Video Object Co-segmentation by Regulated Maximum Weight Cliques

Dong Zhang[1], Omar Javed[2], and Mubarak Shah[1]

[1] Center for Research in Computer Vision, UCF, Orlando, FL 32826
[2] SRI International, Princeton, NJ 08540
{dzhang,shah}@eecs.ucf.edu, omar.javed@sri.com

Abstract. In this paper, we propose a novel approach for object co-segmentation in arbitrary videos by sampling, tracking and matching object proposals via a Regulated Maximum Weight Clique (RMWC) extraction scheme. The proposed approach is able to achieve good segmentation results by pruning away noisy segments in each video through selection of object proposal tracklets that are spatially salient and temporally consistent, and by iteratively extracting weighted groupings of objects with similar shape and appearance (with-in and across videos). The object regions obtained from the video sets are used to initialize per-pixel segmentation to get the final co-segmentation results. Our approach is general in the sense that it can handle multiple objects, temporary occlusions, and objects going in and out of view. Additionally, it makes no prior assumption on the commonality of objects in the video collection. The proposed method is evaluated on publicly available multi-class video object co-segmentation dataset and demonstrates improved performance compared to the state-of-the-art methods.

Keywords: Video Segmentation, Cosegmentation.

1 Introduction and Related Work

Our goal is to discover and segment objects from a video collection in an unsupervised manner. We compensate for the lack of supervision by exploiting commonality of objects in video collection (if it exists) to build better object segmentation models. Unsupervised segmentation of object classes from video collection has applications in large scale video tagging and retrieval, generation of training sets for supervised learning, and forensic video analysis.

Video co-segmentation is a natural extension of image co-segmentation, for which there is a large body of prior work ([20,12,18]). Image co-segmentation was introduced by Rother et al. [20]. Later, image based object co-segmentation was proposed by [26], which introduced the use of object proposals ([5,1]) for segmenting similar object from image pairs. The co-segmentation idea was extended to handle multiple object classes ([13]) and to work in more general internet collections where all images did not share the same object [21].

There is a large body of prior work on single video segmentation techniques ([10,9,19,27]). In general these techniques use appearance information of the

D. Fleet et al. (Eds.): ECCV 2014, Part VII, LNCS 8695, pp. 551–566, 2014.

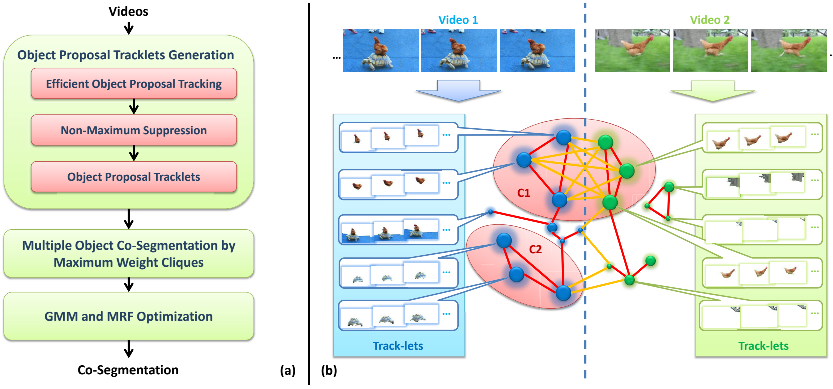
Videos
Object Proposal Tracklets Generation
Efficient Object Proposal Tracking
Non-Maximum Suppression
Object Proposal Tracklets
Multiple Object Co-Segmentation by Maximum Weight Cliques
GMM and MRF Optimization
Co-Segmentation
(a)
(b)
Video 1
Video 2
C1
C2
Track-lets
Track-lets

Fig. 1. (a) Shows the framework of the proposed method. (b) Shows the formulation of the Regulated Maximum Weight Cliques (RMWC) for object proposal tracklets. In this example, we generate 'object proposal' tracklets for two videos and use weighted nodes to represent them. Edges are built between similar nodes, and there are two types of edges (intra-video edges: red; inter-video edges: orange). In this example, the first two maximal cliques which have highest weights are obtained ($C1$ and $C2$). $C1$ contains all the segments for 'chicken' from two videos and $C2$ contains all the segments for 'turtle' in video 1.

videos to group the pixels in a spatio-temporal graph and/or employ motion segmentation techniques to separate objects by using motion cues. There are also several methods ([16,17,29]) designed for primary video object segmentation in single videos through use of the 'objectness' ([1]) measure. However, all of these methods use information extracted from one video for its segmentation and there is no cross-video knowledge transfer.

A related problem of object discovery has been studied in computer vision for images ([14,23,24]) and videos ([28,30]). These methods either use graph based clustering of low-level image features or generative models, such as Latent Dirichlet Allocation (LDA), to learn the distribution of class of image regions based on visual words.

