Video co-segmentation

Jose C. Rubio, Joan Serrat, Antonio López

Computer Vision Center and Computer Science Department
Universitat Autònoma de Barcelona. Bellaterra. Spain

Abstract. Segmentation of a single image is in general a highly underconstrained problem. A frequent approach to solve it is to somehow provide prior knowledge or constraints on how the objects of interest look like (in terms of their shape, size, color, location or structure). Image co-segmentation trades the need for such knowledge for something much easier to obtain, namely, additional images showing the object from other viewpoints. Now the segmentation problem is posed as one of differentiating the similar object regions in all the images from the more varying background. In this paper, for the first time, we extend this approach to video segmentation: given two or more video sequences showing the same object (or objects belonging to the same class) moving in a similar manner, we aim to outline its region in all the frames. In addition, the method works in an unsupervised manner, by learning to segment at testing time. We compare favorably with two state-of-the-art methods on video segmentation and report results on benchmark videos.

1 Introduction

Video segmentation has been defined as the problem of partitioning a video sequence into coherent regions with regard to motion and appearance properties [1]. We adopt here a more specific definition: we are interested only in those regions belonging to the objects of interest, which are those appearing in the foreground, over a possibly changing background. The outcome, thus, is a set of regions spanning space and time, sometimes dubbed 'tubes'. Foreground segmentation is useful for several computer vision tasks including video analysis, object tracking, object recognition, 3D reconstruction, video retrieval, and activity recognition.

In spite of its potential applications, relatively few works address the problem of video segmentation, perhaps because the addition of one dimension increases the difficulty of unconstrained 2D segmentation. The reviewed works show that the most favored approach is to extend single image segmentation techniques to multiple frames, exploiting the fact that there is redundancy along the time axis and that the motion field is smooth. Thus, for instance, Levinshtein et al. [1] extend superpixel grouping [2] (also known as turbopixels) to 3D. Sundaram and Keutzer [3] apply spectral clustering to all the video sequence pixels with an affinity matrix given by the gPb 2D contour detection algorithm [4] which combines intensity, color and texture. Several works pose the problem as one of labeling using minimum energy optimization of a Markov Random Field where nodes are now voxels [5–8] or 2D regions [9], again a successful segmentation strategy in single images. Grundmann et al. [10] build their hierarchical algorithm for long sequences upon Felzenszwalb and Huttenlocher's [11] graph algorithm for 2D image segmentation. Likewise, Huang et al. adapt the graph-cut algorithm to run on 3D hypergraphs whose nodes are regions resulting from an oversegmentation of each frame.

Our approach is different in that we do not pursue video segmentation through the extension of some image segmentation *algorithm* but instead the extension of the *concept* of co-segmentation to include videos. Being segmentation a highly underconstrained problem, many practical methods resort to providing prior knowledge or constraints on how the objects of interest look (in terms ok shape, size, color, location or structure). Image co-segmentation trades the need for such knowledge for something much easier to obtain, namely, additional images showing the same object, or objects of the same class, from different viewpoints. Now the segmentation problem is posed as one of differentiating the similar object regions in all the images from the more varying background. In this paper, for the first time, we extend this approach to video segmentation: given two or more video sequences showing the same object (or an object belonging to the same class) moving in a similar manner, we aim at outlining its spatio-temporal regions in all the videos.

Our method builds on a recent co-segmentation work by Rubio et al. [12], as we want to preserve several of its desirable properties:

- Foremost, our model does not need training segmentation data in order to learn the foreground and background distributions. On the contrary, it learns such distributions at testing (segmentation) time.
- As opposed to other co-segmentation methods, ours models the background in addition to the foreground distribution, thus being able to cope with videos with similar backgrounds.
- Following the rationale of co-segmentation, our model is able to work with more than a pair of videos. In fact, with more videos both the commonality of the foreground and the diversity of the background increase, thus favoring this approach.

We want to make clear that our method is not a simple extension of [12] to one more dimension. In fact, this is not possible because our proposed method is not a segmentation algorithm, but a reformulation that requires a different graphical model on which inference takes place in order to label regions as foreground or background. It also introduces the concept of a hierarchy of tubes and image regions. Moreover, our method does not work by co-segmentation of frames but of whole video sequences, as we will explain.

