM) of [24], since both consider a set of images simultaneously. To appreciate the key difference easily, we provide the energy function of the MIM below

(Eqn.5):

$$E(\{L_{i,j}, j \in \{1, \cdots, M_i\}, i \in \{1, \cdots, N\}\}, \theta) = \sum_{x_{i,j}, \forall i,j} (\psi_1(x_{i,j}, L_{i,j}, \theta) + \pi(L_{i,j}, I_i)) + \sum_{(x_{i,j}, x_{i,j}^{'}) \in S_{i,j}, \forall i,j} \varphi_1(L_{i,j}, L_{i,j}^{'}, x_{i,j}, x_{i,j}^{'}) + \sum_{(x_{i,j}, x_{i,j}^{'}) \in M_{i,j}, \forall i,j} \varphi_1(L_{i,j}, L_{i,j}^{'}, x_{i,j}, x_{i,j}^{'}) \quad (5)$$

where $\pi(L_{i,j}, I_i)$ is zero if the label $L_{i,j}$ is one image-level label of the image I_i and it is set to infinity otherwise. Moreover, $\varphi_1(\cdot)$ is given as follows:

$$\varphi_1(L_{i,j}, L_{i,j}^{'}, x_{i,j}, x_{i,j}^{'}) = \begin{cases} 1 - D(x_{i,j}, x_{i,j}^{'}) & \text{if } x_{i,j}, x_{i,j}^{'} \text{are different} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where $D(\cdot)$ is one similarity metric.

Eqn.5 clearly indicates one strong requirement on the labels, imposed by the choice of $\pi(\cdot)$. Because of that function, MIM cannot be used to solve the general problem defined in this paper. In our formulation, to solve the more general and practical problem, we relaxed the strong requirement in MIM by introducing a new $\pi(\cdot)$ function *plus* one additional pairwise potential to better capture visual similarity of superfixels (those across images and do not have common image-level labels). These resulted in the new model of Eqn.1. In fact, compared with both formulations, we can see that MIM is one special case of our approach, which is used to deal with the less challenging situation where images have completely image-level labels.

4. Experiments

In this section, we demonstrate the effectiveness of the proposed approach based on comparative experiments using the following three datasets: one synthetic dataset, the MSRC-21 dataset [19] and the Siftflow dataset [21]. For the synthetic dataset and the MSRC-21 dataset, we make comparison with the approach in [24], which is among the state-of-art methods in the literature. For the Siftflow dataset, we provide our experimental results and compare with existing approaches in the fully-supervised case and the ordinary weakly-supervised case. The comparison is based on two metrics: per-pixel accuracy (denoted as *pp* and shown in Eqn.7) and average per-class accuracy (denoted as $\bar{pc}$ and shown in Eqn.9). To compute these measures, we need the size of each superpixel $x_{i,j}$, which is denoted by $size(x_{i,j})$.

$$pp = \frac{\sum_{i,j} \delta(L_{i,j} - L_{i,j}^{'}) size(x_{i,j})}{\sum_{i,j} size(x_{i,j})} \quad (7)$$

$$pc_l = \frac{\sum_{i,j} \delta(L_{i,j} - l)\delta(L_{i,j} - L_{i,j}^{'}) size(x_{i,j})}{\sum_{i,j} \delta(L_{i,j} - l) size(x_{i,j})} \quad (8)$$

$$\bar{pc} = \frac{1}{|\bigcup L_i|} \sum_l \text{pc}_l \quad (9)$$

In the above definitions, $L_{i,j}^{'}$ is the predicted label and $L_{i,j}$ is the ground truth of the label of $x_{i,j}$, and pc_l is the pixel-level accuracy for all the pixels whose label is l. Also $|\bigcup L_i|$ is the total number of potential labels .

4.1. Synthetic Dataset

The simulation is designed as follows. First, we generate one synthetic dataset that has 30 pairs of observation images and labelmaps. An observation image is a 200×200 gray-scale image while its labelmap is a 200×200 image whose pixel values are the labels of its corresponding observation. For each observation image, we split it into 20×20 superpixels, each of which has 10×10 pixels. Moreover, we assume that all pixels in one superpixel have the same label and labels are from this set: $\{1, 2, 3, 4, 5\}$.

