# Object-based Multiple Foreground Video Co-segmentation

Huazhu Fu, Dong Xu
Nanyang Technological University

Bao Zhang
Tianjin University

Stephen Lin
Microsoft Research

## Abstract

*We present a video co-segmentation method that uses category-independent object proposals as its basic element and can extract multiple foreground objects in a video set. The use of object elements overcomes limitations of low-level feature representations in separating complex foregrounds and backgrounds. We formulate object-based co-segmentation as a co-selection graph in which regions with foreground-like characteristics are favored while also accounting for intra-video and inter-video foreground coherence. To handle multiple foreground objects, we expand the co-selection graph model into a proposed multi-state selection graph model (MSG) that optimizes the segmentations of different objects jointly. This extension into the MSG can be applied not only to our co-selection graph, but also can be used to turn any standard graph model into a multi-state selection solution that can be optimized directly by the existing energy minimization techniques. Our experiments show that our object-based multiple foreground video co-segmentation method (ObMiC) compares well to related techniques on both single and multiple foreground cases.*

## 1. Introduction

The goal of video foreground co-segmentation is to jointly extract the main common object from a set of videos. In contrast to the unsupervised problem of foreground segmentation for a single video [16, 22, 23], the task of video co-segmentation is considered to be weakly supervised, since the presence of the foreground object in multiple videos provides some indication of what it is. Despite this additional information, there can still remain much ambiguity in the co-segmentation of general videos, which often contain multiple foreground objects and/or low contrast between foreground and background. Taking the pair of videos in Fig. 1 (a) as an example, it can be seen in (b) that co-segmentation methods based on low-level appearance features may not adequately discriminate between the foreground and background. Also, object-based methods designed for single video segmentation do not take advantage of the joint information between the videos, and consequently may extract different objects as shown in (c).

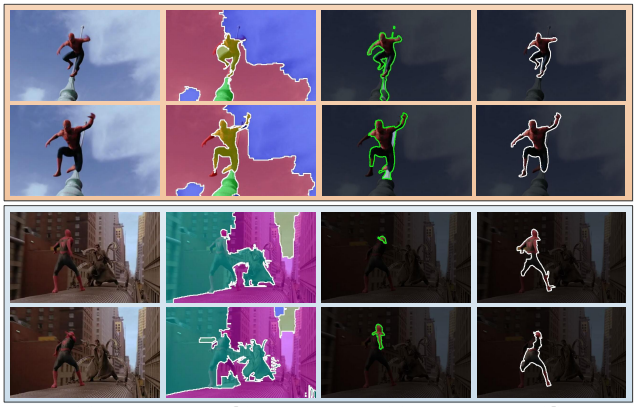


Figure 1. Video co-segmentation example for the case of a single foreground object. (a) Two related video clips. (b) Co-segmentation results from [3] based on low-level appearance features. (c) Results from object-based video segmentation [23] that does not consider the two videos jointly. (d) Results of our object-based video co-segmentation method.

In this paper, we present a general technique for video co-segmentation that is formulated with object proposals as the basic element of processing, and that can readily handle single or multiple foreground objects in single or multiple videos. Our **Ob**ject-based **M**ultiple foreground **Vi**deo **C**o-segmentation method (ObMiC) is developed from two main technical contributions. The first is an object-based framework in which a co-selection graph is constructed to connect each foreground candidate in multiple videos. The foreground candidates in each frame are category-independent object proposals that represent regions likely to encompass an object according to the structured learning method of [6]. This mid-level representation of regions has been shown to more robustly and meaningfully separate foreground and background regions in images and individual videos [21, 14, 16, 22, 23]. We introduce them into the video co-segmentation problem, and propose compatible constraints that assist in foreground identification and promote foreground consistency among the videos.

The second technical contribution is a method for extending the graph models such as the aforementioned co-selection graph to allow selection of multiple states in each

node. In the context of video co-segmentation, we apply this method to expand the co-selection graph into a **m**ulti-state **s**election **g**raph (MSG) in which multiple foreground objects can be dealt with in our object-based framework. The MSG is additionally able to accommodate the cases of a single foreground and/or a single video, and can be optimized by existing energy minimization techniques. Our ObMiC method yields co-segmentation results that surpass related techniques as shown in Fig. 1 (d). For evaluation of multiple foreground video co-segmentations, we have constructed a new dataset with ground truth, which will be made publicly available upon publication of this work.

## 2. Related Works

**Video Co-segmentation:** Only a few methods have been proposed for video co-segmentation, and they all base their processing on low-level features. Chen et al. [2] identified regions with coherent motion in the videos and then find a common foreground based on similar chroma and texture feature distributions. Rubio et al. [19] presented an iterative process for foreground/background separation based on feature matching among video frame regions and spatiotemporal tubes. The low-level appearance models in these methods, however, are often not discriminative enough to accurately distinguish complex foregrounds and backgrounds. Guo et al. [9] employed trajectory co-saliency to match the action form the video pair. However, this method only focuses on the common action extraction rather than the foreground object segmentation. In [3], the Bag-of-Words representation was used within a multi-class video co-segmentation method based on distant-dependent Chinese Restaurant Processes. While BoW features provide more discriminative ability than basic color and texture features, they may not be robust to appearance variations of a foreground object in different videos, due to factors such as pose change. Fig. 1 (b) shows co-segmentation results of [3], where the pixel-level features do not provide a representation sufficient for relating corresponding regions between the input videos. By contrast, our method uses an object-based representation that provides greater discriminability and robustness, as shown in Fig. 1 (d).