2 Method Overview

Our goal is to perform figure/ground separation on multiple videos, posed as an optimization problem. Given a rough initialization of the foreground labeling,

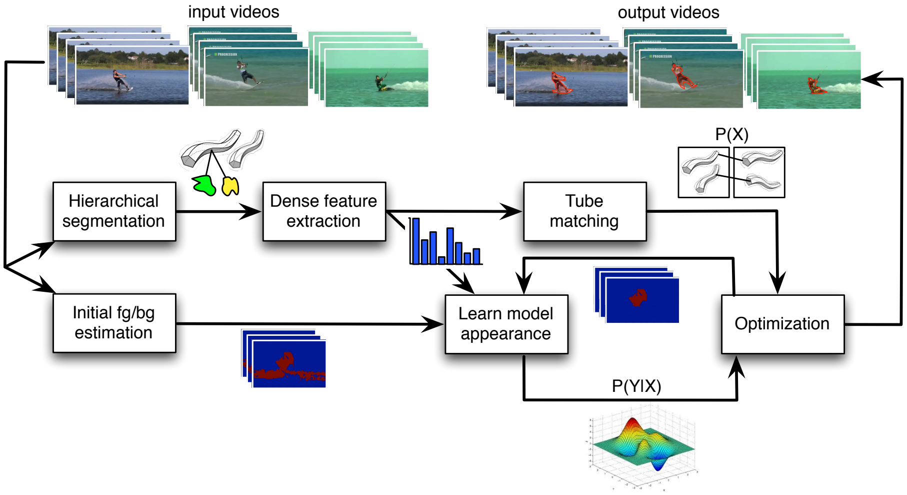

Fig. 1. Diagram describing the steps of the algorithm. First a set of tubes and regions is extracted using a hierarchical segmentation. Dense features are extracted from such elements, in order to learn a motion/appearance model of foreground and background, given a rough initialization. Finally the figure/ground separation is refined in an iterative process, and the labeling inter-video is constrained in the prior term, by matching video tubes.

one can model a joint appearance/motion model of the foreground and background, and iteratively refine the result and update the models. However, such approach is very sensitive to wrong initializations. We introduce a matching process to constrain the labeling across different videos, and provide the model with robustness against wrong initializations.

Given a set of input videos, we start by grouping the pixels at two levels. At the higher level, video pixels are grouped in space-time, defining a set of video tube volumes. At the lower level, pixels are grouped into regions within each frame, as is typically done by regular image segmentation algorithms. Each of these elements (tubes and regions) are described using densely extracted features.

An initial estimation of the foreground and background labeling is needed in order to construct a probabilistic distribution of the feature vectors of tubes and regions. We obtain such a labeling using a measure of objectness together with a saliency algorithm. As this initialization is a rough approximation, the models describing foreground and background are likely to contain incorrectly labeled samples, which may lead the iterative process towards a trivial solution (the whole video labeled as background or foreground).

We present a probabilistic framework where the likelihood of each element belonging to the foreground or background is calculated from the above mentioned representations. In addition, the model constraints are introduced through a prior term that provides region-tube consistency and enforces labeling coher-

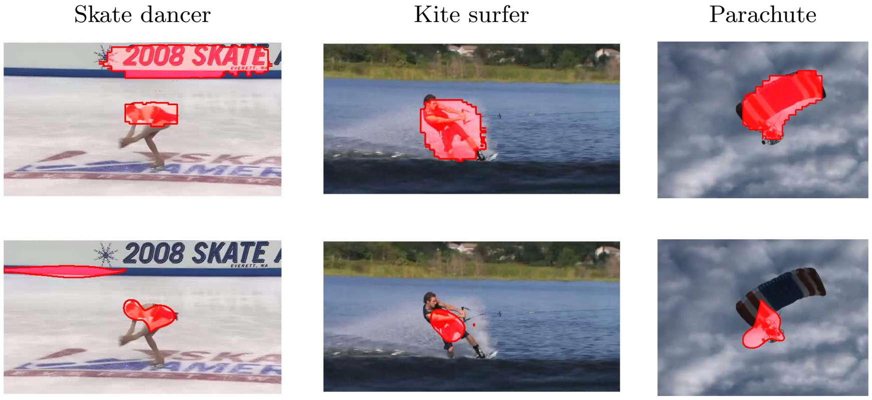

Fig. 2. The first row shows examples of initializations with the objectness measure of [13]. The second row shows examples of the saliency-based measure. In some cases the objectness measure achieves a more accurate result (Parachute), while in others the saliency obtains a better labeling (Dancer).

