CVPR 2013

Graph Transduction Learning with Connectivity Constraints with Application to Multiple Foreground Cosegmentation

Tianyang Ma
Temple University
tianyang.ma@temple.edu

Longin Jan Latecki
Temple University
latecki@temple.edu

Abstract

The proposed approach is based on standard graph transduction, semi-supervised learning (SSL) framework. Its key novelty is the integration of global connectivity constraints into this framework. Although connectivity leads to higher order constraints and their number is an exponential, finding the most violated connectivity constraint can be done efficiently in polynomial time. Moreover, each such constraint can be represented as a linear inequality. Based on this fact, we design a cutting-plane algorithm to solve the integrated problem. It iterates between solving a convex quadratic problem of label propagation with linear inequality constraints, and finding the most violated constraint. We demonstrate the benefits of the proposed approach on a realistic and very challenging problem of cosegmentation of multiple foreground objects in photo collections in which the foreground objects are not present in all photos. The obtained results not only demonstrate performance boost induced by the connectivity constraints, but also show a significant improvement over the state-of-the-art methods.

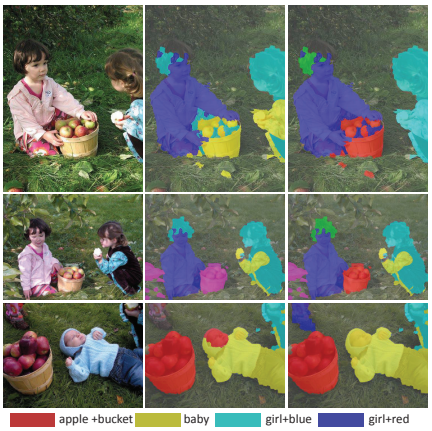

Figure 1. Multiple Foreground Cosegmentation results on three images of the scene *Apple+picking*. First Columns: original images. Second Columns: the results of an excellent graph transduction SSL method RLGC [24]. Third Column: results of the proposed GTC. Compared to RLGC, GTC improves the consistency of label assignment by enforcing connectivity of regions with the same label.

1. Introduction

Given multiple images sharing overlapping contents, the goal of image cosegmentation is to simultaneously divide these images into non-overlapping regions of foreground and background. In an unsupervised setting, foreground is defined as the common regions that repeatedly occur across the input images [15]. In an interactive or supervised setting [1], some foreground objects are explicitly assigned by an user as the regions of interest.

Kim and Xing [12] has recently proposed a multiple foreground cosegmentation (MFC) task, in which K different foreground objects need to be jointly segmented from a group of M input images. This scenario is very realistic, since not all objects need to appear in each image, i.e., each of images contains a different and *unknown* subset of the K objects. Three example images from the same group are shown in the first column of Fig. 1. This task contrasts the classical cosegmentation problem dealt with by most existing algorithms [7, 1, 15, 10, 13, 21, 22], where a much simpler and less realistic setting is usually assumed by requiring that the same set of objects occurs in every image. While this assumption provides a relatively strong prior which has been utilized by most of cosegmentation algorithms, it severely limits the application scope of these cosegmentation algorithms, since it is not valid for most real photo collections.

The fact that the MFC problem does not assume that each objects appears in every image, brings serious challenges to the cosegementation algorithms, which is addressed [12]. There are two iterative steps, foreground modeling and region assignment. The region assignment subproblem is solved by assuming foreground model is given. The authors of [12] consider two settings: supervised and unsupervised.

In the supervised setting, it is straight forward that foreground model can be built through objects labeled by users in the training images. In the unsupervised setting, foreground model can be initialized by running unsupervised cosegmentation method [13, 9]. As clearly demonstrated in [12], the segmentation results in the supervised setting are significantly better. Their supervised setting is still very realistic from the point of view of real applications, since only a very small number of objects of interest must be marked by the user. Only 20% of images is used from groups of images containing 10 to 20 images. For example, this means that the user only needs to mark the objects in 2 out of 10 images. Since this supervised setting contains a very small number of training data, which is very challenging for supervised learning methods.

Our contribution is based on the observation that this is an ideal setting for semi-supervised learning (SSL). In particular, we formulate this problem as graph transduction SSL, which has demonstrated impressive results on many tasks, especially when there exists only a small amount of labeled data samples. Compared to supervised methods, its main advantage relies on using both labeled and unlabeled data during the training process, which yields considerable improvement in labeling accuracy, e.g., [27, 28, 24].

However, the label propagation accuracy in graph transduction SSL highly depends on how reliable the similarity of graph nodes is. Since in the MFC application, the nodes represent image regions (segments or superpixels), their similarity is neither very discriminative nor particularly stable. In particular, due to large appearance variations of the same objects in different images, segments belonging to different objects may accidently have higher similarity than segments belonging to the same object.

To address this problem, we propose to constrain graph transduction SSL framework by integrating global connectivity constraints. In other words, we enforce that segments assigned the same label form connected regions in each image. Connectivity is naturally motivated by the human visual perception, and connectedness is a very intuitive and effective criterium for object segmentation, as has been demonstrated in [20, 14] in the context of supervised image segmentation.