**Object-based Segmentation:** In contrast to the methods based on low-level descriptors, object-based techniques make use of a mid-level representation that aims to delineate an object's entirety. Vicente et al. [21] introduced the use of object proposals for co-segmentation of images. Meng et al. [17] employed the shortest path algorithm to select a common foreground from object proposals in multiple images. Lee et al. [14] utilized object proposal regions as foreground candidates in the context of single video segmentation, with the objectness measure used in ranking foreground hypotheses. More recent works [16, 22, 23] on single video segmentation have extended this object-based approach and incorporated a common constraint that the foreground should appear in every frame. This constraint is formulated within a weighted graph model, with the solution optimized via maximum weight cliques [16], shortest path algorithm [22], or dynamic programming [23]. As these single video segmentation methods do not address the co-segmentation problem, they do not account for the information within other videos. Moreover, they do not present a way to deal with multiple foreground objects. In our work, we present a more general co-selection graph to formulate correspondences between different videos, and extend this framework to handle both single and multiple foreground objects using the MSG model.

**Multiple foreground co-segmentation:** Some co-segmentation methods can handle multiple objects. Kim et al. [11] employed an anisotropic diffusion method to find out multiple object classes from multiple images. They also presented a different approach for multiple foreground co-segmentation in images [12], which builds on an iterative framework that alternates between foreground modeling and region assignment. Joulin et al. [10] proposed an energy-based image co-segmentation method that combines spectral and discriminative clustering terms. Mukherjee et al. [18] segmented multiple objects from image collections, by analyzing and exploiting their shared subspace structure. The video co-segmentation method in [3] can also deal with multiple foreground extraction, which uses a non-parametric Bayesian model to learn a global appearance model that connects the segments of the same class. However, all of these methods are based on low-level feature representations for clustering the foregrounds into classes. On the other hand, object-based techniques operate on a mid-level representation of object proposals but lack an effective way to deal with multiple foregrounds. In our work, we extend the object-based co-segmentation approach to handle multiple foregrounds using the MSG model, where multiple foreground objects can be segmented jointly in multiple videos via the existing energy minimization method.

## 3. Our Approach

We present our object-based video co-segmentation algorithm by first describing it for the case of a single foreground object, and then extending this approach to handle multiple foreground objects using the MSG model.

### 3.1. Single object co-segmentation

We denote the set of videos as $\{V^1, ..., V^N\}$, where each video $V^n$ consists of $T_n$ frames denoted by $\{F_1^n, ..., F_{T_n}^n\}$. In each frame, a set of object-based candidates is obtained using the category-independent object proposals method [6], from which the generated candidates may possibly have some overlapping areas. To identify the foreground object in each frame, we consider various object characteristic-

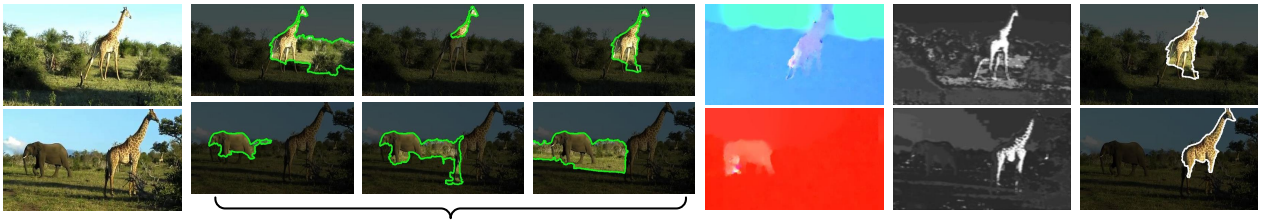


Figure 2. Unary energy factors in foreground detection. (a) Two input frames from a video. (b) Top three proposal regions generated from [6], where the candidates are ranked by their objectness scores. (c) Optical flow maps by [15], which detects dynamic objects and ignores static objects. (d) Co-saliency maps by [8], which indicate common salient regions among the video. (e) Our selected candidates determined from the co-selection graph, which extracts the common foreground (giraffe) and removes background objects (elephant).

s that are indicative of foregrounds, while accounting for intra-video coherence of the foreground as well as foreground coherence among the different videos.

We formulate this problem as a co-selection graph in the form of a conditional random field (CRF). As illustrated in Fig. 3, each video is modeled by a sequence of nodes. Each node represents a frame in the video, and the possible states of a node are comprised of the foreground object candidates in the frame. For the case of a single foreground object, we seek for each node the selected candidate $u_t^n$ that corresponds to it. By concatenating the selected candidates from all the frames of the video set, we obtain a *candidate series* $\mathbf{u} = \{u_t^n | n = 1, ..., N; t = 1, ..., T_n\}$. For each video, *intra-video* edges are placed between the nodes of adjacent frames. The nodes of different videos are fully connected with each other by *inter-video* edges.