ence between corresponding objects across different input videos. The foreground and background representations are iteratively improved from the new labelings that result from the optimization of the joint posterior distribution of the labels of all tubes and regions. Figure 1 shows a diagram detailing the steps of the process.

3 Iterative Foreground/Background Modeling

The task of co-segmentation requires relying on features that discriminate foreground from background regions. It seems a good assumption that foreground objects will have an appearance that is distinct from that of the background in addition to dissimilar motion behavior. For this reason, our model is composed of two types of elements: tubes and regions. The features of the former encode motion while those of the latter encode appearance.

In order to generate these elements we apply the algorithm of [10], which produces hierarchical video segmentations with coherent regions along time. Two levels of segmentation granularity are generated for each frame. With the higher (less granulated) level we generate the collection of tubes by *stacking* the regions having the same label across frames. With the lower (more granulated) level we generate the collection of regions. Since the segmentation is hierarchical, we can trivially establish a non-ambiguous correspondence between region elements and their corresponding tubes at the higher level. Our goal is to consistently label both regions and tubes as foreground or background.

3.1 Initial foreground estimate

The unsupervised and iterative nature of our approach requires an initial estimation of the appearance of the foreground and background. We have experimented with two different approaches. The first is the objectness measure of [13], based on visual cues that we apply to each frame independently. The second is the video saliency measure of [14], which takes motion into account. In Figure 2 we show examples of initializations. The objectness measure tends to fail in cases where the foreground contains motion blur and also when other elements of the scene appear to be objects due to their sharp edges and closed boundaries. The saliency measure works well in general, but performs poorly when the foreground is a fairly complex object, or composed by several parts. To obtain our initialization we combine both estimations by averaging the pixel score returned by objectness and saliency, and thresholding the resulting map.

3.2 Formulation

Let $\mathcal{V} = \{V_1, V_2, \ldots, V_N\}$ be a set of on N input videos. For all those input videos we extract two sets of elements to be labeled, tubes and regions, which we respectively denote as $\mathcal{T} = \{t_1, t_2, \ldots, t_M\}$ and $\mathcal{R} = \{r_1, r_2, \ldots, r_L\}$. A region refers to a set of pixels extracted from each video frame, while a tube is defined as a set of stacked pixels across different frames of the same video. Note that while a region lies within one video frame, a tube spreads over various frames.

We propose a Markov Random Field that comprises tube nodes and region nodes of all videos in two separate layers, as illustrated in Figure 3. We represent these nodes as a vector of boolean random variables divided into two disjoint sets, denoted as $\mathbf{X} = (X_i)_{i\in\mathcal{T}} \cup (X_j)_{j\in\mathcal{R}}$. A variable $X = 0$ indicates that the tube or region belongs to the background, and $X = 1$ otherwise. An observation $Y \in \mathbf{Y}$ is a histogram of visual cues extracted from a tube or region element. Our goal is to obtain the MAP labeling that maximizes the following posterior, given the set of observations $\mathbf{Y}$:

$$P(\mathbf{X}|\mathbf{Y}) = P^{\mathcal{T}}(Y^{\mathcal{T}}|X^{\mathcal{T}})P^{\mathcal{R}}(Y^{\mathcal{R}}|X^{\mathcal{R}})P(\mathbf{X}). \tag{1}$$

The factor $P^{\mathcal{T}}(Y^{\mathcal{T}}|X^{\mathcal{T}})$ defines the likelihood of each tube variable belonging to the foreground or background (they will be detailed in the following Sect. 3.3). Analogously, $P^{\mathcal{R}}(Y^{\mathcal{R}}|X^{\mathcal{R}})$ defines the likelihood of each region belonging to the foreground or background. The factor $P(\mathbf{X})$ is a prior that imposes intra-video and inter-video labeling constraints, to be introduced in Sect. 4. The set of random variables $X^{\mathcal{T}}$ refers to the set of tube variables $(X_i)_{i\in\mathcal{T}}$, while $X^{\mathcal{R}}$ refers to the set of region variables $(X_j)_{j\in\mathcal{R}}$. We describe in detail each of the factors in the following sections.

