Computer Vision and Image Understanding 000 (2015) 1–10

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Optimizing the decomposition for multiple foreground cosegmentation[☆]

Haw-Shiuan Chang, Yu-Chiang Frank Wang[*]

Research Center for Information Technology Innovation, Academia Sinica, 128 Academia Road, Sec. 2, Taipei 115, Taiwan

ARTICLE INFO

Article history:
Received 20 November 2014
Accepted 13 June 2015
Available online xxx

Keywords:
Multiple foreground cosegmentation
Multi-class image cosegmentation
Figure–ground segmentation
Object discovery

ABSTRACT

The goal of multiple foreground cosegmentation (MFC) is to extract a finite number of foreground objects from an input image collection, while only an unknown subset of such objects is presented in each image. In this paper, we propose a novel unsupervised framework for decomposing MFC into three distinct yet mutually related tasks: image segmentation, segment matching, and figure/ground (F/G) assignment. By our decomposition, image segments sharing similar visual appearances will be identified as foreground objects (or their parts), and these segments will be also separated from background regions. To relate the decomposed outputs for discovering high-level object information, we construct foreground object hypotheses, which allows us to determine the foreground objects in each individual image without any user interaction, the use of pre-trained classifiers, or the prior knowledge of foreground object numbers. In our experiments, we first evaluate our proposed decomposition approach on the iCoseg dataset for single foreground cosegmentation. Empirical results on the FlickrMFC dataset will further verify the effectiveness of our approach for MFC problems.

1. Introduction

Aiming at extracting the commonly presented objects, image cosegmentation [1] performs joint segmentation on a set of images sharing overlapping contents. Originally, such cosegmentation is performed on a pair of input images (e.g., [1–4]), later its extension to handling a collection of relevant images attracts more attention from researchers. While supervised cosegmentation methods utilizing user interaction [5] or pre-trained classifiers [6,7] have been presented, some further proposed to observe visual features for performing cosegmentation in an unsupervised setting (e.g., [8,9]), so that the foreground objects can be identified automatically.

Recently, Kim and Xing [10,11] proposed to solve the problem of multiple foreground cosegmentation (MFC), which is to identify multiple foreground objects and the background simultaneously during the cosegmentation process. In MFC, the number of foreground objects in each image is typically unknown. In addition, the background presented across images might be different as well. Therefore, MFC is a very challenging task to address.

In this paper, we propose an unsupervised framework for MFC. As depicted in Fig. 1, we decompose MFC into three distinct computer vision problems: image segmentation, segment matching, and figure/ground (F/G) assignment. While the first task discriminates between image segments, the second task aims at identifying foreground segments across images, and the last task is to separate the foreground segments from background regions. As discussed in Section 3, our decomposition derives and associates solutions of each task in a unified optimization framework. In our experiments, we first evaluate the performance of our method on single foreground object cosegmentation using the iCoseg dataset [5]. The use of the FlickrMFC dataset [10] further verifies the application of our approach for MFC.

1.1. Our contributions

- We propose a novel framework which decomposes MFC into three well-studied computer vision tasks, i.e., image segmentation, segment matching, and figure/ground assignment. By properly associating and updating the outputs from each task, the goal of MFC can be achieved.
- With the proposed decomposition framework, background statistics can be observed across images, and thus background regions can be automatically disregarded. Moreover, the construction of object hypothesis is able to recover foreground objects containing multiple segments, while no prior knowledge on the number of foreground objects is needed.

2. Related works

Markov Random Fields (MRF) have been applied for image cosegmentation, which utilize graph-based optimization for recognizing the common foreground object from a pair of relevant images [1,3,4]. In [3], a variety of MRF models for cosegmentation have been discussed and compared. As noted in [3], dual decomposition

[☆] This paper has been recommended for acceptance by M. Pawan Kumar.
[*] Corresponding author.
E-mail addresses: ken77921@citi.sinica.edu.tw (H.-S. Chang), ycwang@citi.sinica.edu.tw (Y.-C.F. Wang).

http://dx.doi.org/10.1016/j.cviu.2015.06.004
1077-3142/© 2015 Elsevier Inc. All rights reserved.

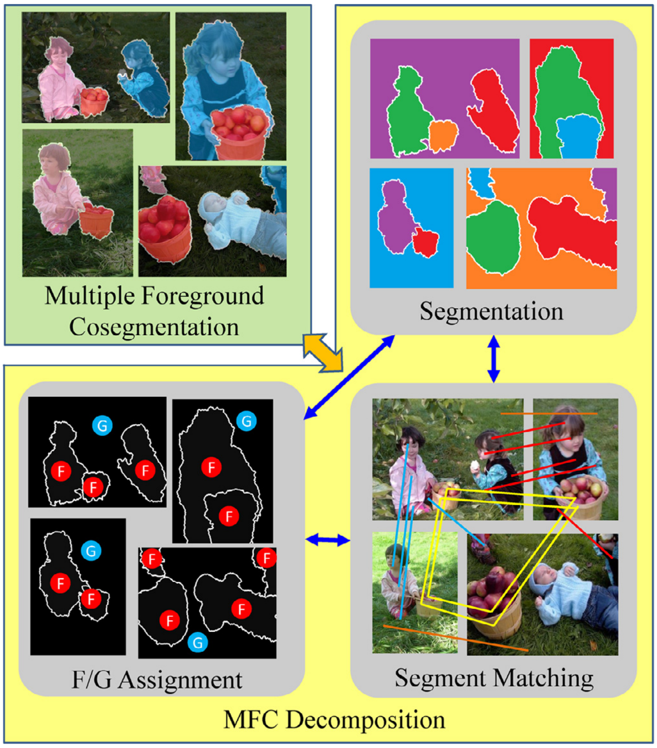

Fig. 1. Illustration of our proposed method, which decomposes MFC into the tasks of segmentation, segment matching, and figure/ground (F/G) assignment.

tackles the cosegmentation problem by advancing alternative optimization, which solves an EM-like optimization task on the smaller sub-problems. We note that, however, existing dual-decomposition based approaches focus on segmenting the single foreground object from a pair of input images with different backgrounds.

If the foreground objects exhibit significant visual appearance variations across multiple input images, more advanced matching techniques will be needed for solving the cosegmentation task (e.g., global descriptor matching [12], random forest regressor [6], graph-based matching [7], and SIFT flow [9]). Since the background regions in the input images are not necessarily distinct, it would be desirable to separate the foreground objects from such regions during cosegmentation. This is known as the figure/ground (F/G) assignment problem, which is typically solved by user interaction [5,13] or pre-trained classifiers [6,7]. Recent unsupervised cosegmentation approaches [8,9,14] derived the background models from each individual image (instead of a set of input images). Therefore, the robustness of their capability in F/G assignment will be limited.

Nevertheless, most of the above cosegmentation approaches focused on extracting a single type of foreground objects from input images. For multi-class cosegmentation methods described in [15,16,20], they did not assign foreground and background labels to their segmentation outputs even if only two classes were of interest.

Recently, Kim and Xing [10,11] proposed a problem called multiple foreground cosegmentation (MFC), which not only segments multiple types of foreground objects from the image collection, but F/G assignment will also be considered. As pointed out in [10], an exhaustive search for proper feature combination for each foreground object would be computationally prohibitive for MFC. Thus, labeled training data are required for F/G assignment in MFC (e.g., MFC-S [10], GTC [17], and MFRC [18]). On the other hand, Wang et al.[19] required the users to provide the exact number of foreground objects in input images. In practice, such user interaction or prior knowledge might not be easy to obtain, especially when the number of input images is large. While CoSand [15] has been applied for MFC in an unsupervised way (i.e., MFC-U in [10]), F/G assignment was not considered.