As in [12], for a given set of images containing common objects, we first perform over-segmentation to obtain several segments for each image separately. While [12] uses a spatial pyramid as the objects model, we only utilize color-SIFT and use bag-of-word (BoW) model to represent segments. Although using BoW enjoys some robustness to the object variations, such as changes in shape and orientation, it also makes the similarity between segments not very discriminative, which in turn significantly degrades the labeling results of SSL methods. To demonstrate this, we examine segmentation results by labeling in Fig. 1. The second column shows the results of an SSL excellent method introduced in [24]. We call it regularized local and global consistency (RLGC). We can see that many disconnected regions are wrongly assigned the same labels because of their similar color and texture, for example, the face of baby and apple basket. This happens because in standard graph transduction SSL framework, each segment is taken out-of-context and labeled independently. While this is suitable for general SSL inference problem, it is clearly suboptimal in our application. In particular, while the segment graph encodes the visual similarity between pairs of segments, the spatial information between segments in the same image is totally neglected. This information is expressed as connectivity in the proposed framework.

In our graph-based formulation, if nodes representing segments from the same image share the same class label, they must form a connected subgraph [11]. This is a global property and it introduces high-order constraints. As shown in [14], although it is an exponential problem (with respect to the number of nodes) to examine if two nodes are connected, finding the most violated connectivity constraint can be done efficiently in polynomial time. Moreover, each such constraint can be represented as a linear inequality.

To solve a SSL problem formulated with connectivity constraints in graph transduction formulation, we design a cutting-plane algorithm, in which we iterate between solving a convex problem of label propagation with linear inequality constraints, and finding the most violated constraint. We investigate two versions of our method.

The output of most graph transduction SSL methods, e.g. [27, 24], represents the confidence of assigning data points to all labels. The discretization step is then performed on each unlabeled data point independently, by simply assigning the label with the largest confidence. The first version of our method enforce the connectivity constraints at the final discretization step of label confidences obtained through SSL learning. This can be considered as a postprocessing method, and could be applied to any SSL method. It can be solved as linear programming with linear inequality constraints.

More importantly, in the second version, we integrate the graph transduction formulation with connectivity constraints, and solve it as a convex quadratic programming with linear inequality constraints. We call this method graph transduction with connectivity constraints (GTC). Its segmentation examples are shown in the third column of Fig. 1. As can be seen it significantly improves on label assignment of RLGC (second column). In particular, the baby face belongs to the baby not to the basket anymore. It even can correct wrong labels as can be seen in the first row, where the basket is wrongly labeled as baby by RLGC, which is corrected by GTC. We have a similar case for the basket in the second row. This examples as well as our ex-

perimental results in Section 6 clearly demonstrate that the connectivity information can be used to increase the robustness of SSL methods.

We evaluate the proposed approach on real world MFC application on FlickrMFC dataset. It significantly outperforms the MFC method in [12] and other state-of-the-art cosegmentation methods.

The remainder of this paper is organized as follow: The related work is introduced in Section 2. In Section 3, we revisit the standard graph transduction SSL framework. In Sections 4 and 5, we introduce the proposed integration of connectivity constraints into the graph transduction framework, and derive a method to solve it efficiently.

2. Related Work

Many approaches have been proposed to solve the image cosegmentation problem [7, 1, 15, 10, 13, 21, 22]. All these approaches only consider two class (forground/background) cosegmentation problem. The initial model presented in [15] provides a framework to enforce consistency among two foreground histograms in addition to the Marov Random Field (MRF) segmentation terms for each image. In [10], a discriminative clustering formulation is adopted, in which the goal is to assign foreground/background labels jointly to all images so that a supervised classifier trained with these labels leads to maximal separation of the two classes. Recently, a Random Walker based method is proposed in [4], and is shown to be an effective framework for cosegmentation problem complementary to MRF formulation. While our method shares similar properties as [10] and [4], in the sense that we also have a graph formulation and utilize the normalized graph Laplacian, we have a very different goal for constructing the graph, consequently, the definitions of nodes and edges in the graph are also very different. In particular, for both [10] and [4], image pixels are taken as nodes, and edges only exist locally between pairs of nearby pixels. This follows the standard framework of spectral clustering for image segmentation [17]. In our method, the graph is constructed using segments as nodes, and the edges exist between every pair of segments, because the graph is used for the purpose of propagating the labels from labeled segments to unlabeled segments following the graph transduction SSL framework.

Semi-supervised learning is the intermediate range of the spectrum between supervised methods and unsupervised methods. It has been widely used to solve many kinds of machine learning and computer vision problems. In [26], Zeisl et al. combined SSL with multiple instance learning to solve the object tracking problem. Fergus et al. [5] introduced a linear SSL method to label tiny images among a gigantic image collections. In [6], SSL method is used to associate keywords (side information) of labeled and unlabeled images, so that a stronger classifier can be obtained for the image classification task. A SSL based hashing method is proposed in [25] for image retrieval. Recently, SSL is used in [18] for solving scene categorization task, where constraints based on mutual exclusion and comparative attributes are imposed. In [23], SSL has been applied to improve the affinity metric for single image segmentation. Our approach is very different from these SSL applications to computer vision problems. To our best knowledge, this is for the first time that connectivity constraints are considered in the SSL framework.