3.3 Figure-ground likelihood model

Once an initial labeling estimation is available we can model the appearance of foreground and background. As the tubes are spatio-temporal entities, their

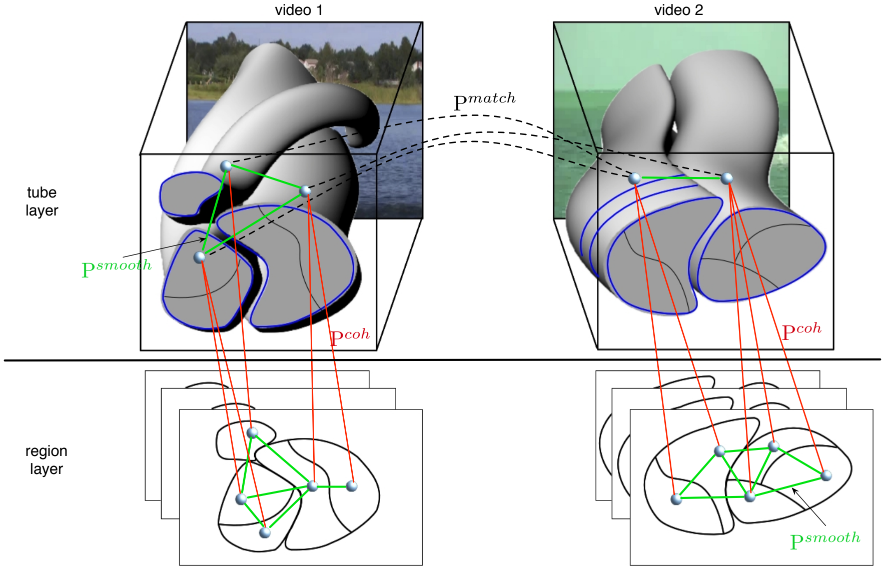

Fig. 3. Two layer model of regions and tubes. The upper layer shows the tube volumes and the lower layer shows the regions. The graph nodes represent the tube and region random variables, and the connections represent the pairwise relationships encoded in the prior. The green edges illustrate the smoothing constraints P^{smooth} and the red lines connecting the two layers represent the tube-region coherence, P^{coh}. The dotted lines that connect tubes from different videos refer to the matching constraint P^{match}.

related likelihood $P^{\mathcal{T}}(Y^{\mathcal{T}}|X^{\mathcal{T}})$ encodes 3D features, or motion features. In the case of the regions, being 2D image areas, their distribution $P^{\mathcal{R}}(Y^{\mathcal{R}}|X^{\mathcal{R}})$ encodes image appearance features such as color or texture. Thus, each type of element encodes complementary information.

Describing the motion characteristics of a tube is achieved by uniformly sampling 3D HoG descriptors [15] along the tube volume. The descriptors span space and time, and are clustered in a unique descriptor vector using a histogram of accumulated descriptors [16]. Similarly, we sample 8×8 patches from the region elements and calculate texture (LBP) descriptors that are also aggregated in a single vector.

The likelihood distributions of the feature vectors of regions and tubes are calculated as:

$$P^{\mathcal{T}}(Y^{\mathcal{T}}|X^{\mathcal{T}}) = \prod_{k \in V} \prod_{t \in \mathcal{T}(k)} \hat{P}_k^f(H_t)X_t + \hat{P}_k^b(H_t)\overline{X}_t. \qquad (2)$$

The term $\hat{P}_k(H_t)$ is the probability of a tube descriptor H_t, given by the logistic regressor k trained with 3D HoG descriptors. Note that there are as many classifiers as input videos. To avoid quering with the same samples the models were learnt with, each regressor k is trained with data from videos $\{V_1, \dots, V_{k-1}, V_{k+1}, \dots, V_n\}$ and tested with the remaining video V_k. This removes the inherent likelihood bias towards training features from the same video.