Table 1
Comparisons of recent cosegmentation methods. The symbol of ✓̸ indicates the task is partially addressed.

Methods	Unsupervised	F/G assignment	MFC
MRF	O	O	X
Batra et al. [5]	X	O	X
Vicente et al. [6]	X	O	X
Rubio et al. [7]	O	O	X
Rubinstein et al. [9]	O	O	X
Faktor and Irani [14]	O	O	X
CoSand [15]	O	✓̸	X
Joulin et al. [16]	O	✓̸	X
MFC-S [10]	X	O	O
GTC [17]	X	O	O
MFRC [18]	X	O	O
Wang et al. [19]	✓̸	O	O
Li et al. [20]	O	✓̸	O
MFC-U [10]	O	✓̸	O
Ours	O	O	O

Table 2
The list of notations. $R()$ and $l()$ denote the region and label of interest, respectively.

Notation		Explanation
Region	Label	
$R(C_k)$	C_k	The region of the kth foreground class/part and its label
$R(G_i)$	G_i	The background region in image I_i and its label
$R(\mathcal{F}_r)$	$\mathcal{F}_r$	The region of the rth foreground objects and its label
p_i^j	$l(p_i^j)$	The jth superpixel in image I_i and its label
s_i^n	$l(s_i^n)$	The nth segment in image I_i and its label (i.e., the set of connecting superpixels with the same label)
O_i^m	$l(O_i^m)$	The mth foreground object hypothesis in image I_i and its labels

As highlighted in Table 1, we propose a decomposition framework for MFC in this paper. Our experiments will verify the effectiveness of our approach for both single and multiple foreground cosegmentation. For the ease of understanding, Table 2 summarizes the notations used in this paper.

3. Decomposing MFC

As illustrated in Fig. 2, we propose to decompose MFC into three different tasks, which can be associated with each other for identifying foreground object parts and background regions from the input images $I_1, \ldots, I_N$. For the jth superpixel p_i^j in image I_i, we will determine whether its label $l(p_i^j)$ belongs to one of the foreground class/part C_k or the background regions. Our decomposition can be viewed as solving the following optimization problem:

$$\min \sum_i E(\mathbf{l}_i) \;\; s.t. \begin{cases} l(p_i^j) = \hat{l}(s_i^n), \text{ for } \; p_i^j \in s_i^n \\ P_{\mathcal{F}}(C_k) > T, \text{ for } \; l(p_i^j) = C_k, \end{cases} \forall i, j, \tag{1}$$

where E indicates the energy function for segmentation, and $\mathbf{l}_i = [l(p_i^j)]_{j=1,\ldots,N_p}$ is the label vector of image I_i with its length equal to the number of superpixels N_p. The jth element $l(p_i^j)$ in $\mathbf{l}_i$ is the label of superpixel p_i^j in image I_i. We have s_i^n and $\hat{l}(s_i^n)$ as the nth segment and its desirable label in image I_i, respectively. Note that the image segment determined in this work represents the set of connecting superpixels with the same label, and the image segments with similar visual appearances across images will be identified (via segment matching) and be assigned the same label (see more details in Section 3.2). The function $P_{\mathcal{F}}()$ in (1) denotes the foreground

Please cite this article as: H.-S. Chang, Y.-C.F. Wang, Optimizing the decomposition for multiple foreground cosegmentation, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.06.004

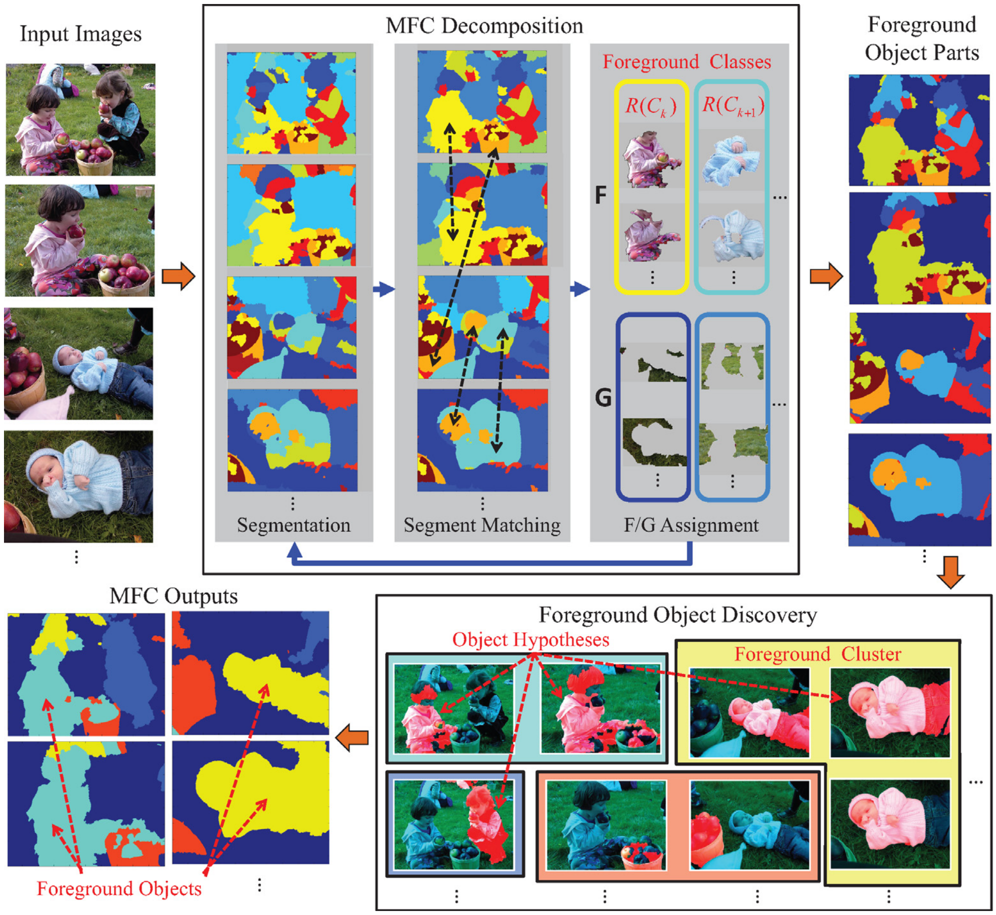

Fig. 2. Our proposed MFC framework, which alternates between three decomposed tasks of segmentation, segment matching and figure/ground (F/G) assignment, followed by an object discovery stage for identifying multiple foreground objects of interest.

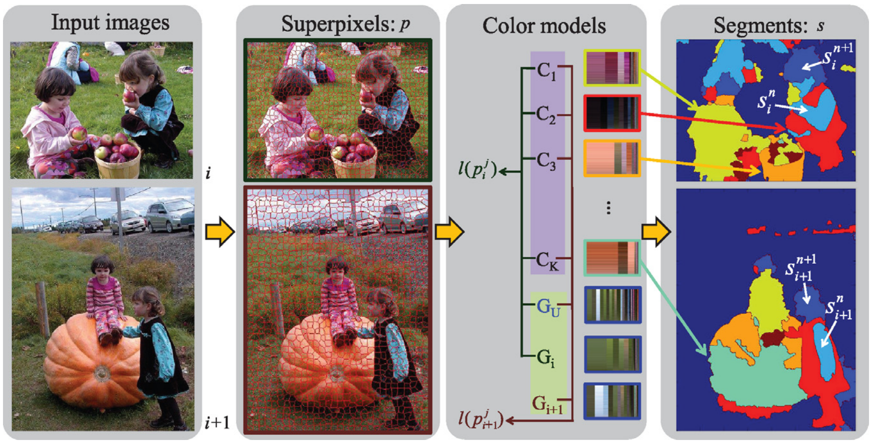

Fig. 3. Segmentation in MFC. First column: examples of input images; second column: superpixels p collected from each image; third column: visualization of the color models derived for the foreground classes $C_1, \ldots, C_K$ and background classes G_U, G_i, and G_{i+1}; fourth column: foreground segments s determined by solving the optimization problem of (2).

probability function with threshold T, which will be discussed later in Section 3.3.