To generate each pair of one observation image and its labelmap, we run the following procedure:

1. We first generate one labelmap randomly and make sure that labels of pixels in the same superpixel are the same.
2. The corresponding observation image is generated based on the new labelmap.
3. The inference algorithm runs for 200 iterations to obtain the final pair of observation image and labelmap.
   (a) For each iteration, we use the current labelmap and the observation image to generate a better labelmap whose energy is smaller. Then based on the new generated labelmap, we generate the new observation image.

During the above procedure, we set the total number of iterations to be 200 since at this iteration the observation-labelmap pair is already stable. Besides the number of iterations, we set the relationship between one observation image and its labelmap as the Gaussian distribution whose standard variation is set to be 10. Samples of the constructed dataset are shown in Figure 6. The average size of the complete image-level labels is 3.46. To generate partial image-level labels, we randomly remove one label from the complete image-level labels. The parameters k, q and p we set in this simulation are 21, 3, 5, respectively.

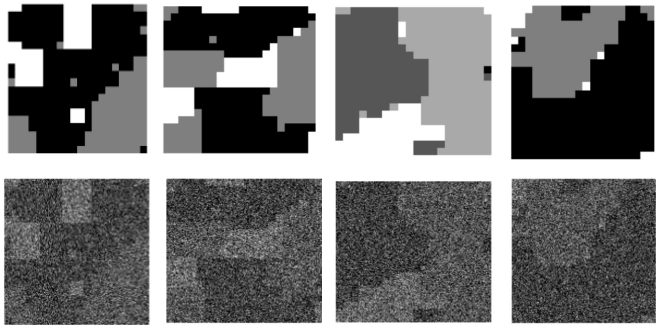

Figure 6. This figure shows some pairs of the observation and the labelmap generated in the synthetic dataset. The first row consists of labelmaps while the second one consists of observation images. For each column, it is a pair of one labelmap and its observation.

The synthetic dataset was then used to compare the performance of the proposed approach and the MIM method. The MIM method would simply assume whatever labels given for an image is complete. The final results are summarized in Table 1. From these results, it is obvious that the MIM method lags the proposed approach by a large margin. We also note the difficulty of the task (even if the dataset is synthetic), since a lot of source of uncertainties were introduced in the process of creating the data. This explains why the overall accuracy numbers are not very high for either approach.

	pp	$\bar{\text{pc}}$
MIM [24]	51.15%	29.81%
Proposed	76.74%	42.72%

Table 1. Comparing with the MIM model on the synthetic dataset.

4.2. MSRC-21 Dataset

In this dataset, there are 591 images and 21 objects [1] in total. We split the dataset into two parts: Set one and Set two, both are the same as those used in [19]. As a result, there are 276 images in Set one, 256 in Set two. Also we call the union of Set one and Set two as the Entire Set. To get the pre-trained SVM classifier, we randomly choose 42 images out of 59 images which consist of the validation set as in [19]. The average numbers of the complete image-level labels for Set one, Set two and the Entire Set are 2.4710, 2.4492 and 2.4605, respectively. To generate partial image-level labels, we randomly remove one label from each complete image-level label set. So the average sizes of Set one, Set two and Entire Set decrease by 40.4 %, 40.8% and 40.6%, respectively. In this experiment, parameters k, p and q are set to be 10, 3 and 8, respectively.

[1]There are 23 objects in total, but 2 of them are not considered by Microsoft research. So we only use 21 objects. Details are shown in the dataset which is available on Microsoft research.

The per-class accuracies from the proposed and the MIM method for Set one, Set two, and the Entire Set are plotted respectively in Figure 7, Figure 8 and Figure 9. Overall, the performance gains of the proposed method over MIM are 5%, 3% and 2% respectively for Set one, Set two, and the Entire Set.