For this co-selection graph, we express its energy function $E_{cs}(\mathbf{u})$ as follows:

$$E_{cs}(\mathbf{u}) = \sum_{n=1}^{N}\sum_{t=1}^{T_n}\left[\Psi(u_t^n) + \Phi_\alpha(u_t^n, u_{t+1}^n)\right] + \sum_{\substack{n,m=1,\\ n\neq m}}^{N}\sum_{t=1}^{T_n}\sum_{s=1}^{T_m}\Phi_\beta(u_t^n, u_s^m), \quad (1)$$

where $\Psi$, $\Phi_\alpha$ and $\Phi_\beta$ represent unary, intra-video and inter-video energy, respectively.

**Unary energy** combines three factors in determining how likely an object candidate is to be the foreground:

$$\Psi(u_t^n) = -\log\left[O(u_t^n)\cdot\max\left(M(u_t^n), S(u_t^n)\right)\right]. \quad (2)$$

The factors that influence this energy are the objectness score $O(u)$, motion score $M(u)$, and saliency score $S(u)$ of the candidate $u$. The objectness score $O(u)$ provides a measure of how likely the candidate is to be a whole object. For this, we take the value returned in the candidate generation process [6]. The motion score $M(u)$ measures the confidence that candidate $u$ corresponds to a coherently moving object in the video. We define the motion score using the definition in [14]:

$$M(u_t^n) = 1 - \exp\left(-\frac{1}{M_m}\chi^2_{flow}(u_t^n, \bar{u}_t^n)\right), \quad (3)$$

where $\bar{u}_t^n$ denotes the pixels around the candidate $u_t^n$ within a minimum bounding box enclosing the candidate, and $\chi^2_{flow}$ is the $\chi^2$-distance between the normalized optical flow histograms with $M_m$ denoting the mean of the $\chi^2$-distances. In our work, the optical flow is computed by using the method in [15].

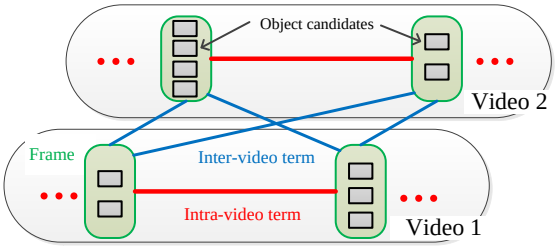


Figure 3. Our co-selection graph is formulated as a CRF model. Each frame of a video is a node, and the foreground object candidates of the frame are the states a node can take. The nodes (frames) from different videos are fully connected by inter-video terms. Within a given video, only adjacent nodes (frames) are connected by intra-video terms.

Most video segmentation methods [14, 16, 22, 23] aim to find a coherently moving foreground object based on its motion score. However, in practice a foreground object may not always be moving in the video, so we additionally consider a static saliency cue and take the maximum between the dynamic motion and static saliency cues in Eq. (2). Different from the objectness score $O(u)$, which is designed to identify extracted regions that are object-like and whole, the saliency score $S(u)$ relates to visually salient stimuli, which has often been used to find regions of interest. For this, rather than performing saliency detection for single images, we compute the co-saliency map on multiple images as described in [8], which takes consistency throughout the video into account.

The differences among the three factors in the unary term are illustrated in Fig. 2. For the input frames in (a), the top proposals ranked only by objectness scores [6] do not

accurately represent the foreground in the video. For example, the actual foreground in the first frame of (b) is ranked third, while in the second frame the correct foreground object is not even among the top three. On the other hand, the optical flow map of [15] in (c) highlights the primary object (giraffe) in the first frame, but instead finds the secondary object (elephant) in the second frame due to its higher motion score. The co-saliency map of [8] in (d) detects a common foreground, but may also give high scores to other regions. Jointly considering these disparate factors leads to more reliable estimates of the foreground, as shown in (e).

**Intra-video energy** provides a spatiotemporal smoothness constraint between neighboring frames in an individual video. It is commonly used in single video segmentation [16, 22, 23], and we define this term as follows:

$$\Phi_\alpha(u_t^n, u_{t+1}^n) = \gamma_1 \cdot D_c(u_t^n, u_{t+1}^n) \cdot D_f(u_t^n, u_{t+1}^n), \quad (4)$$

where $\gamma_1$ is a weighting coefficient, $D_c$ represents the color histogram similarity between two candidates as

$$D_c(u_t^n, u_{t+1}^n) = \frac{1}{M_c}\chi^2_{color}(u_t^n, u_{t+1}^n), \quad (5)$$

where $\chi^2_{color}$ is the $\chi^2$-distance between unnormalized color histograms with $M_c$ denoting the mean of the $\chi^2$-distances among all candidates in all the videos, and $D_f$ represents the overlap between the two candidates in the adjacent frames:

$$D_f(u_t^n, u_{t+1}^n) = -\log\left(\frac{|u_t^n \cap Warp(u_{t+1}^n)|}{|u_t^n \cup Warp(u_{t+1}^n)|}\right), \quad (6)$$

where $Warp(u_{t+1}^n)$ transforms the candidate region $u_{t+1}^n$ from frame $t+1$ to $t$ based on optical flow mapping [15].