Similarly, the likelihood of the region variables can be formulated as:

$$P(Y^{\mathcal{R}}|X^{\mathcal{R}}) = \prod_{l \in V} \prod_{r \in \mathcal{R}(l)} \hat{P}_l^f(LBP_r)X_r + \hat{P}_l^b(LBP_r)\overline{X}_r, \tag{3}$$

where the term $\hat{P}_l(LBP_r)$ refers to the probability of the regressor l trained with LBP texture descriptors, given a region descriptor LBP_r. The super-indices f and b refer to the labeling of foreground and background respectively.

3.4 Iterative likelihood estimation and labeling

Recall our intention is to start with a rough initialization of motion and appearance distributions, and then iteratively update them. At each iteration the motion and texture distributions are modeled using the labeling from the previous iteration (Eq. (2) and Eq. (3)). The likelihood of a given tube and region is obtained by querying the distributions $P^{\mathcal{T}}$ and $P^{\mathcal{R}}$ respectively. Finally, the posterior expression of Eq. (1) is optimized and a new labeling is generated. The process repeats and the likelihood models are progressively refined until satisfying some convergence criteria. Ideally, at each step the labeling solution obtained gets closer to the optimal foreground/background separation. Although fg/bg models are based on Support Vector Machines (SVM) in our framework, one could use statistical models or even modern manifold learning methods. The procedure is graphically detailed in Figure 1 and in Algorithm 1.

Algorithm 1 Iterative Foreground/Background Modeling

```
Initialize $\mathbf{X}_t, \mathbf{X}_r \leftarrow$ (objectness & saliency), $\forall t \in \mathcal{T}, \forall r \in \mathcal{R}$.
repeat
    Train $SVM_k \leftarrow \mathbf{X}_t, \forall V_k \in \mathcal{V}$
    Train $SVM_l \leftarrow \mathbf{X}_r, \forall V_l \in \mathcal{V}$
    Query regressor $k$: $\hat{P}(H_t) \leftarrow SVM_k(H_t), \forall t \in \mathcal{T}$
    Query regressor $l$: $\hat{P}(LBP_r) \leftarrow SVM_l(LBP_r), \forall r \in \mathcal{R}$
    $\mathbf{X}^* \leftarrow \arg\max_{\mathbf{X}} P(\mathbf{X}|\mathbf{Y}) \propto P(\mathbf{Y}|\mathbf{X})P(\mathbf{X})$
    Update labels $\mathbf{X}_t, \mathbf{X}_r \leftarrow \mathbf{X}^*$
until stop criteria / limit num. iterations
```

4 Modeling the Prior

An iterative procedure such as the one described in the previous section does not work in practice if we assume a uniform prior distribution. An inaccurate initial estimation would produce poor appearance models of foreground and background that within such an iterative process are likely to expand until reaching a trivial solution (e.g every label is background).

We propose an informative prior that imposes certain restrictions that bound the foreground/background partition and limit its expansion. Moreover, it is desirable to constrain the labeling of the tubes and the regions included in them, in order to enforce a coherent labeling of regions and tubes. For the first constraint, we establish correspondences between tubes from different videos, similarly to the work of [12]. Our goal is to combine the information from different videos and constrain the label of corresponding elements (tubes) to take the same value. Therefore, we seek to exploit inter-video information in such manner that a defective initialization in one of the videos will be balanced by an expected correct labeling in other input videos. Figure 3 shows a graphical interpretation of the model constraints represented by the connections between tubes from different videos (black, dotted) as well as by the links connecting tubes and regions (red).

The way to impose such constraints is through the prior term $P(\mathbf{X})$, of the posterior in Eq. (1). The prior term factorizes into two terms corresponding to the two types of constraints we want to enforce:

$$P(\mathbf{X}) = \prod_{(t_1,t_2)\in\mathcal{E}} P^{match}(X_{t_1}, X_{t_2}) \prod_{(r,t)\in\mathcal{F}} P^{coh}(X_r, X_t) \tag{4}$$