By optimizing (1), we effectively separate the input images into different segments, which are either associated with a particular foreground object class/part or the background regions. We note that the objective function of (1) addresses our decomposed task of image segmentation, and the constraints of $l(p_i^j) = \hat{l}(s_i^n)$ and $P_{\mathcal{F}}(C_k) > T$ correspond to the remaining tasks of segment matching and F/G assignment, respectively. The technique of alternative optimization is applied for solving the proposed optimization problem of (1). We now detail each decomposed task in the following subsections.

3.1. Segmentation

As the primary objective function of our decomposition task, the goal of image segmentation is to assign class labels to each pixel in the image collection. For computation efficiency, we apply the technique of [21] for producing superpixels as input data (instead of pixels), as shown in Fig. 3. In our work, we fix the number of superpixels of each

Please cite this article as: H.-S. Chang, Y.-C.F. Wang, Optimizing the decomposition for multiple foreground cosegmentation, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.06.004

image as $N_p = 1200$. As a result, we choose to solve the above label assignment problem in the superpixel level.

Following [9], we apply GrabCut [22] for performing segmentation. Given each image I_i and its superpixels, we determine the image segments of this image by minimizing E in (1), which is defined as

$$E(\mathbf{l}_i) = E_D(\mathbf{l}_i) + \lambda E_S(\mathbf{l}_i), \tag{2}$$

where E_D and E_S are the data and smoothness energy terms, respectively. The factor λ in (2) regularizes the smoothness term (a larger λ prefers larger segments as the segmentation output). As suggested by Rother et al. [1], we apply the Bag-of-Words (BoW) model in the CIE Lab color space for describing the image segments. We use GMM with 32 Gaussian components to determine the color histogram $\mathbf{h}_{p_i^j}$ for each superpixel p_i^j. Thus, the data term E_D is defined by

$$E_D(\mathbf{l}_i) = \sum_{j=1}^{N_p} -\log\left(P(l(p_i^j)|\ p_i^j)\right), \tag{3}$$

where $P(l(p_i^j)|p_i^j)$ is the probability of assigning superpixel p_i^j with class label $l(p_i^j)$. To calculate the above probability value, we apply the settings of [2,17] and utilize the normalized χ^2 distance to measure the similarity between $\mathbf{h}_{p_i^j}$ and $\mathbf{h}_{R(l(p_i^j))}$. Note that $R(l(p_i^j))$ represents image regions with class label $l(p_i^j)$.

The smoothness term E_S is to penalize the cases when the superpixels in a homogeneous region are separated into different classes. As suggested in [22], we define E_S as

$$E_S(\mathbf{l}_i) = \sum_{p_i, q_i} \mathbb{1}_{(l(p_i)\neq l(q_i))}\left(-\log\left(P_C(p_i, q_i)\right)\right), \tag{4}$$

where q_i denotes the neighboring superpixels of superpixel p_i. We have $\mathbb{1}()$ as the indicator function, and $P_C(p_i, q_i)$ is the contour probability for separating p_i and q_i. Note that we drop the superpixel index j in (4) for simplicity. In our work, $P_C(p_i, q_i)$ is computed in terms of the distance between color histograms, which is known as probability boundary (Pb). However, unlike Pb determined in [23], we do not require any training data for calculating such probabilities.

As noted earlier, this segmentation task aims at assigning a class label to each superpixel in the image collection. Let K as the number of foreground classes/parts of interest, we thus have $C_1, \ldots, C_K$ as their class labels. Since we do not assume that the background regions in each image are visually similar, we have background class labels G_i representing the unique background regions in each image I_i, and a common background class label G_U to be shared by all images (see Fig. 3 for examples). To prevent possible error propagation during our alternative optimization process, representations of G_i and G_U are simply determined by the features observed from boundaries B_i (of each image I_i) and their union $B_U = B_i \bigcup \cdots \bigcup B_N$, respectively (see details in [24,25]). Once this segmentation step is complete, a class label from $\{C_1, \ldots, C_K\}$ or $\{G_i, G_U\}$ would be assigned to each superpixel, and such outputs can be summarized as $R(C) = \{R(C_1), \ldots, R(C_K)\}$, or $R(G) = \{R(G_1), \ldots, R(G_N), R(G_U)\}$, respectively.

3.2. Segment matching

Recall that, we define an image segment as the set of connecting superpixels sharing the same class label. Take the rightmost column of Fig. 3 for example, the child consists of multiple image segments of head, arm, and body, while each segment contains superpixels with the same class/part label.

Unfortunately, the previous segmentation step does not guarantee that the image segments of the same class label (but across images) would exhibit similar visual appearances. Similarly, those of different class labels (in a single or across images) might share similar visual appearance information. If either of the above cases occurs, one will not be able to successfully identify the foreground objects of interest from the image collection. This is the reason why we need to impose the first constraint in (1), which corresponds to our decomposed task of segment matching. More precisely, we need to match image segments with similar visual appearance across images, so that we can assign/update their labels accordingly for cosegmentation purposes.

In our work, we apply the settings of [2,4,7] and utilize textural features to describe visual appearances of the image segments. We consider the use of 17 Gaussian/Laplacian-type filters and their derivatives in the CIE Lab space [26,27]. We quantize the filter responses into 32 bins by GMM for deriving their histogram representations. With such features, we have the class labels for each segment s_i^n in (1) satisfying

$$\hat{l}(s_i^n) = \underset{l(s_i^n)\in\{C_1,\ldots,C_K,G_i,G_U\}}{\arg\min} \chi^2(\mathbf{t}_{s_i^n}, \mathbf{t}_{R(l(s_i^n))}),\ \forall i, n, \tag{5}$$

where $\mathbf{t}_{s_i^n}$ and $\mathbf{t}_{R(l(s_i^n))}$ are the textural feature of segment s_i^n and that of the image regions with $l(s_i^n)$, respectively.

Note that the label of each segment $\hat{l}(s_i^n)$ can only be selected from $\{C_1, \ldots, C_K, G_i, G_U\}$. Since this constraint is enforced across images, performing (5) is to enforce all superpixels/segments with similar textural features having the same foreground class label. In other words, segment matching is effectively achieved.

3.3. F/G assignment

As noted earlier, it is desirable to making distinction between foreground and background regions for MFC. Thus, the second constraint in (1) addresses this F/G assignment issue. In particular, we enforce the constraint that the probability of each detected foreground class needs to be above a predetermined threshold T, i.e., $P_{\mathcal{F}}(C_k) > T$ where C_k denotes the kth foreground class label (and we set $T = 0.2$ in this paper). As suggested in [24,25,28], the use of image boundaries (e.g., the dark green and brown frames of images in the second column of Fig. 3) is a good background prior for F/G assignment. Therefore, we define our foreground probability function as follows:

$$P_{\mathcal{F}}(C_k) = \frac{1}{1 + boundary(R(C_k))}, \tag{6}$$

where $R(C_k)$ denotes the set of superpixels assigned with label C_k. The function *boundary* $(R(C_k))$ calculates the number of pixels in $R(C_k)$ covering the boundaries of the input images, which is normalized by the size of $R(C_k)$. It can be seen that if no pixel in $R(C_k)$ locates at any image boundaries, we have $P_{\mathcal{F}}(C_k) = 1$. When more pixels in $R(C_k)$ are at image boundaries, smaller $P_{\mathcal{F}}(C_k)$ will be resulted, and thus it is more likely to have this label set as background instead.