Recently, a few methods have been proposed for video co-segmentation ([3,22,4]). The method in [3] attempts to segment the common region from a pair of videos and model the problem as a common foreground and background separation. It represents the video pair by super-voxels and proposes a motion-based video grouping method in order to find common foreground regions. It employs Gaussian mixture models to characterize the common object appearance. The work by Rubio et al. ([22]) aims at segmenting the same object (or objects belonging to the same class) moving in a similar manner from two or more videos. The method starts with grouping the pixels in video frames at two levels: the higher levels consists of space-time tubes and the lower level consists of within frame region segments; an initial foreground and background labeling is generated to construct the probabilistic distribution of the feature vectors of tubes and regions; and a probabilistic framework is employed to get the final co-segmentation results. Both [3] and [22] use strong assumptions of a single class of object common to all videos. Chiu and Fritz ([4]) proposed multi-class video object co-segmentation and also provided a publicly available dataset (MOViCS) with ground truth. In this work, a non-parametric Bayesian model for co-segmentation is used, which is based on a video segmentation prior. This method does not use restrictive assumptions on the number of object classes per frame or requirements on commonality of object in all videos. However, the method groups dense image patches to obtain segments, and can potentially yield noisy results. We provide a comparison of ([4]) results with our Regulated Maximum Weight Clique (RMWC) based method.

Fig.1 shows an illustration of the proposed approach. Compared to the existing video co-segmentation methods, the proposed approach has the following advantages:

1. The proposed method employs object tracklets to obtain spatially salient and temporally consistent object regions for co-segmentation, while most of previous co-segmentation methods simply use pixel-level or region-level features to perform clustering. The perceptual grouping of pixels before matching reduces segment fragmentation and leads to a simpler matching problem.

2. The proposed approach does not rely on approximate solutions for object groups. The grouping problem is modeled as a Regulated Maximum Weight Clique (RMWC) problem for which an optimal solution is available. The use of only the salient object tracklets for grouping keeps the computational cost low.

3. Unlike the state-of-the-art single video object segmentation method ([29]), the proposed method can handle occlusions of objects, or objects going in and out of videos because the object tracklets are temporally local and there is no requirement for the object to continuously remain in the field of view of the video. Furthermore, there is no limitation on the number of object classes in each video and the number of common object classes in the video collection. Therefore the proposed approach can be used to extract objects in an unsupervised fashion from general video collections.

4. The proposed method is different from Maximum Weight Clique Problem which has already been explored in video object segmentation [17], in a way that the clique weights of the proposed method is not simply defined as the summation of node weights, but regulated by the intra-clique consistency term. Therefore, the extracted cliques have more global consistency, and similar objects from different videos are accurately grouped.

In Section 2, we describe our Regulated Maximum Weighted Clique (RMWC) based video co-segmentation approach. In section 3, we present the performance evaluation of the proposed algorithm. In Section 4., the paper is concluded.

2 Regulated Maximum Weight Clique Based Video Co-segmentation

2.1 The Framework

The proposed method consists of 2 stages: **(1)** Object Tracklets Generation: In this stage, we generate a number of object proposals ([5]) for each frame and use each of them as a starting point, and track the object proposals backward and forward throughout the whole video sequence. We generate reliable tracklets from the track set (those with high similarity over time) and perform non-maxima suppression to remove noisy or overlapping proposals. **(2)** Multiple Objects Co-Segmentation by Regulated Maximum Weight Cliques: A graph is generated by representing each tracklet as a node from all videos in the collection. The nodes of the graph are weighted by their appearance and motion scores, and edges are weighted by tracklet similarity. Edges with weight below a threshold are removed. A Regulated Maximum Weight Clique extraction algorithm is used to find objects ranked by score which is a combination of intra-group consistency and *Video Object Scores*. The object regions obtained from the video sets are used to initialize per-pixel segmentation [8] to get the final co-segmentation results.

2.2 Object Tracklets Generation

In this stage, the method in [5] is employed to generate a number of object proposals (which are likely to be 'object regions' in each frame). And each of the object proposals has a ***Video Object Score***, S^{object}, which is a combination of motion and appearance information:

$$S^{object}(x) = A(x) + M(x), \tag{1}$$

in which x is an object proposal. $A(x)$ is the appearance score (which is the objectness score defined by [5]). The appearance objectness score is high for regions that have a well defined closed boundary in space, different appearance from its surrounds and is salient [5]. $M(x)$ is the motion score (which is defined in [29] as the average Frobenius norm of optical flow gradient around the boundary of object proposal).

Efficient Object Proposal Tracking. We track every object proposal from each frame backward and forward to form a number of tracks for the object proposals (please see Fig.2).