3. Semi-supervised Learning (SSL)

In this section, we will first introduce how do we construct the segment graph in Sec 3.1 And in Sec 3.2, we will review how to use the graph transduction method to solve a standard semi-supervised learning problem. Finally, in Sec 5, we focus on how to impose the connectivity constraints under semi-supervised learning framework and how to solve it efficiently.

3.1. Segment Graph Construction

Given a set of images which contain multiple common objects, we first divide each image I_m into segments (or superpixels) $S_m = \{s_m^1, \ldots, s_m^K\}$. Set V be the set of the segments in all images. Any segmentation algorithm can be used here. We used submodular image segmentation method introduced in [13]. We assume that segments in a small number of images are labeled with object categories. We are given a small set of labeled segments, and a large majority of unlabeled segments. Our goal is to infer a label for each unlabeled segment.

We define a weighted graph $G = (V, \mathbf{W})$, where is a nonnegative matrix representing the pairwise similarity of image segments, which is defined as follows. For each segment s_i, we compute its ColorSIFT descriptor [19] and quantize them according to a codebook. Then a bag-of-words histogram $\mathbf{x}_i$ is used to represent segment s_i. For two nodes i and j representing two different segments s_i and s_j, the weight w_{ij} is computed using a RBF kernel:

$$w_{ij} = \exp - \frac{d(\mathbf{x}_i, \mathbf{x}_j)}{2\sigma^2} \tag{1}$$

where $d(\mathbf{x}_i, \mathbf{x}_j)$ computes the $\mathcal{X}^2$ distance between $\mathbf{x}_i$ and $\mathbf{x}_j$, and σ is the kernel bandwidth parameter. We follow [3] to compute σ. In particular, $\sigma = \bar{dist}_k/3$, where $\bar{dist}_k$ is the average distance between each sample and its kth nearest neighbor. Since sparsity is important to remove noise and it has been proved that semi-supervised learning algorithms are more robust when run on a sparse graphs [8], we set $w_{ij} = 0$, if $i \notin k\mathrm{NN(j)}$, where kNN denotes the set of k nearest neighbors (k is the same as used in computing σ).

3.2. Graph Transduction for SSL

We assign the class labels to unlabeled image segments in a standard graph-based semi-supervised learning framework, which we review here. Let the node degree matrix $\mathbf{D} = diag([d_1, \cdots, d_N])$ be defined as $d_i = \sum_{j=1}^{N} w_{ij}$, where $N = |V|$. The binary label matrix $\mathbf{Y} \in \{0,1\}^{N \times C}$ is defined as $y_{il} = 1$ if node s_i has label $l \in L$ and $y_{il} = 0$ otherwise, where C is the number of labels in L. We also assume that $\sum_l y_{il} \leq 1$ for every node i meaning that each node can have at most one class label. The normalized graph Laplacian is defined as $\mathbf{L} = \mathbf{D}^{-1/2}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-1/2}$.

Graph-based semi-supervised learning methods propagate label information from labeled nodes to unlabeled nodes [28]. Most methods define a continuous variable $\mathbf{F} \in \mathbb{R}^{N \times C}$ that is estimated on the graph to minimize a cost function. The cost function typically used has two tradeoff terms. One term is used to measure the smoothness of the function on the graph of both labeled and unlabeled data, with the second term used to measure the fitness between $\mathbf{F}$ and the label information for the labeled nodes. In particular, we follow the formulation introduced in [24]. We call the method regularized local and global consistency (RLGC), since it modifies the cost function from the classic local and global consistency (LGC) method [27] by adding a node regularizer $\mathbf{R}$:

$$\mathcal{Q}(\mathbf{F}) = \text{tr}\{\mathbf{F}^T\mathbf{L}\mathbf{F} + \mu(\mathbf{F} - \mathbf{R}\mathbf{Y}^T)(\mathbf{F} - \mathbf{R}\mathbf{Y}^T)\}, \quad (2)$$

where μ is a constant. The matrix $\mathbf{R}$ is used to balance the influence of labels from different classes. It works as node regularizer that normalizes labels within each class based on node degrees. This is very important for the problems with highly unbalanced labeled nodes, which is the case for our application. $\mathbf{R} = diag(\mathbf{r})$ in which $\mathbf{r} = [r_1, \ldots, r_N]$ is computed as

$$r_i = \begin{cases} \frac{1}{C} \cdot \frac{d_i}{\sum_k y_{kl} d_k} & \text{if } \exists_{l \in L}\ y_{il} = 1 \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

Due to the convexity of the cost function in (2), we obtain a closed form solution by zeroing the partial derivative $\frac{\partial \mathcal{Q}}{\partial \mathbf{F}} = 0$. With simple algebra, we can derive

$$\mathbf{F}^* = (\frac{\mathbf{L}}{\mu} + \mathbf{I})^{-1}\mathbf{R}\mathbf{Y} = \mathbf{P}\mathbf{R}\mathbf{Y} \quad (4)$$

where $\mathbf{P} = (\frac{\mathbf{L}}{\mu} + \mathbf{I})^{-1}$ as the propagation matrix [27].