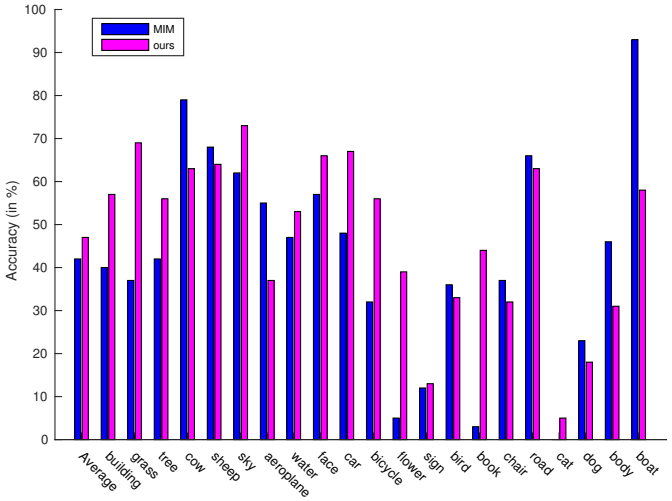

Figure 7. Comparison of per-class accuracies for Set one. The first column is the average performance of two algorithms. The left 21 columns are for each object.

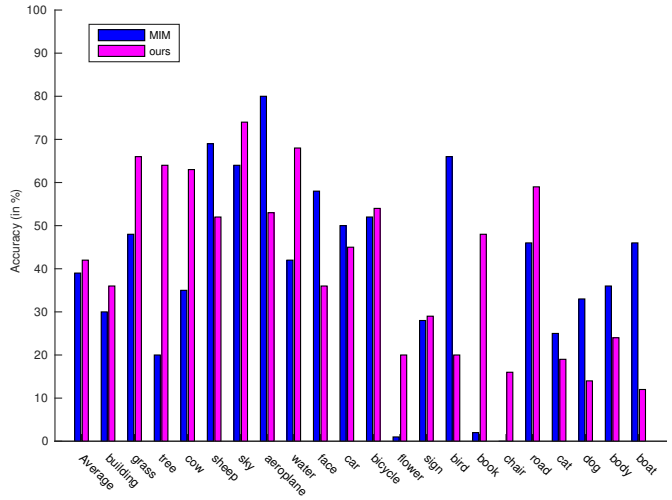

Figure 8. Comparison of per-class accuracies for Set two. The first column is the average performance of two algorithms. The left 21 columns are for each object.

In addition to per-class accuracy, we also provide the per-pixel accuracy in Table 2, where it is clear that the proposed approach was able to outperform MIM by large margins on all the sets of data.

The above results demonstrated the effectiveness of the proposed approach in dealing with incomplete image-level

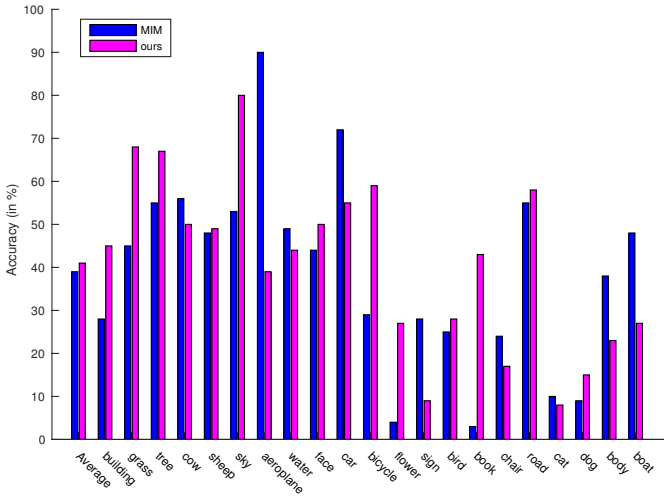

Figure 9. Comparison of per-class accuracies for the Entire Set. The first column is the average performance of two algorithms. The left 21 columns are for each object.

	Set one	Set two	Entire Set
MIM in [24]	43.33%	39.44%	41.82%
Proposed	56.69%	52.80%	53.08%

Table 2. The per-pixel accuracies *pp* of our approach and MIM in [24].

labels. It is worth pointing out that the MIM method reported higher performance numbers in [24], where it was studied as an ordinary weakly-supervised approach with complete image-level labels for training. Our experimental setting is more realistic for simulating the scenario of learning with Web images. In this experiment, considering the dropped label per image, the label set suffers a loss of around 40% labeling information compared with the case where images have complete image-level labels. The proposed approach, even if with only a very simple ICM-based inference algorithm, was shown to be able to better deal with the incomplete label data.