**Inter-video energy** measures foreground consistency among the different videos. In the co-selection graph, candidates from one video are connected to those in the other videos. We define the inter-video energy as follows:

$$\Phi_\beta(u_t^n, u_s^m) = \gamma_2 \cdot D_c(u_t^n, u_s^m) \cdot D_s(u_t^n, u_s^m), \quad (7)$$

where $\gamma_2$ is a weighting coefficient, $D_c$ denotes color histogram similarity computed as in Eq. (5), and $D_s$ measures shape similarity between the two candidates. In our work, shape is represented in terms of the HOG descriptor [4] within a minimum bounding box enclosing the candidate. We define $D_s$ as

$$D_s(u_t^n, u_s^m) = \frac{1}{M_s}\chi^2_{shape}(u_t^n, u_s^m), \quad (8)$$

where $\chi^2_{shape}$ is the $\chi^2$-distance between unnormalized HOGs with $M_s$ denoting the mean of the $\chi^2$-distances.

**Inference:** To solve the co-selection graph, we seek the labeling $\mathbf{u}^*$ that minimizes its energy function:

$$\mathbf{u}^* = \arg\min_{\mathbf{u}} E_{cs}(\mathbf{u}). \quad (9)$$

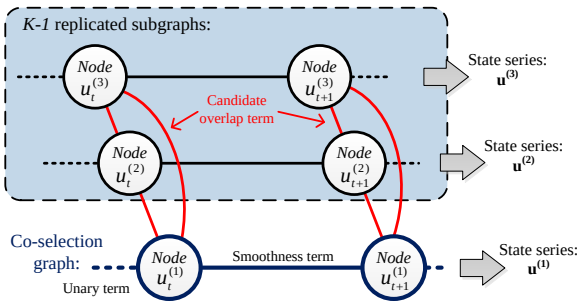


Figure 4. Our MSG model, illustrated for $K = 3$. For $K$-state selection, our method replicates the co-selection graph $K-1$ times to form $K$ subgraphs, and connects the corresponding nodes of different subgraphs with the candidate overlap term. Each subgraph outputs its corresponding candidate series. The smoothness terms include the inter-video terms and intra-video terms in our co-selection graph.

In contrast to the directed graph used in [22, 23], our co-selection graph is a cycle graph that connects candidates among multiple videos. Optimizing a cycle graph is a NP-hard problem. We employ TRW-S [13] to obtain a good approximated solution as in [5].

Since object candidates generated by [6] are only roughly segmented, we refine the results as in [14, 23] with a pixel-level spatiotemporal graph-based segmentation.

## 3.2. Multiple foreground co-segmentation

In this section, we extend our single object video co-segmentation approach to handle multiple foregrounds using a multi-state selection graph model (MSG). With MSG, multiple foregrounds can be solved jointly in the multiple videos via existing energy minimization methods.

### 3.2.1 Multiple foreground selection energy

For the case of multiple foregrounds, $K$ different candidates are to be found in each frame. We refer to the set of selected candidates throughout the videos for the $k^{th}$ foreground object as the candidate series $\mathbf{u}^{(k)}$. In solving for the multiple foreground co-segmentation, we account for the independent co-segmentation energies $E_{cs}(\mathbf{u}^{(k)})$ of each of the $K$ candidate series. In addition, it must be ensured that the $K$ candidate regions have minimal overlap throughout the videos, since an area in a video frame cannot belong to two or more foreground objects. We model this constraint by introducing the *candidate overlap penalty* $E_{ov}(\mathbf{u}^{(k)}, \mathbf{u}^{(j)})$ between different candidate series, and define the multiple foreground selection problem as follows:

**Definition:** *Let $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ be an undirected graph with the set of vertices $\mathcal{V}$ and the set of edges $\mathcal{E}$. By concatenating the variables from all the nodes, we obtain a candidate series $\boldsymbol{u}$. The multiple foreground selection solves for $K$*

*different candidate series* $\{\boldsymbol{u}^{(1)}, ..., \boldsymbol{u}^{(K)}\}$ *in* $\mathcal{G}$ *according to*

$$\min_{\boldsymbol{u}^{(1)},...,\boldsymbol{u}^{(K)}} \sum_{k=1}^{K} E_{cs}(\boldsymbol{u}^{(k)}) + \sum_{k,j=1}^{K} E_{ov}(\boldsymbol{u}^{(k)}, \boldsymbol{u}^{(j)}), \quad (10)$$

*where* $E_{cs}(\cdot)$ *denotes independent co-selection graph energies and* $E_{ov}(\cdot,\cdot)$ *represents the candidate overlap penalty.*