While most existing cosegmentation approaches were not able to handle the F/G assignment problem, some recently proposed works chose to apply more complex techniques for solving this task. Rubinstein et al. [9] utilized RC [29] (i.e., saliency information), while pre-trained F/G classifiers [24] were applied in [6,7] for separating foreground from background regions. Nevertheless, these methods performed F/G assignment on each individual image, not across input images. As verified later, our F/G assignment based on foreground probability functions would exhibit improved capabilities in separating foreground and background regions in the MFC outputs.

3.4. Optimizing MFC decomposition

Now we discuss how we solve the proposed decomposition MFC framework jointly addressing the three fundamental yet challenging computer vision tasks. In fact, even optimizing E in (1) (i.e., performing image segmentation) without considering the two constraints has been known as a NP-hard problem [30,31]. As shown in Algorithm 1, we advance the technique of alternative optimization [3,9,10,16,31], which fixes the solutions to two of the three decomposed tasks and

Please cite this article as: H.-S. Chang, Y.-C.F. Wang, Optimizing the decomposition for multiple foreground cosegmentation, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.06.004

Algorithm 1: MFC decomposition.

```
Input: Images $I_1, \ldots, I_N$, cluster numbers $K_{init}$ and $K_{min}$
Output: Image regions of all classes $R(C)$, $R(G)$
Initialization:
  Collect superpixels from $I_1, \ldots, I_N$ by [21]
  $R(C) \leftarrow$ GMM clustering of superpixels
  with $K = K_{init}$
while $K > K_{min}$ do
    Segmentation:
      Derive color feature models from $R(C)$
      for $i = 1$ to $N$ do
          Labeling vector $\mathbf{l}_i \leftarrow \arg\min E(\mathbf{l}_i)$ (2)
      Update $R(C)$ and $R(G)$ by labels $\mathbf{l}_{1,\ldots,N}$
    Segment matching:
      Derive texture feature models from $R(C)$
      Update $R(C)$ and $R(G)$ by (5)
    F/G assignment:
      for $k = 1$ to $K$ do
          Remove foreground class $C_k$ if $P_{\mathcal{F}}(C_k) \leq T$ (6)
      $K \leftarrow$ Number of the foreground classes
```

solves the remaining one at each iteration. This makes the optimization problem of (1) more tractable.

To initialize the alternative optimization process, we over-segment input images via GMM in the CIELab space with the cluster number $K = K_{init}$. Since we do not solve the F/G assignment problem until the iteration starts, each superpixel will be initially assigned a foreground class label from $\{C_1, \ldots, C_K\}$. For the segmentation step in each iteration, we calculate the color feature models (via GMM) for each $R(C_k)$, which corresponds to image regions with label C_k. With such color models observed, we minimize E in (2) using the technique of α–β swap [30,32,33], and update the labels of each superpixel accordingly.

For the segment matching step satisfying the constraint $l(p_i^j) = \hat{l}(s_i^n)$ in (1), we first collect image segments and their labels determined by the segmentation step. Next, we apply (5) which makes foreground image segments with similar textural features having the same class label, even if they are across different input images. Finally, the F/G assignment step takes (1) and removes foreground classes C_k if $P_{\mathcal{F}}(C_k) \leq T$. If no $P_{\mathcal{F}}(C_k)$ is below T, we still disregard the one with the smallest $P_{\mathcal{F}}(C_k)$. This not only guarantees the decrease of the number of candidate foreground classes until $K = K_{min}$, but also makes our MFC decomposition insensitive to the selection of T.

It is worth noting that the unsupervised MFC problem is very challenging, since it is NP-hard [10]. As a result, one cannot expect a simple, unified learning or segmentation algorithm for producing satisfactory MFC results (e.g., [10]). While it is possible to integrate all our proposed components into a single MRF model for MFC (similar to the MRF models utilized in [1,3,4]), such models would be very complex and thus be much more computationally expensive. In our work, our MFC decomposition framework not only makes the optimization of (1) more feasible, it also introduces additional flexibility which allows users to replace each decomposed component by other existing techniques. Later in our experiments, we will confirm both effectiveness and flexibility of our MFC framework.

4. Discovery of multiple foreground objects

With our proposed decomposition framework, the foreground and background image regions can be identified from the input image collection, and the foreground image segments with the same visual appearances will be further grouped as the same class of interest.

Algorithm 2: Foreground object discovery.

```
Input: All foreground segments $s_i^n$ in $R(C_1), \ldots, R(C_K)$
Output: Labels of all foreground segments

Hypothesis construction:
  for $i \leftarrow 1$ to $N$ do
      for $k \leftarrow 1$ to $K$ do
          Find all object hypotheses which contains $k$
          different class labels by (7)
Object discovery:
  Extract the textural histogram $\mathbf{t}$ of each object
  hypotheses and separate them into $M$ clusters by
  Mean Shift clustering
  $\mathbf{t}_{R(\mathcal{F}_1)}, \ldots, \mathbf{t}_{R(\mathcal{F}_M)} \leftarrow$ Center of each cluster
  foreach $s_i^n$ do
      Determine the labels of segment $s_i^n$ by (9)
```

Nevertheless, as illustrated in Fig. 4, the image segments (i.e., the decomposition outputs) might correspond to object parts instead of the entire foreground object. Therefore, the final task of our work is to discover the objects of interest across images. As later summarized in Algorithm 2, we first construct the foreground object hypotheses, and the foreground objects will be recovered accordingly.

4.1. Construction of foreground object hypotheses

For semantic segmentation like MFC, determining a specific type of object from a set of image segments (with different foreground class labels) is an integer programming (IP) problem [10], which is NP-complete and very difficult to solve even if the object of interest is known in advance. Inspired by [10,24], we propose to construct foreground object hypotheses to discover the foreground objects of interest. This strategy allows us to extract overlapping image regions for identifying foreground objects, so that the foreground objects can be automatically discovered from our decomposed outputs.

We now discuss how we construct the foreground object hypotheses for MFC. Since the foreground objects in input images typically consist of connected segments, we employ the connectivity constraint in the proposed object hypotheses as suggested in [10,17]. More precisely, we consider that each object hypothesis corresponds to a particular foreground object or a part of it (e.g., an entire or upper body of a person). Thus, for each image, a foreground object hypothesis will be defined as a set of connected segments, which contains one or multiple foreground class labels. Given the number of foreground class labels K and the segments produced from our MFC decomposition, we now define O_i^m as the mth object hypothesis of image I_i, and we have $l(O_i^m)$ indicating the set of foreground class labels contained in this hypothesis. To be more specific, O_i^m and $l(O_i^m)$ are defined as follows:

$$O_i^m = \bigcup_j \{s_i^m(j)\}, \; l(O_i^m) = \bigcup_j \{l(s_i^m(j))\}, \tag{7}$$

where $\bigcup\{s_i^m(j)\}$ represents the connected image segments in image I_i, and j is the segment index. We note that the number of foreground class labels is between 1 and K (i.e., $|l(O_i^m)| = k$ and $k = 1, \ldots, K$). Fig. 4 shows an example of our foreground object hypotheses, in which $|l(O_i^m)| = 3$ is considered.