4.3. Siftflow Dataset

In this experiment, we show the performance of our algorithm on the Siftflow dataset [21]. This dataset consists of 2688 images and 33 labels. We use the entire training set which has 2488 images, as defined in [17]. The average number of image-level labels for each image in the entire Siftflow dataset is 4.4297 and for the part we use, on average, there are 4.3881 labels per image. To simulate incomplete image-level labeling, we create partial image-level labels for each image by randomly removing one label from the original label set. This means we remove 22.79% label information on average for each image. During the experiment, parameters k, p and q are set to be 10, 3 and 8, respectively. Our results are: $pp = 57.09\%$ and $\bar{pc} = 22.34\%$. Since the related work do not report the per-pixel accuracy (*pp*) on this dataset, we only report the per-class accuracy (by quoting) in Table 3, including the results from some fully-supervised methods ([19][10]) and weakly-supervised methods assuming complete image-level labels ([24][25][11][31]). From the table, we see that our approach was able to deliver nearly comparable performance, although we subject our approach to the heavy loss of information, while the competing methods either utilize pixel-level labels or assume and use complete image-level labels.

[24]	[25]	[19]	[10]	[11]	[31]	Ours
14%	21%	24%	24%	26%	27.73%	22.34%

Table 3. Average per-class accuracy $\bar{pc}$ from our approach and those from a set of competing approaches, either fully-supervised or weakly-supervised with complete image-level labels. The results above are in percentage.

5. Conclusion & Future Work

We identified a key limitation in existing methods for semantic segmentation and proposed a new multi-image formulation for addressing the limitation. An inference algorithm was designed for finding a solution under the proposed multi-image model. To demonstrate the effectiveness of our algorithm, we performed experiments on both synthetic data and real datasets including MSRC-21 and Siftflow. While current results have shown advantages of the proposed method, there are still a few leads for future exploration. In particular, current results indict that some classes have low per-class accuracy, possibly due to their rare presence in the images. Such information (some classes being rare), if known *a priori*, may be explicitly factored into the formulation so that rare classes do not get overshadowed by other more common classes.

Acknowledgments

The work was supported in part by an ARO grant (#W911NF1410371) and an ONR grant (#N00014-15-1-2344). Any opinions expressed in this material are those of the authors and do not necessarily reflect the views of ARO or ONR.

References

[1] AARTS/KORST. *Simulated annealing and boltzmann machines. A stochastic approach to combinatorial optimization and neural computing*. John Wiley., 1990.

[2] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(11):2274–2282, 2012.

[3] P. Arbeláez, B. Hariharan, C. Gu, S. Gupta, L. Bourdev, and J. Malik. Semantic segmentation using regions and parts. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3378–3385. IEEE, 2012.

[4] F.-J. Chang, Y.-Y. Lin, and K.-J. Hsu. Multiple structured-instance learning for semantic segmentation with uncertain training data. In *Computer Vision and Pattern Recognition, 2014 IEEE Conference on*. IEEE, 2014.

[5] M. Chen, A. Zheng, and K. Weinberger. Fast image tagging. In *Proceedings of the 30th International Conference on Machine Learning*, pages 1274–1282, 2013.

[6] A. Delong, A. Osokin, H. N. Isack, and Y. Boykov. Fast approximate energy minimization with label costs. *International journal of computer vision*, 96(1):1–27, 2012.

[7] X. He, X. Li, G. Yang, J. Xu, and Q. Jin. Adaptive tag selection for image annotation. In *Advances in Multimedia Information Processing–PCM 2014*. Springer, 2014.

[8] J. Kittler and J. Föglein. Contextual classification of multispectral pixel data. *Image and Vision Computing*, 2(1), 1984.

[9] P. Kohli, P. H. Torr, et al. Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, 82(3):302–324, 2009.

[10] C. Liu, J. Yuen, and A. Torralba. Nonparametric scene parsing: Label transfer via dense scene alignment. In *Computer Vision and Pattern Recognition, 2009 IEEE Conference on*, pages 1972–1979. IEEE, 2009.

[11] Y. Liu, J. Liu, Z. Li, J. Tang, and H. Lu. Weakly-supervised dual clustering for image semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2075–2082. IEEE, 2013.

[12] K. P. Murphy, Y. Weiss, and M. I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the 5th conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1999.