A combined color, location, and shape similarity function is employed for object proposal tracking:

$$S^{simi}(x_m, x_n) = S^{app}(x_m, x_n) \cdot S^{loc}(x_m, x_n) \cdot S^{shape}(x_m, x_n), \tag{2}$$

in which x_m and x_n are object proposals from frame m and n respectively, S^{app} is the appearance similarity, S^{loc} is the location similarity which computes the overlap ratio between two regions, and S^{shape} is the shape similarity between the object proposals. Color histograms are used to model appearance. The descriptor for estimating shape similarity is computed by representing the contour of a region in normalized polar coordinates and sampling it from 0 to 360 degrees to form a vector. Dot products of descriptors are used for computing both shape similarity and appearance similarity.

Once the similarity function is defined, a simple greedy tracking method is employed to track large number of object proposals. By using the similarity scores defined in Eq.2, for a specific object proposal in the frame, the most similar object proposal in adjacent frame is selected to be the tracked proposal. The reason to use this method is mainly due to efficiency. As shown in Fig.2, the similarity matrices between all object proposals in adjacent frames are pre-computed. Based on the greedy method, tracking a specific object proposal to the next frame equals to finding the index of max value in a specific row of the similarity matrix. Thus this tracking process is computationally economical.

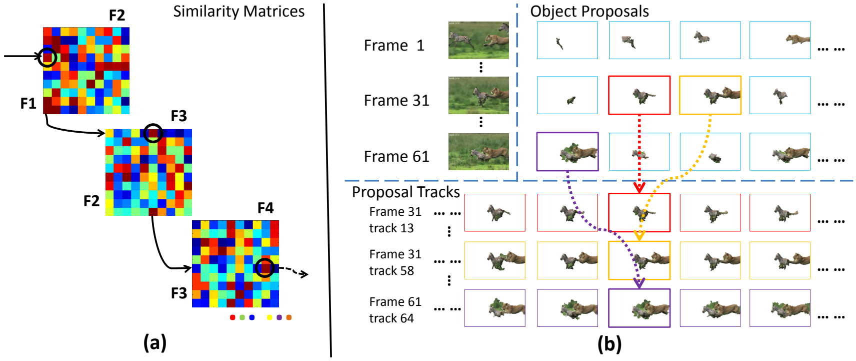

Fig. 2. Object Proposal Tracking. (a) Shows the similarity matrices between F1 (frame 1) and F2, F2 and F3, and F3 and F4. It also shows an example for tracking a specific object proposal (the 4th in F1): it finds the largest item from row 4 of similarity matrix F1 and F2 (the 1st item in this example); then it finds the largest item from row 1 of similarity matrix F2 and F3; and so on. Note that, only 10 object proposals (the matrices are 10 by 10) are shown in this figure for simplicity, but hundreds of objects proposals are used in the experiments. (b) Shows some object proposal tracks. In this example, several object proposals are generated for frame 31, and the object proposal shown in red box is tracked backward and forward to form a track throughout all the video frames. The same process is repeated for other object proposals (in orange and purple boxes as another two examples). This process is repeated for all the frames.

Non-maximum Suppression for Object Proposal Tracks. One can sample a large number of proposals per frame and, therefore, generate a larger number of tracks for an input video. Specifically, for a video that has F frames and each frame has N object proposals, $F \times N$ tracks could be obtained, since we generate tracks for each proposal. However, many of the object samples are overlapping and therefore their tracks are similar. A non-maximum suppression (NMS) ([7]) scheme is used to prune near duplicate tracks. For each object proposal track $X = \{x_1, ..., x_i, ...x_F\}$, an overall *Video Object Score* is computed as:

$$S^{object}(X) = \sum_{i=1}^{F}(S^{object}(x_i)), \tag{3}$$

where i is the frame index, and F is the number of frames.

Next, the track that has highest score is selected and all other tracks which have high overlap ratio $R^{overlap}$ with the selected track are removed. The value 0.5 is used for $R^{overlap}$, as suggested in ([7])). After that, the track with the second highest score among the surviving tracks is selected and the process is repeated. The process is continued iteratively until all tracks have been processed. The overlap ratio between two tracks X and Y is defined as:

$$R^{overlap}(X,Y) = \frac{\sum_{i=1}^{F}(x_i \cap y_i)}{\sum_{i=1}^{F}(x_i \cup y_i)}, \tag{4}$$

in which x_i and y_i are object proposals in the track X and Y respectively, and F is the number of frames for the video.

After the non-maximum suppression, typically only a small percentage of the total tracks (prior to NMS) are retained. To ensure validity of the track associations, we remove associations that are 1.5 standard deviations away from the mean track similarity (shown in Fig.3). This reduces the likelihood of a single track containing different objects.