Incorporating Eq. (1) into the multiple foreground selection energy function in Eq. (10), we obtain

$$\begin{aligned} E_{msg} &= \sum_{k=1}^{K} E_{cs}(\mathbf{u}^{(k)}) + \sum_{k,j=1}^{K} E_{ov}(\mathbf{u}^{(k)}, \mathbf{u}^{(j)}) \\ &= \sum_{k=1}^{K} \left\{ \begin{array}{l} \sum_{n=1}^{N} \sum_{t=1}^{T_n} \left[ \Psi(u_t^{n,(k)}) + \Phi_\alpha(u_t^{n,(k)}, u_{t+1}^{n,(k)}) \right] \\ \quad + \sum_{\substack{n,m=1,\\ n\neq m}}^{N} \sum_{t=1}^{T_n} \sum_{s=1}^{T_m} \Phi_\beta(u_t^{n,(k)}, u_s^{m,(k)}) \end{array} \right\} \\ &\quad + \sum_{\substack{k,j=1\\ k\neq j}}^{K} \sum_{n=1}^{N} \sum_{t=1}^{T_n} \Delta(u_t^{n,(k)}, u_t^{n,(j)}), \end{aligned} \quad (11)$$

where $u_t^{n,(k)}$ denotes the $k^{th}$ selected candidate in frame $F_t^n$, and $\Delta(\cdot,\cdot)$ is the candidate overlap term. In our co-segmentation method, the candidate overlap term is defined as the intersection-over-union metric between two candidates:

$$\Delta(u_t^{n,(k)}, u_t^{n,(j)}) = \gamma_3 \frac{|u_t^{n,(k)} \cap u_t^{n,(j)}|}{|u_t^{n,(k)} \cup u_t^{n,(j)}|}, \quad (12)$$

where $\gamma_3$ is a scale parameter.

#### 3.2.2 Multi-state selection graph model

To optimize the multiple foreground selection energy in Eq. (11), we propose the multi-state selection graph model (MSG). In MSG, the co-selection graph for single object co-segmentation is replicated $K-1$ times to produce $K$ subgraphs in total, one for each candidate series. We observe that the candidate overlap penalty $\Delta(\cdot,\cdot)$ in Eq. (11) can be treated as edges between corresponding nodes in the subgraphs, as illustrated in Fig. 4. Linking the subgraphs in this way combines the subgraphs into a unified MSG, such that the single foreground co-selection graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ is extended into the multi-state selection graph $\mathcal{G}' = \langle \mathcal{V}', \mathcal{E}' \rangle$, where the vertex set $\mathcal{V}'$ is composed of the vertices from the $K$ subgraphs, and edge set $\mathcal{E}'$ includes the edges in the $K$ subgraphs as well as the edges between the subgraphs for the candidate overlap term.

With the MSG model, we can express the multiple foreground selection energy of Eq. (11) as follows:

$$\begin{aligned} E_{msg} &= \sum_{k=1}^{K} \left\{ \sum_{q\in\mathcal{V}} \Psi(u^q) + \sum_{(q,r)\in\mathcal{E}} \Phi(u^q, u^r) \right\} \\ &\quad + \sum_{(q,r)\in\mathcal{V}_\Delta} \Delta(u^q, u^r) \quad (13) \\ &= \sum_{q\in\mathcal{V}'} \Psi(u^q) + \sum_{(q,r)\in\mathcal{E}'} \Theta(u^q, u^r), \quad (14) \end{aligned}$$

where $(q,r)$ denotes the edge between nodes $q$ and $r$, $\mathcal{V}_\Delta$ denotes the edge set for the candidate overlap term in multi-state selection, and $\Theta$ is the combination of the smoothness term $\Phi$ and the candidate overlap term $\Delta$. Note that $\Phi(\cdot)$ in Eq. (13) encompasses the intra-video terms $\Phi_\alpha(\cdot)$ and inter-video terms $\Phi_\beta(\cdot)$ in Eq. (11).

Our MSG energy in Eq. (14) can be derived in the context of Markov Random Fields: A minimum of $E_{msg}$ corresponds to a *maximum a posteriori* (MAP) labeling $\{\mathbf{u}^{(1)}, ..., \mathbf{u}^{(K)}\}$. Thus, our MSG can be solved directly by the existing energy minimization method (e.g., A$^*$ search [1] and belief propagation [7]), yielding the multiple foreground objects in one shot. Moreover, our MSG can be applied not only to extend our co-selection graph, but also to turn any standard graph model into a multi-state selection solution. In this paper, we employ TRW-S [13] to obtain the approximated solution.

## 4. Experiments

The ObMiC method is general enough to handle single/multiple videos and single/multiple foreground segmentation. In our experiments, we test our method in the two video co-segmentation cases, with a single foreground and with multiple foregrounds. We employ two metrics for the evaluation. The first is the average per-frame pixel error [20] defined as $\frac{|\mathbf{XOR}(R,GT)|}{F}$, where $R$ is the segmentation result of each method, $GT$ is the ground truth, and $F$ is the total number of frames. The second measure is the *intersection-over-union metric* [3] defined as $\frac{R \cap GT}{R \cup GT}$.