After obtaining the continuous solution $\mathbf{F}^* \in \mathbb{R}^{N \times C}$, we need to binarize it into $\mathbf{Y}^* \in \{0,1\}^{N \times C}$. As is usually the case in graph transduction SSL, this is a simple argmax step: for every node i determine $l^* = \arg\max_l \mathbf{F}^*_{il}$, and then set $\mathbf{Y}^*_{il} = 1$ if $l = l^*$ and $\mathbf{Y}^*_{il} = 0$ if $l \neq l^*$.

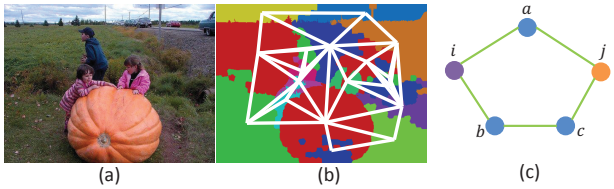

Figure 2. (a) Original image (b) Segments and adjacent graph (c) A simple adjacency graph. For a pair of nodes (i, j), there are three vertex-separator sets {a, b}, {a, c} and {a, b, c}. Only {a, b} and {a, c} are essential vertex-separator sets.

4. Constrained SSL

According to the cost function defined in (2), to solve the SSL problem, we need to solve a QP problem defined on continuous variable $\mathbf{F} \in \mathbb{R}^{N \times C}$. In this section we extend this problem by adding linear constraints to enforce connectivity.

Let $\mathcal{C}$ denotes a set of matrices $\mathbf{M} \in \{-1, 0, 1\}^{C \times N}$ representing linear constraints. We consider the following constrained formulation of Eq. (2):

$$\begin{aligned} \mathcal{Q}(\mathbf{F}) \quad &= \quad \text{tr}\{\mathbf{F}^T\mathbf{L}\mathbf{F} + \mu(\mathbf{F} - \mathbf{R}\mathbf{Y}^T)(\mathbf{F} - \mathbf{R}\mathbf{Y}^T)\} \\ s.t. \quad &\text{tr}(\mathbf{M}\mathbf{F}) \leq 1, \ \ \forall\, \mathbf{M} \in \mathcal{C}. \end{aligned} \quad (5)$$

With an empty constraints set $\mathcal{C}$, minimizing (5) is equivalent to minimizing (2). Hence it is a convex QP problem and it has a closed form solution $\mathbf{F}$ as shown (4). With a non-empty set of linear constraints, convexity still holds. Although the closed form solution cannot be derived, problem (5) can be solved efficiently by many existing solvers. In this work, we use IBM CPLEX (v12.4) to get the optimal solution.

5. Enforcing Connectivity Constraints in SSL

Before we give the formal definition of the connectivity constraints, we first introduce a binary adjacency graph $G = (V, \mathbf{A})$ to represent the spatial adjacency of segments, i.e., $\mathbf{A}(i, j) = 1$ if two segments s_i, s_j belong to the same image and are adjacent and $\mathbf{A}(i, j) = 0$ otherwise. Let $conn(G)$ denotes the set of all connected subgraphs of G. Of course, the nodes of each connected subgraph must represent segments belonging to the same image.

Each subgraph of G can be expressed with an indicator vector $\mathbf{u} \in \{0,1\}^N$. Hence we can identify $conn(G)$ with the set of indicator vectors $\mathbf{u} \in \{0,1\}^N$ representing connected subgraphs of G, i.e., $conn(G) \subset \mathcal{P}(\{0,1\}^N)$. By taking the convex hull of $conn(G)$ we obtain a polytope $Z = conv(conn(G)) \subset [0,1]^N$, where $[0,1]^N$ is the N-dimensional hypercube. We call Z a *connected subgraph polytope* of G.

The most well-known problem defined on Z is finding maximum-weight connected subgraph. As proved in [11],

even with a linear target function in this problem, it is NP-hard to optimize. Therefore, to make an optimization problem defined on Z to be polynomially solvable, we have to relax Z. To do this, we follow the method introduced in [14]. It is proved that each facet of Z can be defined by a linear inequality equation. For a better characterization of the facet, we need to define *vertex-separator sets* [14], as follows:

Given an undirected graph $G = (V, \mathbf{A})$, for any pair of vertices $i, j \in V, i \neq j, A(i, j) = 0$, the set $S \subseteq V \setminus \{i, j\}$ is said to be a *vertex-separator set* with respect to {i, j} if the removal of S from G disconnects i and j, which means that there exists no path between i and j in the subgraph with the vertex set $V \setminus S$.

In addition, we define $\bar{S}$ as an *essential vertex-separator set* if it is a vertex-separator set with respect to $\{i, j\}$ and any strict subset $T \subset \bar{S}$ is not. We denote with $\mathcal{S}(i, j)$ the set of all essential vertex-separator sets with respect to $\{i, j\}$. An example of essential vertex-separator sets is shown in Fig 2(c).

The proposed SSL algorithm with connectivity constraints is an iterative cutting-plane method. It alternates between solving a convex quadratic programming (QP) with linear inequality constraints (5) according to graph $(G, \mathbf{W})$, and adding a new connectivity constraint (facet) according to graph $(G, \mathbf{A})$.