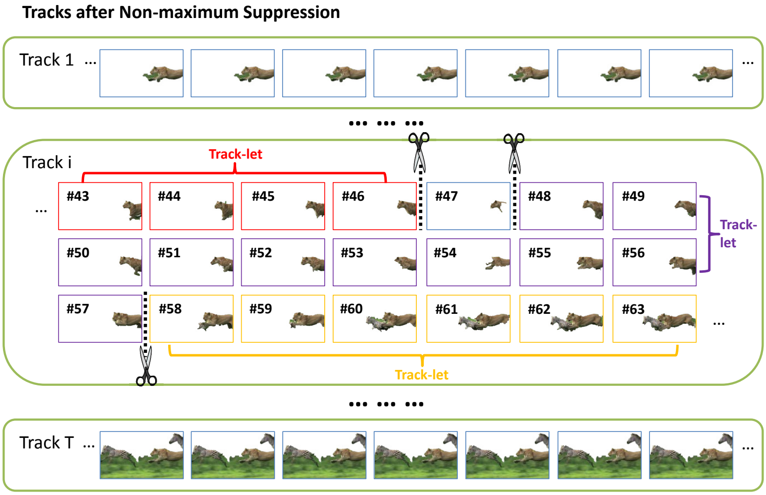

Fig. 3. Tracklet Splitting. In this example, after the Non-Maximum Suppression, there are T object proposal tracks selected. Track i is shown as an example to generate the object proposal tracklets. There are several adjacent frames which are not very similar compared to other adjacent frame pairs, therefore, they are split and several tracklets are generated (red, orange and purple).

2.3 Multiple Object Co-segmentation by Regulated Maximum Weight Cliques

Once object tracklets from the video collection have been obtained (Section 2.2), the next step is to discover salient object groupings in the video collection. We formulate the grouping problem as a Regulated Maximum Weight Clique Problem.

Clique Problems. Let $G = (V, E, W)$ be an undirected graph, where V is the set of vertices, E is the set of edges and W is a set of weights for each vertex. A clique is a complete subgraph of G, i.e. one whose vertices are pairwise adjacent. A **Maximal Clique** is a complete subgraph that is not contained in any other complete subgraph ([15]). **Finding All Maximal Cliques** from a graph is NP-hard and Bron-Kerbosch Algorithm ([2]) which has the worst case time complexity $O(3^{(n/3)})$ is known to be the most efficient algorithm in practice ([25]). The **Maximum Clique** Problem is to find maximum complete subgraph of G. The **Maximum Weight Clique** Problem deals with finding the clique which has maximum weight.

Problem Constraints. We use the following constraints for co-segmenting the objects from videos: **1)** The object proposal tracklets for the same class of objects should have similar appearance both within a video and across videos; however, due to the illumination differences across videos, for building color histograms in LAB space ([6]), the L channel (which represents the brightness) is only used for tracklets from the same video (intra-video edges), but a, b channels are used for tracklets from both same (intra-video edges) and different videos (inter-video edges). **2)** The shape of the same object would not change significantly during the same video, so the shape similarity is also used for building the edges for tracklets of the same objects in a video. **3)** The dominant objects should have high *Video Object Scores*, and **4)** the tracklets generated by an object should have low appearance variation. Based on these constraints, the graph is built as illustrated in Fig.1.

Graph Structure. The co-segmentation problem is formulated into a Regulated Maximum Weight Cliques Problem by denoting the object proposal tracklets to be the nodes. Based on constraints 1 and 2, edges are built between tracklets. There are two types of edges: intra-video edges and inter-video edges. The intra-video edge values are computed as a combined color histogram similarity in LAB color space and shape similarity:

$$\begin{aligned} E(X,Y) =& (shape(X) \cdot shape(Y)^T) \cdot \\ & \prod_{i=\{L,a,b\}} (hist(LAB_i(X)) \cdot hist(LAB_i(Y))^T), \end{aligned} \tag{5}$$

where $shape(X)$ and $shape(Y)$ are the shape descriptors (Sec.2.2) for object proposal tracklet X and Y respectively. The nearest two object proposals in the two tracklets are selected to represent the shapes of the tracklets.

And the inter-video edge values are computed as color histogram similarity of $\{a, b\}$ channels in LAB color space:

$$E(X, Y) = \prod_{i=\{a,b\}} (hist(LAB_i(X)) \cdot hist(LAB_i(Y))^T). \tag{6}$$

After computing the edges, the weak edges are removed (by a threshold).

Regulated Maximum Weight Clique Extraction. Based on constraint 3 and according to Equation 1, the weight of a node (object proposal tracklet) is computed as:

$$W(X) = \sum_{i=1}^{f} (S^{object}(x_i)), \tag{7}$$

in which f is the number of object proposals in this tracklet. $W(X)$ is the sum up of the *Video Object Score* of all object proposals contained in this tracklet.

Based on constraint 4, the weight of a clique is defined as:

$$W(C) = \mathit{\Gamma}_{hist}(C) \cdot \sum_{i=1}^{n(C)} (W(X_i)), \tag{8}$$

in which $C = \{X_1, ..., X_{n(C)}\}$ is a clique, X_i is a node (tracklet) contained in this clique, $n(C)$ is the number of nodes in this clique, and $\mathit{\Gamma}_{hist}(C)$ is the color histogram consistency regulator which computes the mean color histogram consistency of all the object proposals contained in the clique:

$$\mathit{\Gamma}_{hist}(C) = \frac{\sum_{i=1}^{f(C)} \sum_{(j=1 \wedge j \neq i)}^{f(C)} (hist(x_i) \cdot hist(x_j)^T)}{f^2(C) - f(C)}, \tag{9}$$

in which x_i and x_j are object proposals in clique C, $f(c)$ is the number of object proposals in this clique, and $hist(\cdot)$ is the $\{a, b\}$ channel color histogram in LAB space.