It is worth noting that although the number of foreground class labels in $|l(O_i^m)| = k$ is between 1 and K, we do not need to exhaustively search for all possible label combinations when constructing the corresponding object hypotheses. In addition to the identified background regions identified by the decomposition framework,

Please cite this article as: H.-S. Chang, Y.-C.F. Wang, Optimizing the decomposition for multiple foreground cosegmentation, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.06.004

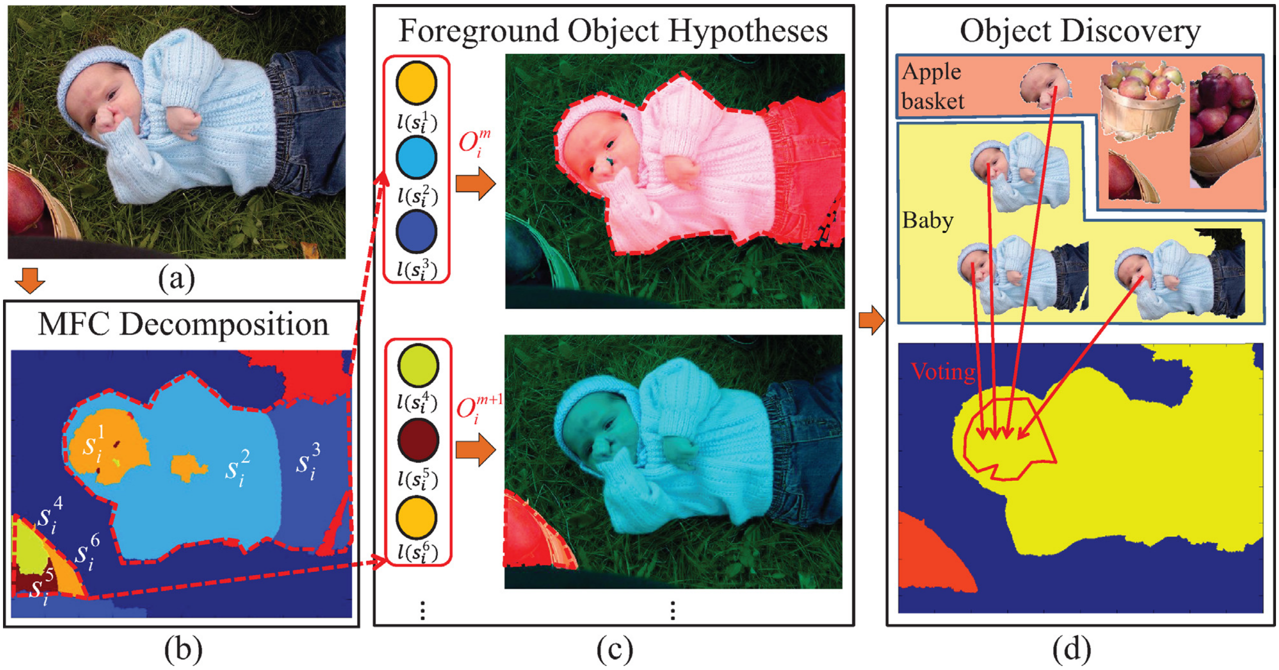

Fig. 4. Example of object discovery via foreground object hypotheses. (a) Input image I_i, (b) output segments $s_i^1, \ldots, s_i^6$ with labels $l(s_i^1), \ldots, l(s_i^6)$ of Section 3, (c) example object hypotheses O_i^m and O_i^{m+1} with $|l(O_i^m)| = |l(O_i^{m+1})| = 3$, and (d) assigning segments of faces into the cluster which corresponds to the foreground object of *baby*.

the use of the aforementioned connectivity constraint further disregards the extracted yet disconnected image segments. Other MFC approaches like [10] only consider a fixed number of segment combinations for identifying the foreground objects, as verified by our experiments. This would limit their capabilities in identifying the foreground objects as verified later in our experiments.

4.2. Object discovery via foreground object hypotheses

With all object hypotheses O_i^m with different $l(O_i^m)$ numbers (from 1 to K) are constructed, the remaining task of object discovery is to identify which particular foreground objects each O_i^m corresponds to. Similar to segment matching in our MFC decomposition, we represent each object hypothesis O_i^m using its textural histogram $\mathbf{t}_{R(O_i^m)}$ (see textural features determined in Section 3.2). The Mean Shift algorithm [34] is applied to group the constructed hypotheses into M different clusters (i.e., foreground objects $\mathcal{F}_1, \ldots, \mathcal{F}_M$). Once the clustering process is complete, each cluster can be viewed as the foreground object, while it can be described by the associated cluster center (i.e., $\mathbf{t}_{R(\mathcal{F}_1)}, \ldots, \mathbf{t}_{R(\mathcal{F}_M)}$).

It is worth repeating that while K in our MFC decomposition represents the number of foreground class/part labels, M in this final clustering step indicates the number of foreground objects of interest. Take Fig. 4 for example, we have $K = 6$ foreground classes/parts extracted during decomposition, and $M = 2$ objects determined as the MFC outputs (i.e., *apple basket* and *baby*). As depicted in Fig. 4(d), for overlapping O_i^m (and their superpixels) which appear in multiple clusters, a simple voting strategy will be applied to decide its final foreground object label.

When performing the above clustering process for object discovery, we measure the χ^2 distance between the textural histogram of object hypothesis O_i^m and that of foreground object $\mathcal{F}_r$, i.e., $\chi^2(\mathbf{t}_{R(O_i^m)}, \mathbf{t}_{R(\mathcal{F}_r)})$. With the converge and termination of the clustering process, we then calculate the distance $d(s_i^n, \mathcal{F}_r)$ between each segment s_i^n and foreground object $\mathcal{F}_r$ as follows:

$$d(s_i^n, \mathcal{F}_r) = \sum_{\{O_i^m | s_i^n \in O_i^m\}} \chi^2(\mathbf{t}_{R(O_i^m)}, \mathbf{t}_{R(\mathcal{F}_r)}), \tag{8}$$

where the distance $d(s_i^n, \mathcal{F}_r)$ is calculated by summing up all χ^2 distances between foreground object $\mathcal{F}_r$ and object hypotheses O_i^m which contain s_i^n (i.e., $\{O_i^m | s_i^n \in O_i^m\}$). Finally, the label $l(s_i^n)$ of segment s_i^n is determined by

$$l(s_i^n) = \arg\min_{\mathcal{F}_r} d(s_i^n, \mathcal{F}_r). \tag{9}$$

In other words, s_i^n will be assigned to the cluster (i.e., the foreground object) which is closest to it.

5. Experiments

To evaluate our proposed MFC approach, we conduct experiments on cosegmentation datasets which contain single and multiple foreground objects. The source code can be accessed at http://mml.citi.sinica.edu.tw/papers/MFC_code_CVIU_2015/.

5.1. Single foreground cosegmentation

We first conduct cosegmentation experiments on the iCoseg dataset [5], which contains 643 images collected from Flickr.[1] For each image, a single type of foreground object is presented, while there are 38 different objects of interest available. We select $\lambda = 0.4$ in (3) and the number of foreground class labels $K_{init} = 16$ for initialization. Due to the presence of only a single foreground object in each image, we set $K = K_{min} = 1$ directly taking the decomposition outputs as cosegmentation results (i.e., no MFC object discovery is required).

We compare our proposed method with recent cosegmentation approaches proposed by Vicente et al. [6], Rubio et al. [7], Rubinstein et al. [9], Wang et al. [35], Dai et al. [36], Joulin et al. [16], and the methods of DC [2] and CoSand [15]. Note that training data are required in [6]. For the completeness of comparisons, we also conduct experiments on two different subsets of iCoseg: the subset of 16 foreground objects considered in [6], and that of 30 objects in [9]. For our proposed method, we present the averaged cosegmentation results of 10 trials, each starts from a random initialization for GMM in the beginning of our MFC decomposition.