Let $\mathbf{F}^t$ be a solution of (5) obtained at iteration t. We need to examine whether $\mathbf{F}^t$ violates the connectivity constraints. In order to do this, we need to define the connectivity constraints as linear constraints. Since our goal is to enforce connectivity of image segments belonging to the same object, i.e., having the same label, for a pair of segments s_i and s_j we only check the connectivity constraints if they are in the same image and have the same label l. We denote with $\mathcal{H}$ a set of all triples (i, j, l) such that s_i and s_j are in the same image, are not adjacent, i.e., $A(i, j) = 0$, and the probability for both segments have label $l \in L$ is positive, i.e., $\mathbf{F}^t_{il}, \mathbf{F}^t_{jl} > 0$. We call $\mathcal{H}$ a *check condition set*, since only for triples in $\mathcal{H}$ the connectivity condition needs to be checked.

As proved in [14], each facet of the polytope containing Z is defined by the following linear inequality for a label $l \in L$ and for all pairs (i, j) such that $(i, j, l) \in \mathcal{H}$:

$$\mathbf{F}^t_{il} + \mathbf{F}^t_{jl} - \sum_{k \in S} \mathbf{F}^t_{kl} - 1 \leq 0, \forall S \in \mathcal{S}(i, j) \qquad (6)$$

For a triple $(i, j, l) \in \mathcal{H}$, proving that no violated inequality exists or finding the most violated inequality in (6), which is given by

$$S^*(i, j, l) = \arg \max_{S \in \mathcal{S}_{(i,j)}} \sum_{k \in S} \mathbf{F}^t_{kl}, \qquad (7)$$

can be solved efficiently by computing max-flow [1] on an auxiliary directed graph. More details on how to construct the auxiliary directed graph can be found in [14].

Then find $(i^*, j^*, l^*) \in \mathcal{H}$ with the largest violation as

$$(i^*, j^*, l^*) = \arg \max_{(i,j,l) \in H} \sum_{k \in S^*(i,j,l)} \mathbf{F}^t_{kl} \qquad (8)$$

Let $S^*(i^*, j^*, l^*)$ be the vertex-separator set that yields the maximum value in (8). If

$$\mathbf{F}^t_{il} + \mathbf{F}^t_{jl} - \sum_{k \in S^*(i^*, j^*, l^*)} \mathbf{F}^t_{kl} - 1 \leq 0, \qquad (9)$$

the iterative process stops, since no constraints are violated. Otherwise, there is constraint violated, and it can be represented by the l^*th column in $\mathbf{M}$, with $M_{i^*l^*}, M_{j^*l^*} = 1$, and $M_{kl^*} = -1$ if $k \in S^*(i^*, j^*, l^*)$, and $M_{kl^*} = 0$ otherwise. Then $\mathbf{M}$ is added to the constraint set $\mathcal{C}$, and in next iteration, we solve Eq. (5) with the updated $\mathcal{C}$. This iterative process stops when no constraints are violated, or the change between $\mathbf{F}^t$ and $\mathbf{F}^{t+1}$ is smaller than a threshold.

Finally, the output $\mathbf{F}^*$ is binarized to the label indictor $\mathbf{Y}^*$ the same way as at the end of Section 3.2: for every node i determine $l^* = \arg\max_l \mathbf{F}^*_{il}$, and then set $\mathbf{Y}^*_{il} = 1$ if $l = l^*$ and $\mathbf{Y}^*_{il} = 0$ if $l \neq l^*$.

We call the proposed method **graph transduction with connectivity constraints (GTC)**, since it integrates RLGC graph transduction formulation and global connectivity constraints. The entire algorithm is described in Alg. 1.

Algorithm 1 Graph Transduction with Connectivity Constraints (GTC)

Input: $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-\frac{1}{2}}, \mathbf{A}, \mu, \sigma$
Output: $\mathbf{F}^* = \mathbf{F}^t$

Initial $\mathcal{C}^1$ as an empty set, t = 1
repeat
  obtain $\mathbf{F}^t$ by solving Eq (5).
  find the most violated constraints $S^*(i^*, j^*, l^*)$ using Eq (8)
  if Eq (9) holds for $S^*(i^*, j^*, l^*)$ **then**
    break
  end if
  derive linear equality constraint $\mathbf{M}$ from $S^*(i^*, j^*, l^*)$
  $\mathcal{C}^{t+1} \leftarrow \mathcal{C}^t \cup \mathbf{M}$
until $|\mathbf{F}^t - \mathbf{F}^{t-1}| < \sigma$

In Fig. 3, we visualize some examples of the most violated connectivity constraints discovered by our algorithm. For each left image, we use two green dots to show the pair

[1] http://pub.ist.ac.at/~vnk/software/[2]

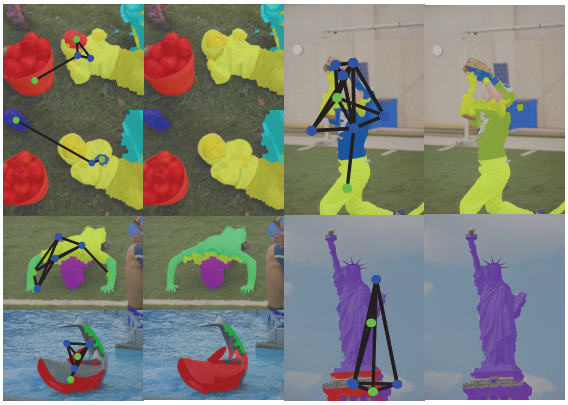

Figure 3. Visualization of the most violated connectivity constraints. Green dots: pair of segments with the same label that are not connected. Blue dots: essential vertex-separator set. Adjacency connection between segments is displayed using black lines.