### 4.1. Single foreground video co-segmentation

In evaluating for the single foreground case, we employ the MOViCS dataset [3], which includes four video sets in total with five frames of each video labeled with the ground truth. The foregrounds in these video sets are taken to be the primary objects, namely the Chicken, Giraffe, Lion and Tiger. Using the codes obtained from the corresponding authors, we compare our ObMiC algorithm to six state-of-the-art methods that are the most closely related works published in recent years: (1) Co-saliency detection (CoSal) in [8], which is based on bottom-up saliency cues and employs a global coherence cue to detect the common saliency region in multiple images. CoSal [8] can also produce the co-segmentation results via a binary segmentation.

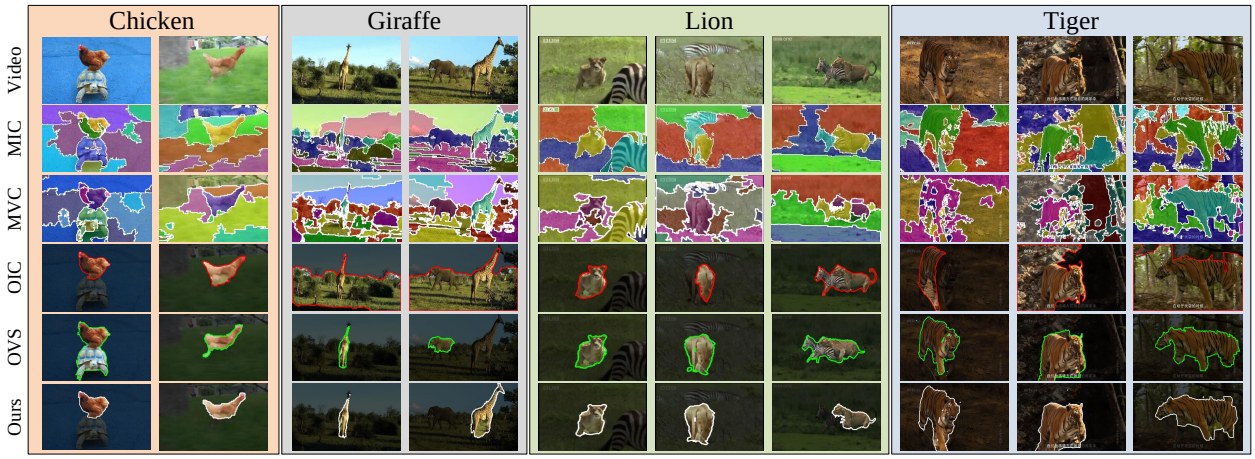


Figure 5. Single object segmentation results on the MOViCS dataset, where the displayed video frames are from different videos. From top to bottom: input videos, MIC [10], MVC [3], OIC [17], OVS [23], and our ObMiC method. (Best viewed in color.)

| Methods | Chicken | Giraffe | Lion | Tiger | Avg. |
|---|---|---|---|---|---|
| CoSal [8] | 6092 | 5791 | 8007 | 53253 | 18284 |
| ObjPro [6] | 13624 | 8917 | 5243 | 56743 | 21132 |
| OIC [17] | 3107 | 69001 | 9534 | 82303 | 40986 |
| OVS [23] | 5579 | 23735 | 7853 | 24200 | 15342 |
| MIC [10] | 7771 | 4053 | 4067 | 44809 | 15175 |
| MVC [3] | 3985 | 3244 | 3181 | 34352 | 11191 |
| Our SeC | 2450 | 3953 | 3058 | 24147 | 8402 |
| Our ObMiC | **1567** | **2938** | **1598** | **21005** | **6726** |

Table 1. The average per-frame pixel errors on MOViCS dataset.

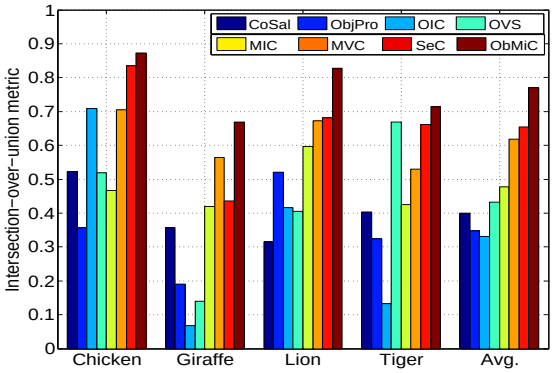


Figure 6. The intersection-over-union metric on MOViCS dataset.