[1] http://www.flickr.com/ .

Please cite this article as: H.-S. Chang, Y.-C.F. Wang, Optimizing the decomposition for multiple foreground cosegmentation, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.06.004

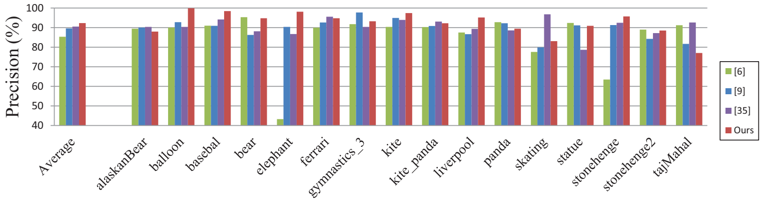

Fig. 5. Performance comparisons (in terms of precision) on the subset of the iCoseg dataset considered in [6].

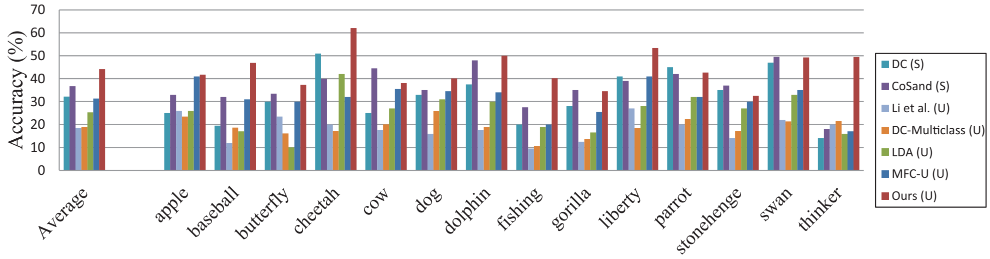

Fig. 6. Accuracy comparisons in terms of Jaccard similarity (%) on the FlickrMFC dataset.

Table 3
Precision and Jaccard similarity comparisons (%) of the iCoseg Dataset. Note that the best performance for each task is highlighted in bold.

Methods	Entire dataset		Subset from [9]		Subset from [6]	
	P	J	P	J	P	J
[6]	–	–	–	–	85.3	62.0
[7]	–	–	–	–	83.9	–
[9]	–	–	**89.8**	**69.3**	89.6	**67.6**
[35]	–	–	–	–	90.5	–
[36]	89.5	–	–	–	–	–
[2]	80.0	41.5	80.0	43.4	74.8	47.9
[15]	–	–	70.2	42.6	–	–
[16]	70.5	39.5	72.5	43.0	73.0	46.6
Ours	**90.0**	**64.2**	89.6	65.6	**92.3**	65.1

Table 4
Cosegmentation results of FlickrMFC (%). Note that S, U, RC, Decom, Iter, OD, Seg, Match represent supervised, unsupervised, region-contrast saliency detection [29], MFC decomposition, iterative optimization, object discovery, image segmentation, and segment matching, respectively.

Methods		Accuracy (J)
DC [2] (S)		32.2
CoSand [15](S)		36.7
Li et al. [20] (U)		18
DC-Multiclass [16] (U)		18.9
LDA [37] (U)		25.2
MFC-U [10] (U)		31.2
Ours (U)		**44.2**
Ours (U)	w/ RC	43
	Decom w/o OD	40.6
	Decom w/o OD and Iter	38.8
	Seg + Match	31.8
	Seg	27.5
	Initialization	24.8

For quantitative evaluation, we apply two different metrics: precision P indicating the percentage of correctly labeled pixels, and the Jaccard similarity J calculating $\frac{GT \bigcap R}{GT \bigcup R}$ (note that GT and R represent the ground truth and segmented foreground object regions, respectively). Table 3 lists the cosegmentation results of different approaches. In addition, Fig. 5 compares the precision of each foreground object in the subset of [6]. From the above table and figure, we see that our method outperformed multiclass cosegmentation approaches like [15,16]. We achieved comparable results as the state-of-the-art single foreground approach of [9], which was particularly proposed for cosegmentation of iCoseg. We note that the variance of our performance was less than 1.5%, and thus our method is not sensitive to GMM initialization.

5.2. Multiple foreground cosegmentation

To evaluate the MFC performance, we consider the FlickrMFC dataset [10]. This dataset contains 14 image groups which are also sampled from Flickr, while each image group consists of 12–20 images with the number of foreground objects ranging from 3 to 8. This dataset is very challenging, since it allows distinct backgrounds presented in the image collection. As noted in prior discussions, this makes MFC (especially for the F/G assignment task) very difficult to solve.

In our experiments, we follow the setting of [10] and extend the Jaccard similarity for evaluating the cosegmentation accuracy, i.e., $J(GT_c, R(\mathcal{F})) = \max_{r=1,\ldots,M} \frac{GT_c \bigcap R(\mathcal{F}_r)}{GT_c \bigcup R(\mathcal{F}_r)}$. Note that GT_c and $R(\mathcal{F}_r)$ indicate the regions of each ground truth object category c and those detected for object j, respectively. For other parameters to be determined, we have the initial number of foreground class labels $K_{init} = 32$ (which is larger than $K_{init} = 16$ for single foreground cosegmentation on iCoseg), and we choose $\lambda = 0.1$ in (3) which prefers smaller image segments in our MFC decomposition.

Fig. 6 compares the cosegmentation performance of different MFC methods: DC [2], CoSand [15], Li et al. [20], DC-Multiclass [16], LDA [37], MFC-U [10], and ours. As note that in [10], DC and CoSand are supervised approaches for solving MFC, while the others and ours are all performed in an unsupervised setting. It is worth noting that, images of *cheetah* are typically with foreground objects exhibiting large visual appearance variances, while those of *thinker* generally contain various types of backgrounds across images. Due to the introduced capability of performing segment matching and F/G assignment across images, we achieved significantly improved MFC results on these two categories.

Table 4 lists the average accuracy of different methods. It clearly shows that we achieved about 7.5–19% improvements over others (e.g., we outperformed the unsupervised MFC approach (MFC-U) [10] by 13.3%). Similar to our experiments on iCoseg, we present our results using the average accuracy of 10 trials (each starts from a