of segments with the same label that are not connected. Essential vertex-separator set, which corresponds to the violated constraints, is shown using blue dots. We do not show the actual segments for better visualization. The edges are shown as black lines. In the right image, we show the result of resolved constraints after the next iteration. In particular, it should be noticed that there are two ways to resolve the constraints. One is to change the label for either of the two green dots so that two segments are no longer with the same label. The other one is to change the labels of some of the separating segments marked in blue dots to the label of the segments with green dots, which makes the two green dots segments connected. As the examples illustrate, our algorithms automatically determines which of the two kinds of solutions is better.

For any semi-supervised learning method that yields a continuous label confidence matrix $\mathbf{F}^*$, it is only possible to impose the connectivity constraints at the final binarization step of $\mathbf{F}^*$. For this we formulate the binarization step as solving a linear MRF problem with the connectivity constraints:

$$\mathbf{Y}^* = \arg\max_{\mathbf{Y}\in[0,1]^{N\times C}} \sum_{i=1}^{N}\sum_{l=1}^{C} \mathbf{Y}_{il}\mathbf{F}^*_{il} \qquad (10)$$

$$s.t. \quad \mathrm{tr}(\mathbf{MY}) \leq 1, \ \forall\, \mathbf{M} \in \mathcal{C}, \quad \sum_{l=1}^{C} \mathbf{Y}_{il} = 1.$$

This constrained problem can be solved using our GTC framework presented above (by only replacing the target function in (5) with the linear target function in (10)). This can be considered as a postprocessing step, and it can be applied to any semi-supervised learning method. We name this method as **GTCP**, where *P* stands for postprocessing.

If the constraint set $\mathcal{C}$ is empty, the solution of (10) is simply the argmax rule, as described at the end of Section 3.2, which is a standard binarization procedure for graph transduction SSL algorithms. Hence the proposed GTCP can be viewed as binarization of SSL solutions with connectivity constraints.

To summarize, RLGC solves the problem under a standard SSL framework, where only affinity graph $(G, \mathbf{W})$ is utilized, and the connectivity between nodes is not considered. In GTCP, the constraints are considered, but only at the final binarization step of label confidences. For GTC, we integrate connectivity with RLGC in an iterative framework. By utilizing the additional information from adjacent graph $(G, \mathbf{A})$, GTC can improve the label propagation process by increasing its robustness to the unstable affinity measurement in $(G, \mathbf{W})$. This is demonstrated by the experimental results in the next section.

Time Complexity: For the proposed GTC algorithm, in each iteration, solving convex QP with inequality constraints is very efficient. The main computation comes from finding the most violated connectivity constraints. However, this is carried out for each image and for each label independently. Therefore, if there are M images each decomposed into at most K segments, we only need to solve max-flow problem for at most MCK^2 times, where we recall that C is the number of object classes. In our method, K is usually a very small number (we follow [12], and obtain K = 18 segments using [13]). Also, this computation can be easily parallelized, which would further reduce the computation time.

6. Experimental Evaluation

We evaluate the proposed approach on a realistic and very challenging dataset called FlickrMFC dataset [12][2]. It consists 14 groups of images. Each group has 10 to 20 images, which are sampled from a Flickr photo stream. A finite number of repeating objects is contained in the same group, but the objects are not present in every image.

We follow the protocol of the interactive multiple foreground cosegmentation in [12], in which for each image group, 20% of images are randomly selected as training images, and the objects label in those images are provided. The labels represent a manual input of an user who marks the regions with main objects. The rest of images is used for testing. For each image set, 10 random splits is used, and the segmentation accuracy is averaged.

To evaluate the segmentation accuracy, the standard metric of PASCAL challenges is adopted, in which the intersection-over-union metric is measured. In particular, we follow the evaluation metric used in [12], where the segmentation accuracy is computed as $(\frac{GT_i \cap R_i}{GT_i \cup R_i})$.

We compare our methods GTCP and GTC to the state-

[2] http://www.cs.cmu.edu/~gunhee/r_mfc.html

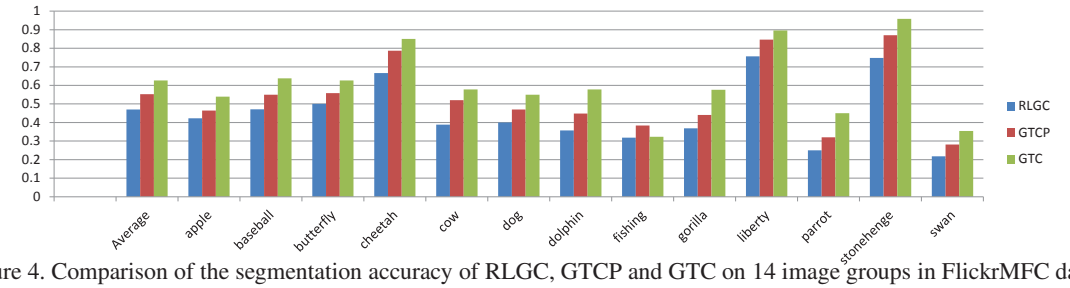

Figure 4. Comparison of the segmentation accuracy of RLGC, GTCP and GTC on 14 image groups in FlickrMFC dataset.