By this formulation, the clique that has the highest score represents the object with largest combined score of inter-object consistency and objectness. This problem is different from Maximum Weight Clique problem and can not be solved by standard methods ([15,11]), because the clique weights are not simply defined as the summation of node weights and the weights varies over iterations as we extract objects one by one. Therefore, we call this as **Regulated Maximum Weight Cliques Problem**. Note that, we want to retrieve all Regulated Maximum Weighted Cliques as possible objects. This is achieved through iteratively finding and removing the Regulated Maximum Weight Cliques from the graph to get a ranked list of cliques (i.e. objects).

A modified version of Bron-Kerbosch Algorithm ([2]) which also has a worst-case complexity of $O(3^{(n/3)})$ is proposed to solve this problem:

Step 1, Apply Bron-Kerbosch Algorithm to find all the maximal cliques from the graph;
Step 2, Compute the weight of each clique in linear time;
Step 3, Find the clique with the highest weight and remove all the nodes associated with this clique, update the clique structures and recompute the weights. This process could be performed for multiple times in order to extract multiple object groupings from the videos.

Note that, the high-complexity doesn't prohibit the use of this algorithm. The object tracklets generation stage removes most of the spurious tracklets. For videos evaluated in this paper, the maximum clique extraction process took less than a second on a standard laptop. The object regions obtained from the video sets are used to initialize per-pixel segmentation [8] to get the final co-segmentation results.

3 Experiments

The proposed method was tested on the video co-segmentation dataset (MOViCS dataset ([4])) and was compared with several other methods. The results show that it performs better both qualitatively and quantitatively. Detailed analysis is presented to show that the co-segmentation method produces better segmentation results by using information from multiple videos. Results also show that the proposed method could handle occlusions on which the state-of-the-art single video object segmentation method fails.

3.1 MOViCS Dataset

To the best of our knowledge, MOViCS dataset ([4]) is the only video co-segmentation dataset which has the ground truth annotations for quantitative analysis. It contains 4 video sets which totally has 11 videos, 5 frames of each video have pixel-level annotations for the object labels.

Experimental Setup. Following the setup in [4], the intersection-over-union metric is employed to quantify the results: $M(S,G) = \frac{S \cup G}{S \cap G}$, where S is a set of segments and G is the ground truth. The co-segmentation score for a set of video is defined as $Score_j = \max_i M(S_i, G_j)$, where S_i denotes all segments grouped into an object class i. And a single average score is defined for all object classes as: $Score = \frac{1}{C} \sum_j Score_j$, where C is the number of object classes in the ground truth.

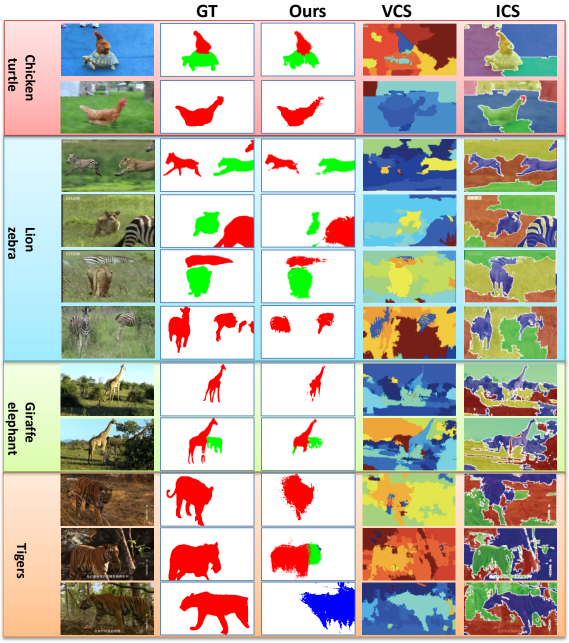

Fig. 4. Video Co-Segmentation Results on MOViCS Dataset. Each row is the results of a video in MOViCS dataset. Column 1 is one original frame from the video; column 2 ('GT') is the ground truth for co-segmentation, red regions correspond to the first object in the video set and green regions correspond to the second object in the video set; column 3 ('Ours') is the results of the proposed method, red and green regions correspond to the first and second objects in the video set and blue region corresponds to the third object; column 4 ('VCS') and 5 ('ICS') are the results of video co-segmentation method from [4] and [13] respectively. Row 1 and 2 are for 'chicken&turtle' video set, row 3-6 are for 'lion&zebra' video set, row 7 and 8 are for 'giraffe&elephant' video set and row 9-11 are for 'tigers' video set. Please refer to supplementary material for more detailed results.