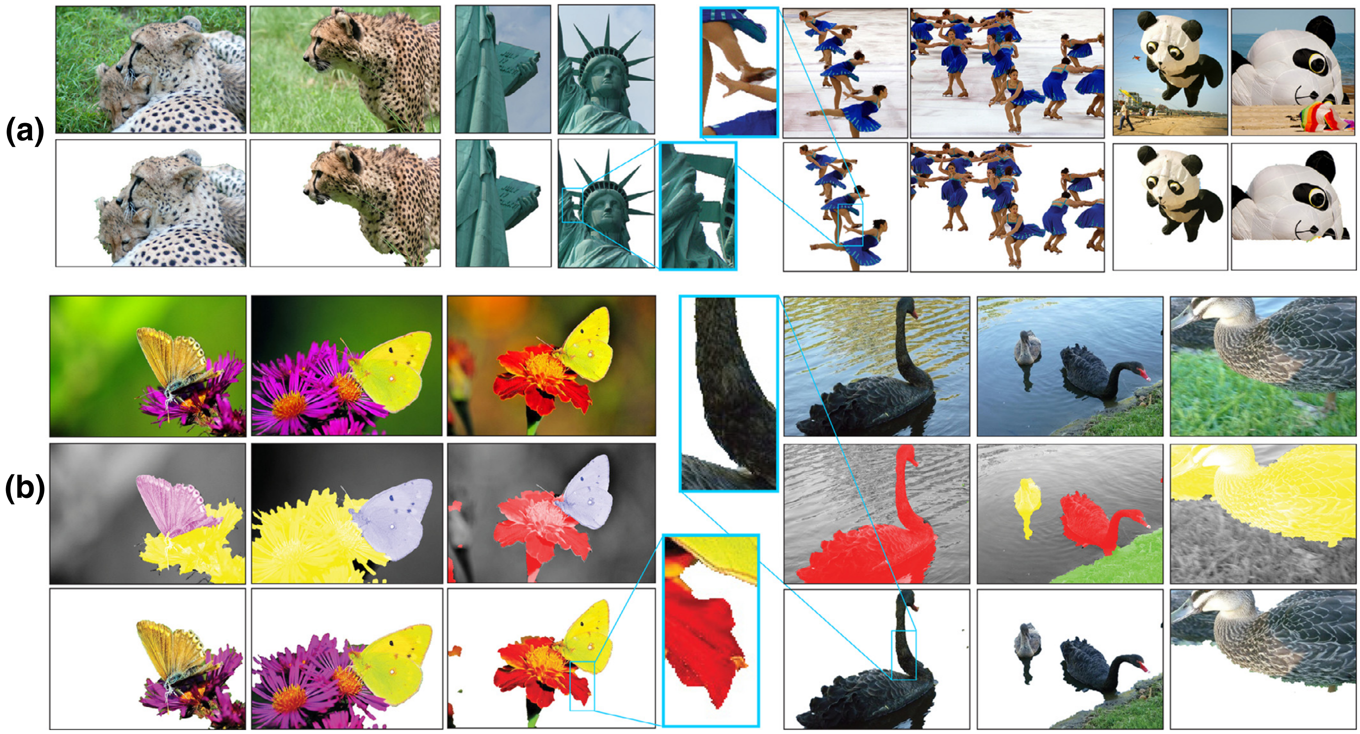

Fig. 7. Example cosegmentation results of (a) iCoseg and (b) FlickrMFC datasets. For iCoseg, we show pairs of the original images (up) and our cosegmentation outputs (bottom). For FlickrMFC which contains multiple foreground objects, we show the original inputs, extracted foreground objects, and the associated F/G outputs from top to bottom.

random initialization for GMM in the beginning of our MFC decomposition). We also observe that the resulting performance variance was less than 1.5%.

To investigate the contributions of each component in our proposed framework, we perform additional controlled experiments in Table 4, which verifies the roles of each proposed component. As shown this table, the full version of our proposed method achieved the highest Jaccard similarity of 44.2, while the uses of our framework without object discovery, iteration (i.e., one-pass only), F/G assignment, etc. stages only produced poorer results. Without handling F/G assignment (e.g., method of Seg + Match), the proposed method would degenerate into a multi-class cosegmentation algorithm, which resulted in a performance decrease of 7%. It is obvious that the only use of our initialization step (via GMM clustering) or the segmentation stage by GrabCut cannot provide satisfactory performance either (denoted as Initialization and Seg in Table 4, respectively).

We note that the use of RC-based saliency information [29] for performing F/G assignment on each individual image did not produce improved results. This verifies the use of our proposed strategy in Section 3.3 for F/G assignment across all input images. Nevertheless, Table 4 not only shows that our proposed method outperformed state-of-the-art cosegmentation approaches, it also supports our unique decomposition for improved MFC performance. Example cosegmentation results on both datasets are shown in Fig. 7.

5.3. Remarks on computation time

We now comment on the computation time of our proposed method. For the iCoset dataset, it took about 10.4 s to cosegment an image. Processing the image features (including feature extraction, image over-segmentation, and the calculation of contour probabilities) took about 9.1 s (which is 87.1% of the entire computation time), while our MFC decomposition only spent the remaining 1.3 s (i.e., 12.9% of the computation time). Recall that since only single foreground cosegmentation is considered for iCoseg, no further object discovery stage is required for cosegmentation of this dataset.

As for FlickrMFC, it took an average of 10.7 s for processing an image. In particular, the additional object discovery stage required about 0.3 s which was 3% of the processing time, while those for feature processing and MFC decomposition were 84% and 13%, respectively. The above runtime estimates were obtained by Matlab on an Intel Quad Core PC with 3.4 GHz with 16 GB memory.

To compare our computation time with those of other cosegmentation approaches, we applied the code released by [2,16]. Due to their memory requirements, we performed such comparisons on iCoseg using a workstation with Intel Quad Core processors of 2.4 GHz with 40 GB memory. While the average processing time of our method was less than 25 s per image, it required about 3 and 5 min per image for the approaches of [2] and [16], respectively. From the above observations, it can be concluded that our proposed MFC approach is computationally feasible.

6. Limitation and future works

The main challenge of MFC lies in the detection of multiple foreground objects from the input images, and the separation between them and the remaining backgrounds. While recent research attention has been focusing on the challenging setting of MFC, most of the existing works like [10,17–20,37] choose to evaluate their performance on natural images (e.g., outdoor images with different objects presented).

When it comes to perform MFC on images of indoor scenes, degraded performance will be expected. To further discuss this issue, we consider indoor scene images of the RGB-D object dataset [38]. We apply our proposed method and show example results in Fig. 8(a) and (b). It can be seen from Fig. 8(a) that while our approach was able to identify and distinguish between different foreground objects in

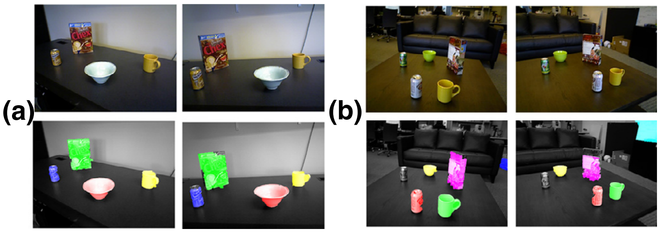

Fig. 8. Example cosegmentation results of RGB-D object dataset. (a) Successful MFC outputs and (b) results with missed and falsely detected errors. Note that only a pair of input images is considered in each, and the results in different colors denote the identified foreground objects. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

indoor scenes, we have one missed foreground object and two falsely identified foreground regions in Fig. 8(b). This is because that compared to objects in outdoor scenes, objects or background regions in indoor scene images are mostly artificial and typically exhibit significant appearance variations, especially if the images are taken from different views. As a result, the images of such scenes would contain regions with distinct color and textural information, which make the MFC problem even more difficult to solve. We believe that if MFC of indoor scene image will be of interest, one would require prior and sufficient knowledge (e.g., types and numbers of the objects of interest, training data, or image context information). Nevertheless, our work shows that our proposed method is able to perform favorably against state-of-the-art MFC methods on outdoor scene images in an unsupervised setting.

7. Conclusion

We presented an unsupervised MFC framework, which decomposes the original MFC problem into the tasks of segmentation, segment matching, and F/G assignment. Our proposed framework aims at solving and alternating between the above three tasks, so that image segments sharing similar visual appearances will not only be identified as foreground object parts, they will also be separated from undesirable background regions. Followed by an object discovery stage which utilizes the observed foreground object hypotheses across each image, the final objects of interest can be automatically extracted from the input image collection. Experiments on iCoseg and FlickrMFC datasets confirmed that our approach performs favorably against state-of-the-art cosegmentation methods on both single and multiple foreground cosegmentation problems.