of-the-arts methods that have been evaluated on this dataset. The results are reported in Table 1 as the average accuracy over all 14 image sets. MFC-S [12] and our method can be viewed as typical SSL methods, since both require a small number of labeled data (labeled foreground objects in training images). The algorithm CoSand (COS) [13] and the discriminative clustering method (DC) [10], are not designed to handle irregularly appearing multiple foreground objects. Hence they require that all images are first manually divided into several subgroups so that the images of each subgroup share the same foreground object. Hence they also require user input, although no label information need to be explicitly provided as in a semi-supervised scenario. Only LDA-based unsupervised localization method (LDA) [16] is truly unsupervised. The results of LDA, DC, COS, MFC-S are copied from [12].

As can be seen in Table 1, the performance of RLGC [24], which belongs to classic graph transduction SSL methods, is comparable to MFC-S. This demonstrates the effectiveness of solving MFC problem in SSL framework, and in particular, the benefits of utilizing unlabeled data in addition to labeled data for label inference. Our postprocessing method GTCP applied directly to the label confidence scores of RLGC is able to significantly increase the segmentation accuracy, which demonstrates the benefits of the global connectivity constraints. Finally, our main proposed method GTC significantly outperforms all other methods. In particular, it increased the segmentation accuracy of MFC-S by 14%. Moreover, the fact that GTC outperforms our postprocessing method GTCP by over 7% shows the importance of enforcing the global connectivity constraints directly in the graph transduction SSL framework. Some example segmentation results of GTC are shown in Fig. 5.

LDA [16]	DC [10]	COS [13]	MFC-S [12]	RLGC [24]	GTCP our	GTC our
25.2	31.3	32.1	48.2	47.6	55.0	**62.6**

Table 1. Average segmentation accuracy (PASCAL intersection-over-union metric) on FlickerMFC dataset from [12].

We also give a detailed comparison of the segmentation accuracy of RLGC, GTCP and GTC on the 14 image groups in FlickerMFC dataset in Fig. 4. GTC outperforms RLGC and GTCP on all 14 groups of images except *fishing*.

7. Conclusion

In this work, we integrate the global connectivity constraints with graph transduction learning framework to address a very challenging task: multiple foreground cosegmentation. Connectivity constraints are naturally motivated by human visual perception in that we prefer to identify objects as connected image regions. They play a similar role in our approach by enforcing consistent class label assignment to connected image regions, which significantly improves the segmentation results. State-of-the art results are achieved on the benchmark dataset FlickrMFC, which clearly demonstrates the effectiveness of the proposed approach.

8. Acknowledgement

This work was supported by National Science Foundation under Grants IIS-0812118, BCS-0924164, OIA-1027897, and IIS-1257024.

References

[1] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen. Interactively co-segmentating topically related images with intelligent scribble guidance. *IJCV*, 2011. 1, 3

[2] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *PAMI*, 2004. 5

[3] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning*. 2006. 3

[4] M. D. Collins, J. Xu, L. Grady, and V. Singh. Random walks based multi-image segmentation: Quasiconvexity results and gpu-based solutions. In *CVPR*, 2012. 3

[5] R. Fergus, Y. Weiss, and A. Torralba. Semi-supervised learning in gigantic image collections. In *NIPS*, 2009. 3

[6] M. Guillaumin, J. J. Verbeek, and C. Schmid. Multimodal semi-supervised learning for image classification. In *CVPR*, 2010. 3

[7] D. S. Hochbaum and V. Singh. An efficient algorithm for co-segmentation. In *ICCV*, 2009. 1, 3

[8] T. Jebara, J. Wang, and S.-F. Chang. Graph construction and *b*-matching for semi-supervised learning. In *ICML*, 2009. 3