Comparisons with State-of-the-art Methods. The proposed method is compared with several state-of-the-art Co-Segmentation methods, see Table 1 (the results of VCS [4] and ICS [13] are obtained from [4]). As mentioned in Section 2.3, we use a threshold to remove the weak edges, here we show the results by using a single threshold for all video sets (see column 'Ours1'), and also using optimal thresholds for different video sets (in column 'Ours2'). Qualitative results on this dataset are shown in Fig.4.

The evaluation shows that the proposed method improves on the state of the art. The average improvement is more than 20%. From Fig.4, we can see that ours are the only results which are visually very similar to the ground truth. Unlike prior methods, our method does not have the propensity for breaking objects into a number of fragments and the method also produces better contours for the objects. The only video in which the object regions are not accurately segmented is the 3rd video in video set ‘tigers’.This is due to the large difference in appearance of animals from other two videos and qualitatively our method is still the best for this video set.

Table 1. Quantitative comparisons with the state of the art on MOViCS dataset. ‘Ours1’ shows results of using a single threshold (0.65) for removing the edges, and ‘Ours2’ shows results of using different thresholds for each video sets (the thresholds are [0.65 0.86 0.45 0.65] for these four video sets respectively.)

Video Set	Ours1	Ours2	VCS [4]	ICS [13]
Chicken&turtle	**0.860**	**0.860**	0.65	0.08
Zebra&lion	**0.588**	**0.636**	0.48	0.23
Giraffe&elephant	**0.528**	**0.639**	0.52	0.07
Tiger	**0.336**	**0.336**	0.30	0.30
Overall	**0.578**	**0.617**	0.49	0.17

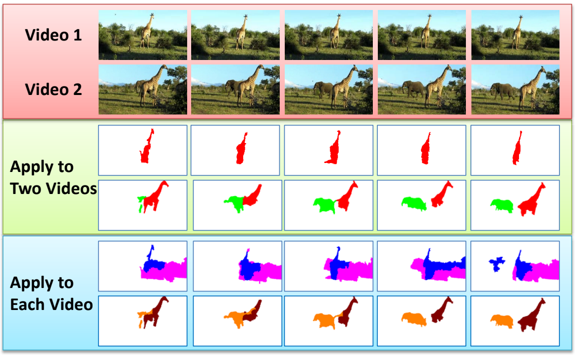

Fig. 5. Advantages of the Proposed Video Co-Segmentation Method. Row 1 and row 2 show sample frames from two videos respectively. Row 3 and 4 are the video co-segmentation results of the proposed method for these two videos. Red regions correspond to the first object and green regions correspond to the second object. Row 5 and 6 are the segmentation results of applying the method separately to each video. Blue and dark red regions correspond to the first objects, and pink and orange regions correspond to the second objects.

Advantages of Video Co-segmentation Method. Fig.5 shows how the Co-Segmentation framework help the segmentation results for each video. In this example, we have two videos, if the proposed method is applied to these two videos, the segmentation results are shown in row 3 and 4; if the proposed method is applied separately for each video, the segmentation results are shown in row 5 and 6. It is quite clear that the Co-Segmentation method not only helps to relate the object labels (red regions for the giraffe in row 3 and 4), but also helps to get more accurate segmentation results (video 2 helps video 1 to get better segments for giraffe in row 3; without video 2, it could only get poor segmentation of giraffe in row 5).

Advantages over Single Video Object Segmentation Method. Fig.6 shows the comparisons between the proposed method (VOCS) and the state-of-the-art single video object segmentation (SVOS) method ([29]). Results show that the proposed method could segment objects by using information from other videos (row 2 and 5 in the figure), in contrast, single video object segmentation method mistakenly merges two objects together if they have similar motions (row 3 and 6 in the figure). Also, the proposed method is able to handle occlusions well (row 8 in the figure), while the single video object segmentation method generates wrong labels when there are occlusions of the objects (row 9 in the figure). We compared video object segmentation results quantitatively on MOViCS dataset in Table 2. We observe that, if there are two or more objects appearing in the video, or there are occlusions (e.g. 'elephant_giraffe_all2') of the objects, or the objects do not appear in all the frames (e.g. 'lion_zebra_all1'), the proposed method works much better than single video object segmentation method; if there is only one object in the video, the single video object segmentation method sometimes works better (e.g. 'tiger1_all8' results).