Acknowledgment

This work is supported in part by the National Science Council, Taiwan via NSC102-3111-Y-001-015 and NSC102-2221-E-001-005-MY2.

References

[1] C. Rother, T.P. Minka, A. Blake, V. Kolmogorov, Cosegmentation of image pairs by histogram matching – incorporating a global constraint into MRFs, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2006.
[2] A. Joulin, F.R. Bach, J. Ponce, Discriminative clustering for image co-segmentation, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2010.
[3] S. Vicente, V. Kolmogorov, C. Rother, Cosegmentation revisited: models and optimization, in: Proceedings of European Conference on Computer Vision (ECCV), 2010.
[4] L. Mukherjee, V. Singh, J. Peng, Scale invariant cosegmentation for image groups, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2011.
[5] D. Batra, A. Kowdle, D. Parikh, J. Luo, T. Chen, iCoseg: interactive co-segmentation with intelligent scribble guidance, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2010.
[6] S. Vicente, C. Rother, V. Kolmogorov, Object cosegmentation, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2011.
[7] J.C. Rubio, J. Serrat, A.M. Lopez, N. Paragios, Unsupervised co-segmentation through region matching, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2012.
[8] F. Meng, H. Li, G. Liu, K.N. Ngan, Object co-segmentation based on shortest path algorithmand saliency model, IEEE Transactions on Multimedia 14 (5) (2012) 1429–1441.
[9] M. Rubinstein, A. Joulin, J. Kopf, C. Liu, Unsupervised joint object discovery and segmentation in Internet images, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2013.
[10] G. Kim, E.P. Xing, On multiple foreground cosegmentation, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2012.
[11] G. Kim, E. P. Xing, Jointly aligning and segmenting multiple web photo streams for the inference of collective photo storylines, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2013.
[12] W.-S. Chu, C.-P. Chen, C.-S. Chen, MOMI-cosegmentation: simultaneous segmentation of multiple objects among multiple images, in: Proceedings of Asian Conference on Computer Vision (ACCV), 2010.
[13] M.D. Collins, J. Xu, L. Grady, V. Singh, Random walks based multi-image segmentation: quasiconvexity results and GPU-based solutions, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2012.
[14] A. Faktor, M. Irani, Co-segmentation by composition, in: Proceedings of International Conference on Computer Vision (ICCV), 2013.
[15] G. Kim, E.P. Xing, F.-F. Li, T. Kanade, Distributed cosegmentation via submodular optimization on anisotropic diffusion, in: Proceedings of International Conference on Computer Vision (ICCV), 2011.
[16] A. Joulin, F. Bach, J. Ponce, Multi-class cosegmentation, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2012.
[17] T. Ma, L.J. Latecki, Graph transduction learning with connectivity constraints with application to multiple foreground cosegmentation, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2013.
[18] H. Zhu, J. Lu, J. Cai, J. Zheng, N. Magnenat-Thalmann, Multiple foreground recognition and cosegmentation: an object-oriented CRF model with robust higher-order potentials, in: Proceedings of Workshop on Applications of Computer Vision (WACV), 2014.
[19] F. Wang, Q. Huang, M. Ovsjanikov, L.J. Guibas, Unsupervised multi-class joint image segmentation, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
[20] H. Li, F. Meng, Q. Wu, B. Luo, Unsupervised multiclass region cosegmentation via ensemble clustering and energy minimization, IEEE Transactions on Circuits and Systems for Video Technology 24 (5) (2014) 789–801.
[21] A. Levinshtein, A. Stere, K.N. Kutulakos, D.J. Fleet, S.J. Dickinson, K. Siddiqi, Turbopixels: fast superpixels using geometric flows, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (12) (2009) 2290–2297.
[22] C. Rother, V. Kolmogorov, A. Blake, GrabCut: interactive foreground extraction using iterated graph cuts, ACM Transactions on Graphics (TOG) 23 (3) (2004) 309–314.
[23] D.R. Martin, C. Fowlkes, J. Malik, Learning to detect natural image boundaries using local brightness, color, and texture cues, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (5) (2004) 530–549.
[24] J. Carreira, C. Sminchisescu, CPMC: automatic object segmentation using constrained parametric min-cuts, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (7) (2012) 1312–1328.
[25] Y. Chen, A.B. Chan, G. Wang, Adaptive figure-ground classification, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2012.
[26] J. Shotton, J.M. Winn, C. Rother, 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) 2–23.
[27] Z. Yu, A. Li, O.C. Au, C. Xu, Bag of textons for image segmentation via soft clustering and convex shift, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2012.
[28] C. Yang, L. Zhang, H. Lu, X. Ruan, M.-H. Yang, Saliency detection via graph-based manifold ranking, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2013.
[29] M.-M. Cheng, G.-X. Zhang, N.J. Mitra, X. Huang, S.-M. Hu, Global contrast based salient region detection, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2011.
[30] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (11) (2001) 1222–1239.
[31] S. Vicente, V. Kolmogorov, C. Rother, Joint optimization of segmentation and appearance models, in: Proceedings of International Conference on Computer Vision (ICCV), 2009.
[32] Y. Boykov, V. Kolmogorov, An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (9) (2004) 1124–1137.
[33] V. Kolmogorov, R. Zabih, What energy functions can be minimized via graph cuts? IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (2) (2004) 147–159.
[34] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (5) (2002) 603–619.
[35] F. Wang, Q. Huang, L.J. Guibas, Image co-segmentation via consistent functional maps, in: Proceedings of International Conference on Computer Vision (ICCV), 2013.

[36] J. Dai, Y.N. Wu, J. Zhou, S.-C. Zhu, Cosegmentation and cosketch by unsupervised learning, in: Proceedings of International Conference on Computer Vision (ICCV), 2013.

[37] B.C. Russell, W.T. Freeman, A.A. Efros, J. Sivic, A. Zisserman, Using multiple segmentations to discover objects and their extent in image collections, in: Proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2006.

[38] K. Lai, L. Bo, X. Ren, D. Fox, A large-scale hierarchical multi-view RGB-D object dataset, in: Proceedings of International Conference on Robotics and Automation (ICRA), 2011.

Haw-Shiuan Chang received his B.S. degree in Electrical Engineering and Computer Science from National Chiao Tung University, Hsinchu, Taiwan in 2011. He was an exchange student in Electrical and Computer Engineering at Carnegie Mellon University in Fall 2010. He is currently a Research Assistant at Academia Sinica.

Yu-Chiang Frank Wang received the B.S. degree in Electrical Engineering from the National Taiwan University, Taipei, Taiwan in 2001. From 2001 to 2002, he was a Research Assistant with the National Health Research Institutes, Taiwan. He received the M.S. and Ph.D. degrees in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, USA, in 2004 and 2009, respectively.
Dr. Wang joined the Research Center for Information Technology Innovation (CITI), Academia Sinica, Taiwan, in 2009, where he currently holds the position as a tenure-track Associate Research Fellow. He leads the Multimedia and Machine Learning Laboratory, CITI, and works on research projects of computer vision, pattern recognition, machine learning, and image processing. He serves as an Organizing or Program Committee Member at multiple international conferences or activities, and several of his papers were nominated for the Best Paper Awards at related international conferences such as IEEE ICIP, IEEE ICME and IAPR MVA. In 2013, he was selected among the Outstanding Young Researchers by the National Science Council of Taiwan.

Please cite this article as: H.-S. Chang, Y.-C.F. Wang, Optimizing the decomposition for multiple foreground cosegmentation, Computer Vision and Image Understanding (2015), http://dx.doi.org/10.1016/j.cviu.2015.06.004