(2) Object-based proposals (ObjPro) in [6], which generates a set of object candidates in each frame based on a category independent generic object model. We use the top-ranked proposals as the result in [6]. (3) Object-based image co-segmentation (OIC) in [17], which selects a common object from the multiple images via the shortest path algorithm. (4) Object-based video segmentation (OVS) in [23], which employs a directed acyclic graph based framework to select the primary object in a single video. (5) Multi-class image co-segmentation (MIC) in [10], which segments the multiple images into regions of multiple classes. We select the class that has the most overlap with the ground truth over the video set as its foreground segmentation result. Since the number of clusters $K$ needs to be predefined in MIC, we sample values of $K$ between 5 and 8, and choose the value that yields the best performance for each video set. (6) Multi-class video co-segmentation (MVC) in [3], which produces a segmentation of multiple classes from the multiple videos. As with MIC [10], we select the class that has the most overlap with the ground truth over the entire video set as its segmentation result. (7) We also present an intermediate result of our method: the selected candidates (SeC) from Eq. (1), i.e., our ObMiC results prior to pixel-level refinement. The segmentation results are shown in Fig. 5, and quantitative errors are given in Table 1 and Fig. 6.

ObjPro [6] does not perform well, because it lacks intra-video and inter-video constraints. Slightly better performance is obtained by CoSal [8], as it employs inter-video cues. However, the bottom-up saliency cue used in [8] can become less effective in complex videos (e.g., Tiger), as also mentioned in [8]. Our SeC integrates objectness, motion, and co-saliency related cues in the unary term, which together with the additional inter-video and intra-video constraints leads to significant improvements over ObjPro and CoSal on average.

OIC [17] is an object-based co-segmentation method for multiple images. Image co-segmentation methods often make use of the assumption that the multiple images have different backgrounds. However, the backgrounds of a video are temporally continuous and similar in content, which leads to incorrect foregrounds from OIC as seen in the videos of Giraffe and Tiger. In video segmentation methods, the use of motion cues and intra-video smoothness provides powerful constraints that help to avoid this issue.

OVS [23], which is designed for single video segmentation, extracts the foreground without considering the other videos. As a result, the segmented foreground might not be the same among the videos in the set. For example, it extracts both the lion and zebra together as the fore-

ground region in the third video of Lion, since they both have foreground-like characteristics and appear connected through most of the video.

MIC [10] combines local appearance and spatial consistency terms with class-level discrimination. However, as an image co-segmentation method, it does not include a temporal smoothness constraint for video co-segmentation. The low-level representation in MIC without an objectness constraint may lead to fragmentary segmentation, as seen in the second video of Lion and the third video of Tiger.

Inter-video constraints are incorporated in MVC [3]. However, its segmentation with pixel-level features often does not capture the foreground object in its entirety, as shown for the videos of Chicken and Tiger. The use of pixel-level features can also affect its class labeling, as seen in the third video of Tiger, though this problem is not penalized in this comparison since the region that has the maximum overlap with the ground truth is taken as the segmentation result of MVC. By contrast, the use of objectness and intra-video smoothness constraints in our ObMiC method helps to avoid these issues and provides more meaningful foreground co-segmentation results. ObMiC obtains the best results on the four video sets.

### 4.2. Multiple foreground video co-segmentation

Since there are no datasets for multiple foreground video co-segmentation, we have collected our own, consisting of four sets, each with a video pair and two foreground objects in common. The dataset includes ground truth manually obtained for each frame. With these videos, we compare our method to two multi-class co-segmentation methods: MIC [10] and MVC [3]. We also provide two other baselines: selected candidates via our MSG, and iterative selection (IterSel) which solves for the foreground objects one at a time from Eq. (11). IterSel first computes one candidate series based on single object co-selection, then updates the unary term of each node by adding the candidate overlay term in Eq. (12) for the selected candidate to prevent re-selection of its associated states in subsequent iterations. These two steps are repeated until $K$ state series are selected. The total energy function of IterSel thus becomes equivalent to Eq. (11) after selecting all the state series. For most object-based segmentation methods, the number of foregrounds (i.e., $K$) needs to be predefined. In this experiment, we set $K = 2$. Fig. 8 displays multiple foreground segmentation results with our dataset, and quantitative errors are given in Table 2 and Fig. 7.

MIC [10] employs a global constraint to group similar regions from different images. It also classifies pixels based on a low-level representation without an objectness constraint, which may result in wrongly merged object classes from the foreground and background. For example, the black dog in the first video of the Dog set is wrongly classified together with the background tree shadows in the second video. Also, for the complex foreground (e.g, the bigger monster) in the Monster set, MIC produces a fragmentary segmentation from the low-level features.

| Methods | Dog | Person | Monster | Skating | Avg. |
|---|---|---|---|---|---|
| MVC [3] | 1807 | 10389 | 7394 | 10223 | 7453 |
| MIC [10] | 4794 | 11033 | 7836 | 26616 | 12570 |
| IterSel | 1527 | 12482 | 6631 | 3537 | 6044 |
| Our MSG | 1209 | 12120 | 5699 | 3455 | 5621 |
| Our ObMiC | **1115** | **9321** | **3551** | **3274** | **4315** |

Table 2. The average per-frame pixel errors on our multiple foreground video dataset.