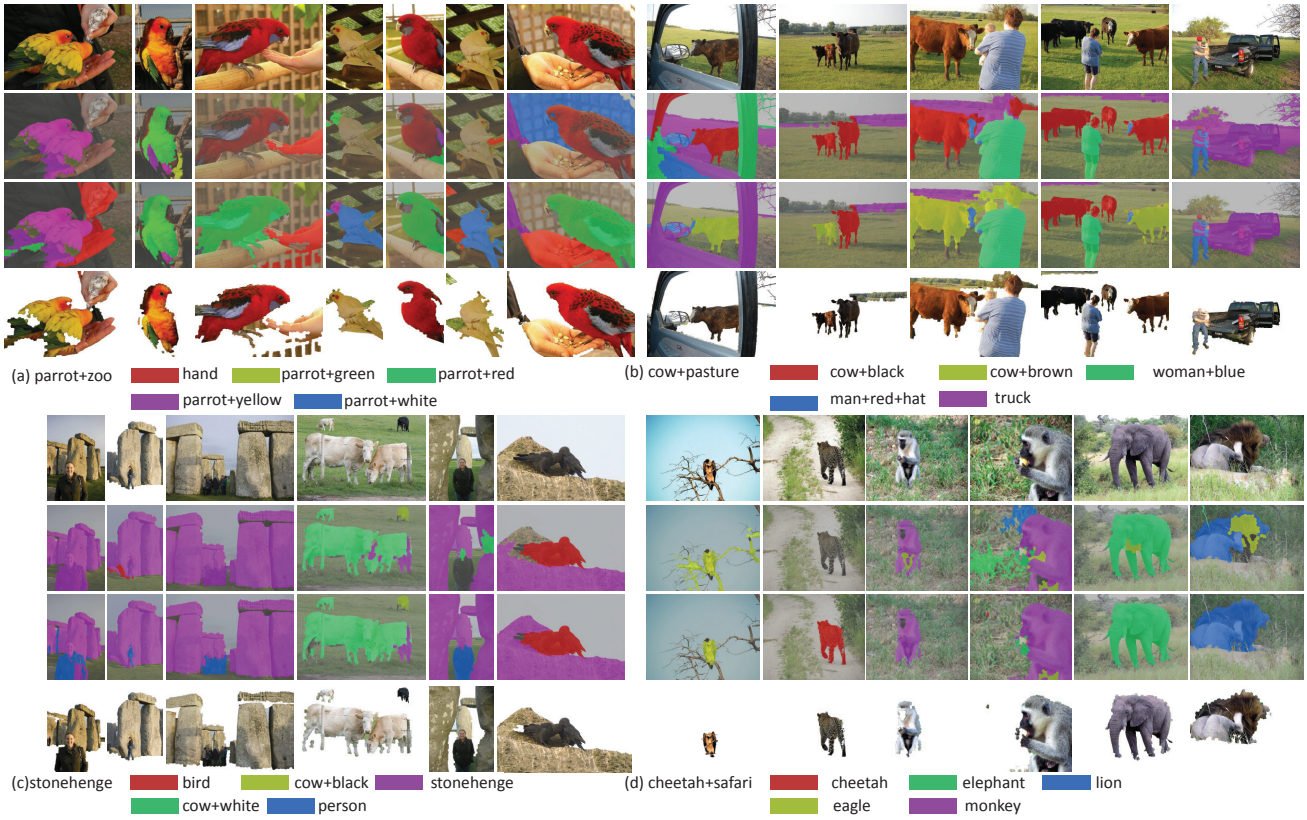

Figure 5. Examples of segmentation results on FlickrMFC dataset. First row: original images. Second row: segmentation results by RLGC. Third row: segmentation results by the proposed GTC. Fourth row: figure-ground segmentation results by GTC.

[9] A. Joulin, F. Bach, and J. Ponce. Multi-class cosegmentation. In *CVPR*, 2012. 2

[10] A. Joulin, F. R. Bach, and J. Ponce. Discriminative clustering for image co-segmentation. In *CVPR*, 2010. 1, 3, 7

[11] R. M. Karp. Maximum-weight connected subgraph problem, 2002. 2, 4

[12] G. Kim and E. P. Xing. On multiple foreground cosegmentation. In *CVPR*, 2012. 1, 2, 3, 6, 7

[13] G. Kim, E. P. Xing, F.-F. Li, and T. Kanade. Distributed cosegmentation via submodular optimization on anisotropic diffusion. In *ICCV*, 2011. 1, 2, 3, 6, 7

[14] S. Nowozin and C. H. Lampert. Global interactions in random field models: A potential function ensuring connectedness. *SIAM J. Img. Sci.*, 2010. 2, 5

[15] C. Rother, T. Minka, A. Blake, and T. Minkaand. Cosegmentation of image pairs by histogram matching incorporating a global constraint into mrfs. In *CVPR*, 2006. 1, 3

[16] B. Russell, W. Freeman, A. Efros, J. Sivic, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *CVPR*, 2006. 7

[17] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 2000. 3

[18] A. Shrivastava, S. Singh, and A. Gupta. Constrained semi-supervised learning using attributes and comparative attributes. In *ECCV*, 2012. 3

[19] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *PAMI*, 2010. 3

[20] S. Vicente, V. Kolmogorov, and C. Rother. Graph cut based image segmentation with connectivity priors. In *CVPR*, 2008. 2

[21] S. Vicente, V. Kolmogorov, and C. Rother. Cosegmentation revisited: models and optimization. In *ECCV*, 2010. 1, 3

[22] S. Vicente, C. Rother, and V. Kolmogorov. Object cosegmentation. In *CVPR*, 2011. 1, 3

[23] B. Wang and Z. Tu. Affinity learning via self-diffusion for image segmentation and clustering. In *CVPR*, 2012. 3

[24] J. Wang, T. Jebara, and S. fu Chang. Graph transduction via alternating minimization. In *ICML*, 2008. 1, 2, 4, 7

[25] J. Wang, S. Kumar, and S.-F. Chang. Semi-supervised hashing for scalable image retrieval. In *CVPR*, 2010. 3

[26] B. Zeisl, C. Leistner, A. Saffari, and H. Bischof. Online semi-supervised multiple-instance boosting. In *CVPR*, 2010. 3

[27] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and Schölkopf. Learning with local and global consistency. In *NIPS*, 2004. 2, 4

[28] X. Zhu. Semi-supervised learning literature survey, 2006. 2, 4