The set $\mathcal{E}$ contains all the pairs of tubes that are in correspondence, while the set $\mathcal{F}$ defines pairs of regions and tubes, such that for a given pair (r, t) the region r is contained in the tube t. The first factor P^{match} is responsible for ensuring that corresponding tubes from different videos take the same label, and we formulate it as:

$$P^{match}(X_{t_1}, X_{t_2}) = \begin{cases} \alpha & \text{if } X_{t_1} \neq X_{t_2} \\ \beta & \text{otherwise} \end{cases} \tag{5}$$

The second factor P^{coh} enforces a coherent labeling between the hierarchical levels of the model, such that a region takes the same label as the tube to which it belongs. Formally,

$$P^{coh}(X_r, X_t) = \begin{cases} \gamma & \text{if } X_r \neq X_t \\ \delta & \text{otherwise} \end{cases} \tag{6}$$

The parameters $\alpha, \beta, \gamma, \delta$ are constant probability values. These are set in order to assign a high probability ($\beta, \delta \sim 1$) to a coherent labeling and a low probability otherwise ($\alpha, \beta \sim 0$).

Whereas the set $\mathcal{F}$ derives directly from the hierarchical segmentation, the set $\mathcal{E}$ requires a matching strategy to put tubes from different videos in correspondence. In this work we use a simple approach based on the tube appearance and motion features given by the 3D HoG descriptor. We define the set $\mathcal{E}$ as:

$$\mathcal{E} = \bigcup \; (t_a, t_b) \;\mid\; d(H_a, H_b) < \theta, t_a \in V_1, t_b \in V_2 \tag{7}$$

where H_a, H_b refer to the aggregation of the descriptor vectors of tubes t_a and t_b calculated from sampled 3D HoG descriptors, as explained in Section 3.3. The function d is any distance metric between descriptor vectors, and θ is a threshold which defines what is a *good* correspondence. In this work we use the Euclidean distance.

The prior can also include terms that encourage a smooth labeling in local video areas (P^{smooth} in Fig. 3) defined analogously as the term P^{match} and P^{coh} of Eq. (5) and (6). This is a typical approach in the segmentation literature which usually helps to reduce noise and improves labeling consistency.

5 Experiments

In this section we show quantitative results on a subset of the Chroma database [17], in 4 different sequences with the same foreground objects but different backgrounds. In a second experiment we show qualitative results on other typical benchmark videos from the video segmentation literature.

5.1 Optimization and Experimental Set-up

We choose to optimize the posterior distribution using graph cuts (step 7 in Algorithm 1), but in principle any other inference or optimization algorithm could be used as well. We apply the logarithm to the probability values and produce scalar unary and pairwise penalties for all possible variable realizations. Every pairwise term is sub-modular, as we only apply a high cost (low probability) on the configurations $[X_i \neq X_j]$. The high probability parameters $\alpha = \gamma$ are set to 0.9 and the low probability parameters $\beta = \delta$ are set to 0.1. The iterative process stops when the percentage of pixels that switch labels between two iterations is less than 2.5%, or the number of iterations exceeds the maximum (6). Finally, the threshold on the distance between tube descriptors that defines the set $\mathcal{E}$ is set to 0.4. We leave the smoothing term as an optional feature as we did not observe a significant improvement in our experiments, and we avoid setting extra parameters.

5.2 *Cha-cha-cha* videos

The Chroma database is possibly the only benchmark video segmentation dataset available that provides videos containing the same instance of an object (two *cha-cha-cha* dancers), which is particularly suitable for the co-sgementation problem.

The videos show two people dancing with two different synthetic animated backgrounds. Even though the foreground corresponds to the same object instance in all sequences, the videos are very challenging due to the high clutter and movement of the background and camera [17]. We apply our algorithm to the first 100 frames of the 4 videos of the *cha-cha-cha* class. In Figure 4 we show quantitative and qualitative results with respect to the initial labeling. . The background appearance and movement changes between different videos (one emulates zooming while others emulate camera translation). This difficulties the tube matching, that performs poorly between pairs of videos with different

video	init	result
chacha1	0.57	0.61
chacha2	0.73	0.81
chacha3	0.45	0.56
chacha4	0.65	0.74