[13] H. Myeong and K. M. Lee. Tensor-based high-order semantic relation transfer for semantic scene segmentation. In *Computer Vision and Pattern Recognition, 2013 IEEE Conference on*, pages 3073–3080. IEEE, 2013.

[14] P. O. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. In *CVPR*, pages 1713–1721, 2015.

[15] X. Ren and J. Malik. Learning a classification model for segmentation. In *Computer Vision. Proceedings. Ninth IEEE International Conference on*, pages 10–17. IEEE, 2003.

[16] G. Roig, X. Boix, R. D. Nijs, S. Ramos, K. Kuhnlenz, and L. V. Gool. Active map inference in crfs for efficient semantic segmentation. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2312–2319. IEEE, 2013.

[17] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. *IJCV*, 77(1-3):157–173, 2008.

[18] A. Sharma, O. Tuzel, and D. W. Jacobs. Deep hierarchical parsing for semantic segmentation. *arXiv preprint arXiv:1503.02725*, 2015.

[19] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *International Journal of Computer Vision*, 81(1), 2009.

[20] J. Tighe and S. Lazebnik. Finding things: Image parsing with regions and per-exemplar detectors. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3001–3008. IEEE, 2013.

[21] J. Tighe and S. Lazebnik. Superparsing. *International Journal of Computer Vision*, 101(2):329–349, 2013.

[22] A. Vezhnevets and J. M. Buhmann. Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning. In *Computer Vision and Pattern Recognition, 2010 IEEE Conference on*, pages 3249–3256. IEEE, 2010.

[23] A. Vezhnevets, J. M. Buhmann, and V. Ferrari. Active learning for semantic segmentation with expected change. In *Computer Vision and Pattern Recognition, 2012 IEEE Conference on*, pages 3162–3169. IEEE, 2012.

[24] A. Vezhnevets, V. Ferrari, and J. M. Buhmann. Weakly supervised semantic segmentation with a multi-image model. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 643–650. IEEE, 2011.

[25] A. Vezhnevets, V. Ferrari, and J. M. Buhmann. Weakly supervised structured output learning for semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 845–852. IEEE, 2012.

[26] W. Voravuthikunchai, B. Crémilleux, and F. Jurie. Histograms of pattern sets for image classification and object recognition. In *Computer Vision and Pattern Recognition, 2014 IEEE Conference on*. IEEE, 2014.

[27] Y. Wang, S. Wang, J. Tang, H. Liu, and B. Li. Unsupervised sentiment analysis for social media images. In *24th International Joint Conference on Artificial Intelligence*, 2015.

[28] Z. Wang and B. Li. Learning to recommend tags for online photos. In *Social Computing and Behavioral Modeling*, pages 1–9. Springer, 2009.

[29] W. Xie, Y. Peng, and J. Xiao. Semantic graph construction for weakly supervised image parsing. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[30] W. Xie, Y. Peng, and J. Xiao. Weakly-supervised image parsing via constructing semantic graphs and hypergraphs. In *Proceedings of the ACM International Conference on Multimedia*, pages 277–286. ACM, 2014.

[31] L. Zhang, Y. Gao, Y. Xia, K. Lu, J. Shen, and R. Ji. Representative discovery of structure cues for weakly-supervised image segmentation. *Multimedia, IEEE Transactions on*, 2014.

[32] L. Zhang, M. Song, Z. Liu, X. Liu, J. Bu, and C. Chen. Probabilistic graphlet cut: Exploiting spatial structure cue for weakly supervised image segmentation. In *Computer Vision and Pattern Recognition, 2013 IEEE Conference on*, pages 1908–1915. IEEE, 2013.

[33] L. Zhang, Y. Yang, Y. Gao, Y. Yu, C. Wang, and X. Li. A probabilistic associative model for segmenting weakly-supervised images. 2014.

[34] W. Zhang, S. Zeng, D. Wang, and X. Xue. Weakly supervised semantic segmentation for social images. In *CVPR*, pages 2718–2726, 2015.

[35] Y. Zhang, J. Wu, and J. Cai. Compact representation for image classification: To choose or to compress? In *CVPR 2014 IEEE Conference on*, pages 907–914. IEEE, 2014.