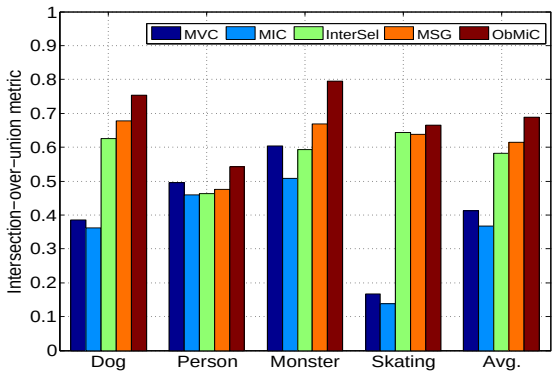


Figure 7. The intersection-over-union metric on our multiple foreground video dataset.

MVC [3] includes a temporal smoothness constraint and obtains better performance than MIC. However, as with the single foreground segmentation, the pixel-level processing of MVC leads to some errors in class labeling and hence to some incorrect correspondence of objects (e.g., the changing of the bigger monster classes in the Monster set). Even though our comparison does not penalize these class switches in MVC (by taking the region with the maximum overlap with the ground truth), our ObMiC still outperforms both MVC and MIC on all the videos.

Since IterSel and MSG both generate results directly from the object proposals of [6] without using pixel-level refinement, their segmentation results are coarse and have greater error than those with pixel-level segmentation. IterSel is similar to a greedy process that sequentially obtains a local optimum for each candidate series. By contrast, our MSG method optimizes this multi-state problem jointly via a single global energy function, which leads to less error than IterSel (see Table 2).

We note that our method assumes the existence of a common object proposal among the videos, which is a standard assumption among object-based co-segmentation methods (e.g., [17, 21]). When common objects exist, but not in all the videos, our method can still extract them, but will also extract an unrelated region in videos where the common object is missing. How to deal with missing common objects is a direction for future work.


Figure 8. Segmentation results on our newly collected multiple foreground video dataset, where different videos in a set are separated by a line. From top to bottom: input videos, MIC [10], MVC [3], and our ObMiC method. (Best viewed in color.)

## 5. Conclusion

We proposed an object-based multiple foreground video co-segmentation method, whose key components are the use of object proposals as the basic element of processing, with a corresponding co-selection graph that places constraints among objects in the videos, and the multi-state selection graph for addressing the problem of multiple foreground objects. Our MSG, which can handle single/multiple videos with single/multiple foregrounds, provides a general and global framework that can be used to extend any standard graph model to handle multi-state selection while still allowing optimization by existing energy minimization techniques.

**Acknowledgements:** This work is supported by the Singapore A*STAR SERC Grant (112-148-0003).

## References

[1] M. Bergtholdt, J. Kappes, S. Schmidt, and C. Schnorr. A study of parts-based object class detection using complete graphs. *IJCV*, 2010.

[2] D. Chen, H. Chen, and L. Chang. Video object cosegmentation. In *ACM MM*, 2012.

[3] W. Chiu and M. Fritz. Multi-class video co-segmentation with a generative multi-video model. In *CVPR*, 2013.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[5] T. Deselaers, B. Alexe, and V. Ferrari. Weakly supervised localization and learning with generic knowledge. *IJCV*, 2012.

[6] I. Endres and D. Hoiem. Category independent object proposals. In *ECCV*, 2010.

[7] P. Felzenszwalb and D. Huttenlocher. Efficient belief propagation for early vision. *IJCV*, 2006.

[8] H. Fu, X. Cao, and Z. Tu. Cluster-based co-saliency detection. *TIP*, 2013.

[9] J. Guo, Z. Li, L. Cheong, and S. Zhou. Video co-segmentation for meaningful action extraction. In *ICCV*, 2013.

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

[11] G. Kim, E. Xing, L. Fei-Fei, and T. Kanade. Distributed cosegmentation via submodular optimization on anisotropic diffusion. In *ICCV*, 2011.

[12] G. Kim and E. P. Xing. On multiple foreground cosegmentation. In *CVPR*, 2012.

[13] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *TPAMI*, 2006.

[14] Y. Lee, J. Kim, and K. Grauman. Key-segments for video object segmentation. In *ICCV*, 2011.

[15] C. Liu. *Beyond pixels: exploring new representations and applications for motion analysis*. PhD thesis, MIT, 2009.

[16] T. Ma and L. Latecki. Maximum weight cliques with mutex constraints for video object segmentation. In *CVPR*, 2012.

[17] F. Meng, H. Li, G. Liu, and K. Ngan. Object co-segmentation based on shortest path algorithm and saliency model. *TMM*, 2012.

[18] L. Mukherjee, V. Singh, J. Xu, and M. D.Collins. Analyzing the subspace structure of related images: Concurrent segmentation of image sets. In *ECCV*, 2012.

[19] J. Rubio, J. Serrat, and A. López. Video co-segmentation. In *ACCV*, 2012.

[20] D. Tsai, M. Flagg, and J. Rehg. Motion coherent tracking with multi-label MRF optimization. In *BMVC*, 2010.

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

[22] B. Zhang, H. Zhao, and X. Cao. Video object segmentation with shortest path. In *ACM MM*, 2012.

[23] D. Zhang, O. Javed, and M. Shah. Video object segmentation through spatially accurate and temporally dense extraction of primary object regions. In *CVPR*, 2013.