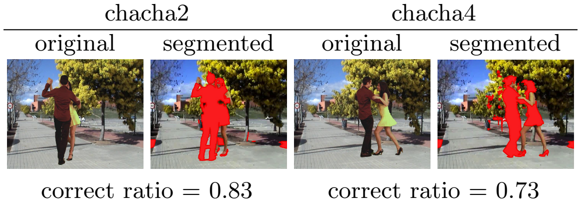

Fig. 4. Results and example frames on the chroma dataset. In the left, table with results of the four videos of the *chachacha* class. The central column shows the accuracy of the initialization, and the right column the final labeling. The score is calculated by counting the ratio of pixels correctly labeled.

background. Videos 1 and 3 show a significant lower segmentation accuracy than videos 2 and 4 due to a wrong initialization. This is mainly caused by the background appearance features that confuse the objectness and saliency measures.

5.3 Videos from heterogeneous sources

Our video co-segmentation method requires several videos depicting an object with similar appearance and motion. Most benchmarking databases do not contain this type of data. Therefore, we selected videos from existing databases and collected other videos from Youtube with similar characteristics in order to perform co-segmentation. We use one video from the *segTrack* [6] database (*parachute*) and two from the set provided in [10] (*kite surfing* and *ice dancer*). These videos were chosen because similar videos, in terms of appearance and motion of the foreground elements, were available on Youtube.

Figure 5 shows frame examples of the video results, compared to the works of [10] and [8]. Unfortunately no ground-truth is available in most of the videos, so we have limited the evaluation to a qualitative perspective.

The performance of our method is comparable to the state-of-the-art results presented in [8]. However, the segmentation accuracy greatly depends on the motion and appearance consistency of the input set, as well as on the quality of the initial labeling. Our method requires good initializations in at least some of the video inputs, so that the learnt foreground/background model is sufficiently representative. If every input video is badly initialized, the method performs poorly. In cases where the videos downloaded from Youtube do not contain objects with enough motion and appearance similarity, the performance is also hampered, and generally decreasing with every iteration, because the prior constraints are not able to prevent the foreground from expanding or contracting.

The sets of input videos for each benchmarking sequence are shown in the left column of Figure 5. We observe noise and segmentation artifacts in some of our result videos (around the kite surfer). This could be avoided applying a local smoothing on the label values. However the performance does not change dramatically due to this constraint, as the foreground appearance model already encourages the labeling to *cluster* around local areas with high likelihood appearance similarity.