Table 2. Quantitative Comparison of Single Video Object Segmentation (SVOS) with Video Object Co-Segmentation (VOCS)

Video Name (Object)	SVOS ([29])	VOCS
ChickenNew (chicken)	0.740	**0.857**
Chicken_on_turtle (chicken)	0.306	**0.823**
Chicken_on_turtle (turtle)	0.563	**0.807**
Elephant_giraffe_all1 (giraffe)	0.570	**0.680**
Elephant_giraffe_all2 (giraffe)	0.122	**0.557**
Elephant_giraffe_all2 (elephant)	0.085	**0.557**
Lion_zebra2 (lion)	0.254	**0.817**
Lion_zebra2 (zebra)	0.510	**0.619**
Lion_zebra_all1 (lion)	0.391	**0.727**
Lion_zebra_all1 (zebra)	**0.529**	0.361
Lion_zebra_all2 (lion)	**0.883**	0.830
Lion_zebra_all2 (zebra)	0.000	**0.547**
Zebra_grass (zebra)	0.403	**0.508**
Tiger1_all8 (tiger)	**0.494**	0.428
Tiger1_all9 (tiger)	**0.841**	0.522
Tiger1_all10 (tiger)	0.384	**0.637**

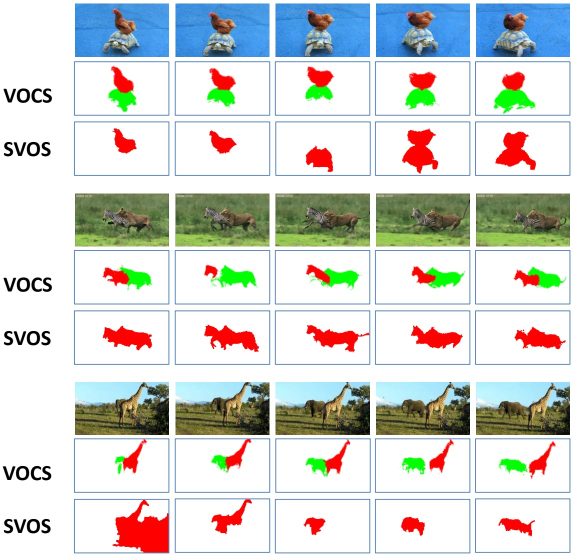

Fig. 6. Comparison between the proposed method (VOCS) and Single Video Object Segmentation (SVOS) method ([29]). Three groups of results are shown here. In each of them, the first rows (row 1, 4 and 7) are sample frames from the videos; the second rows (row 2, 5 and 8) are results of the proposed method; and the third rows (row 3, 6 and 9) are results of the single video object segmentation method. For the results, the red regions correspond to the first objects and green regions correspond to the second objects. Since the single video object segmentation method only extract primary objects from the videos, only red regions could be shown in the results.

Table 3. Quantitative results on Safari dataset

Object:	Buffalo	Elephant	Giraffe	Lion	Sheep
Baseline [4]	0.686	0.266	0.024	0.302	0.048
Ours	**0.869**	**0.353**	0.024	**0.317**	**0.363**

3.2 Safari Dataset

Since video object co-segmentation problem is new and there is only one publicly available dataset with ground truth, we collected another challenging dataset (named 'Safari dataset'[1]) by getting new videos and also reusing some videos from MOViCS dataset. We annotated the key frames. The new dataset will be made publicly available. This Safari dataset is challenging, since the Safari contains 5 classes of animals and a total of 9 videos. For each animal class, Safari dataset has a video which only contains this class. Other videos contain

[1] http://crcv.ucf.edu/projects/video_object_cosegmentation/

two of the animal classes. The goal is to input the 9 videos together and do co-segmentation simultaneously for all of them. We show the ground truth and our co-segmentation results in Fig.7 and show quantitative results in Table 3.

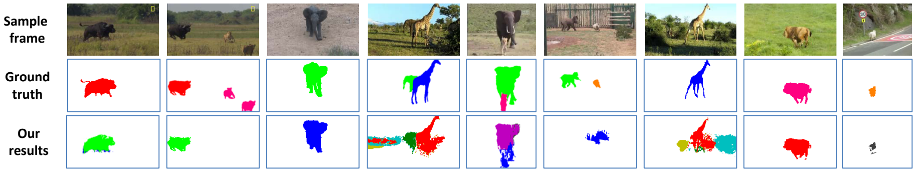

Fig. 7. The ground truth and our results on Safari dataset. Row 1 shows one frame from each of the video. Row 2 shows the ground truth annotations, in which same color represents same object class. And row 3 shows our results, in which same color also represents same object classes. Please note that, there is no relationship between the colors of row 2 and row 3.

4 Conclusions

This paper formulates the video object discovery and co-segmentation problem into a Regulated Maximum Weight Clique (RMWC) Problem and solves it using a modified version of Bron-Kerbosch Algorithm. The success of the proposed method relies on i) use of the objectness measure to obtain spatially coherent region proposals, ii) tracking of region proposals, which selects proposals with consistent appearance and smooth motion over time, and iii) using different weighting functions for within video and across video matching for graph construction, which results in improved grouping. Experimental results shows that the method outperforms the state-of-the-art video co-segmentation methods.

Acknowledgments. This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20066. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S.Government.