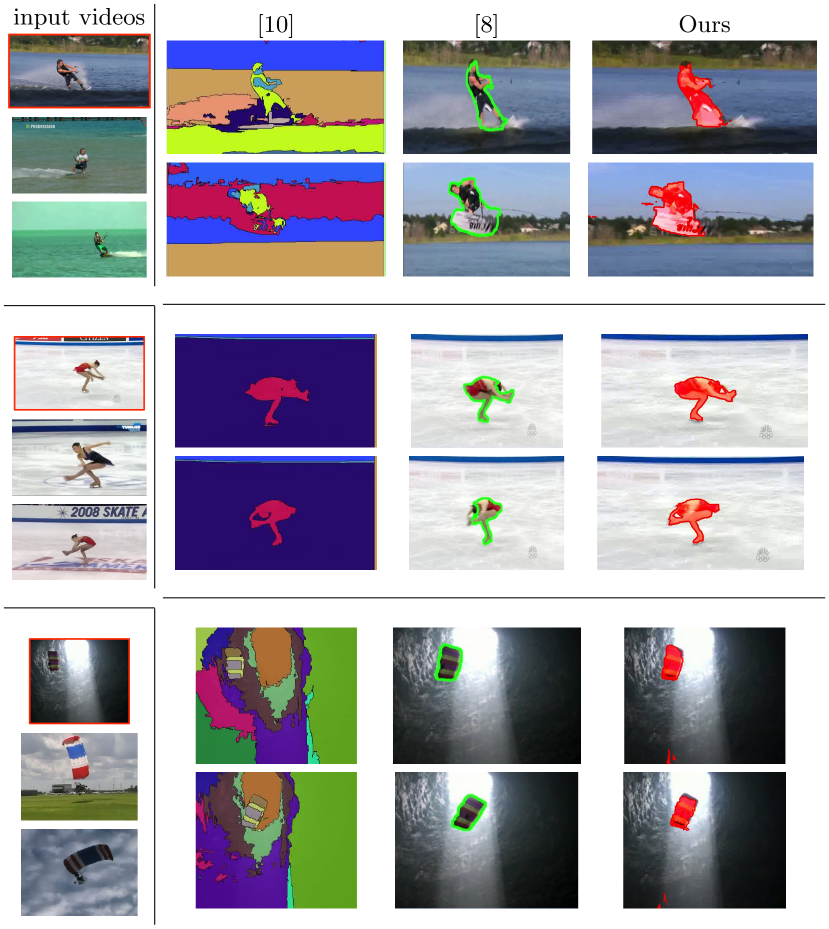

Fig. 5. Results on three videos from SegTrack [6] and [10]. For each of these videos (framed by a red border), two other videos were downloaded from Youtube for the purposes of co-segmentation. One sample frame from each input video is shown in the left column. In the right column we show examples of two frames for each of the result videos. We compare our method with the results obtained by [10] and[8].

6 Conclusions

In this work we have presented a novel non-supervised video co-segmentation algorithm. To the best of our knowledge, it is the first method that applies the concept of co-segmentation to video, understood as gathering information from several sources in order to jointly separate foreground and background. We introduce a two layered multi-image model that labels video volumes and image regions simultaneously, by iteratively learning and updating the foreground and background distributions built over motion and appearance features. We provide experimental validation on a subset of benchmarking video segmentation videos.

We also introduce the problem of co-segmentation using heterogeneous video sources. Our method proves to be qualitatively comparable to state-of-the-art results, although our validation is limited to cases where additional input videos with similar motion and appearance are available.

In the future we would like to generate and release ground-truth on video co-segmentation in order to improve experimental validation for this problem. We would also like to explore the application of video co-segmentation to other computer vision tasks like action recognition or video retrieval.

Acknowledgement This work was supported by the Spanish Ministry of Education and Science, under project TRA2011-29454-C03-01 and the Research Program Consolider Ingenio 2010: MIPRCV (CSD2007-00018).

References

1. Levinshtein, A., Sminchisescu, C., Dickinson, S.J.: Spatiotemporal closure. In: ACCV. (2010)
2. Levinshtein, A., Sminchisescu, C., Dickinson, S.J.: Optimal contour closure by superpixel grouping. In: ECCV. (2010)
3. Sundaram, N., Keutzer, K.: Long term video segmentation through pixel level spectral clustering on gpus. In: ICCV Workshops. (2011)
4. Maire, M., Arbelaez, P., Fowlkes, C., Malik, J.: Using contours to detect and localize junctions in natural images. In: CVPR. (2008)
5. Li, Y., Sun, J., Shum, H.Y.: Video object cut and paste. ACM Trans. Graph. (2005)
6. Tsai, D., Flagg, M., Rehg, J.M.: Motion coherent tracking with multi-label mrf optimization. In: BMVC. (2010)
7. Criminisi, A., Cross, G., Blake, A., Kolmogorov, V.: Bilayer segmentation of live video. In: CVPR (1). (2006)
8. Lee, Y.J., Kim, J., Grauman, K.: Key-segments for video object segmentation. In: ICCV. (2011)
9. Huang, Y., Liu, Q., Metaxas, D.N.: Video object segmentation by hypergraph cut. In: CVPR. (2009)
10. Grundmann, M., Kwatra, V., Han, M., Essa, I.A.: Efficient hierarchical graph-based video segmentation. In: CVPR. (2010)
11. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. International Journal of Computer Vision (2004)
12. Rubio, J.C., Serrat, J., López, A.M.: Unsupervised co-segmentation through region matching. In: CVPR. (2012)
13. Alexe, B., Deselaers, T., Ferrari, V.: What is an object? In: CVPR. (2010)
14. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE Trans. Pattern Anal. Mach. Intell. (1998)
15. Kläer, A., Marszalek, M., Schmid, C.: A spatio-temporal descriptor based on 3d-gradients. In: In BMVC08. (2008)
16. Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: CVPR. (2010)
17. Tiburzi, F., Escudero, M., Bescós, J., Sanchez, J.M.M.: A ground truth for motion-based video-object segmentation. In: ICIP. (2008) http://www-vpu.ii.uam.es/CVSG/.