References

1. Alexe, B., Deselares, T., Ferrari, V.: Measuring the objectness of image windows. PAMI (2012)
2. Bron, C., Kerbosch, J.: Algorithm 457: finding all cliques of an undirected graph. Communications of the ACM 16(9), 575–577 (1973)

3. Chen, D.J., Chen, H.T., Chang, L.W.: Video object cosegmentation. In: ACM MM, pp. 805–808 (2012)
4. Chiu, W.C., Fritz, M.: Multi-class video co-segmentation with a generative multi-video model. In: CVPR (2013)
5. Endres, I., Hoiem, D.: Category independent object proposals. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 575–588. Springer, Heidelberg (2010)
6. Fairchild, M.D.: Color appearance models. John Wiley & Sons (2013)
7. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. PAMI 32(9), 1627–1645 (2010)
8. Fulkerson, B., Vedaldi, A., Soatto, S.: Class segmentation and object localization with superpixel neighborhoods. In: ICCV, pp. 670–677. IEEE (2009)
9. Galasso, F., Iwasaki, M., Nobori, K., Cipolla, R.: Spatio-temporal clustering of probabilistic region trajectories. In: ICCV, pp. 1738–1745. IEEE (2011)
10. Grundmann, M., Kwatra, V., Han, M., Essa, I.: Efficient hierarchical graph based video segmentation. In: CVPR (2010)
11. Jain, B., Obermayer, K.: Extending bron kerbosch for solving the maximum weight clique problem. arXiv preprint arXiv:1101.1266 (2011)
12. Joulin, A., Bach, F., Ponce, J.: Discriminative clustering for image co-segmentation. In: CVPR, pp. 1943–1950. IEEE (2010)
13. Joulin, A., Bach, F., Ponce, J.: Multi-class cosegmentation. In: CVPR, pp. 542–549. IEEE (2012)
14. Kim, G., Torralba, A.: Unsupervised detection of regions of interest using iterative link analysis. In: NIPS (2009)
15. Kumlander, D.: A new exact algorithm for the maximum-weight clique problem based on a heuristic vertex-coloring and a backtrack search. In: Proc. 5th Int. Conf. on Modelling, Computation and Optimization in Information Systems and Management Sciences, pp. 202–208 (2004)
16. Lee, Y., Kim, J., Grauman, K.: Key-segments for video object segmentation. In: ICCV, pp. 1995–2002. IEEE (2011)
17. Ma, T., Latecki, L.: Maximum weight cliques with mutex constraints for video object segmentation. In: CVPR, pp. 670–677. IEEE (2012)
18. Mukherjee, L., Singh, V., Peng, J.: Scale invariant cosegmentation for image groups. In: CVPR, pp. 1881–1888. IEEE (2011)
19. Ochs, P., Brox, T.: Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions. In: ICCV, pp. 1583–1590. IEEE (2011)
20. Rother, C., Minka, T., Blake, A., Kolmogorov, V.: Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. In: CVPR, pp. 993–1000. IEEE (2006)
21. Rubinstein, M., Joulin, A., Johannes, K., Liu, C.: Unsupervised joint object discovery and segmentation in internet images. In: CVPR (2013)
22. Rubio, J.C., Serrat, J., López, A.: Video co-segmentation. In: Lee, K.M., Matsushita, Y., Rehg, J.M., Hu, Z. (eds.) ACCV 2012, Part II. LNCS, vol. 7725, pp. 13–24. Springer, Heidelberg (2013)
23. Russell, B.C., Freeman, W.T., Efros, A.A., Sivic, J., Zisserman, A.: Using multiple segmentations to discover objects and their extent in image collections. In: CVPR, vol. 2, pp. 1605–1614. IEEE (2006)
24. Sivic, J., Russell, B.C., Efros, A.A., Zisserman, A., Freeman, W.T.: Discovering objects and their location in images. In: ICCV, vol. 1, pp. 370–377. IEEE (2005)

25. Tomita, E., Tanaka, A., Takahashi, H.: The worst-case time complexity for generating all maximal cliques and computational experiments. Theoretical Computer Science 363(1), 28–42 (2006)
26. Vicente, S., Rother, C., Kolmogorov, V.: Object cosegmentation. In: CVPR, pp. 2217–2224. IEEE (2011)
27. Xu, C., Xiong, C., Corso, J.J.: Streaming hierarchical video segmentation. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VI. LNCS, vol. 7577, pp. 626–639. Springer, Heidelberg (2012)
28. Yuan, J., Zhao, G., Fu, Y., Li, Z., Katsaggelos, A.K., Wu, Y.: Discovering thematic objects in image collections and videos. IEEE Transactions on Image Processing 21(4), 2207–2219 (2012)
29. Zhang, D., Javed, O., Shah, M.: Video object segmentation through spatially accurate and temporally dense extraction of primary object regions. In: CVPR (2013)
30. Zhao, G., Yuan, J., Hua, G.: Topical video object discovery from key frames by modeling word co-occurrence prior. In: CVPR (2013)