Video Object Segmentation by Salient Segment Chain Composition

*Dan Banica[1], *Alexandru Agape[1], Adrian Ion[2], Cristian Sminchisescu[3,1]
[1]Institute of Mathematics of the Romanian Academy
[2]Vienna University of Technology, [3]Lund University
{dan.banica, alexandru.agape}@imar.ro, ion@prip.tuwien.ac.at, cristian.sminchisescu@math.lth.se

Abstract

We present a model for video segmentation, applicable to RGB (and if available RGB-D) information that constructs multiple plausible partitions corresponding to the static and the moving objects in the scene: i) *we generate multiple figure-ground segmentations, in each frame, parametrically, based on boundary and optical flow cues, then track, link and refine the salient segment chains corresponding to the different objects, over time, using long-range temporal constraints;* ii) *a video partition is obtained by composing segment chains into consistent tilings, where the different individual object chains explain the video and do not overlap. Saliency metrics based on figural and motion cues, as well as measures learned from human eye movements are exploited, with substantial gain, at the level of segment generation and chain construction, in order to produce compact sets of hypotheses which correctly reflect the qualities of the different configurations. The model makes it possible to compute multiple hypotheses over* both *individual object segmentations tracked over time,* and *for complete video partitions. We report quantitative, state of the art results in the SegTrack single object benchmark, and promising qualitative and quantitative results in clips filming multiple static and moving objects collected from Hollywood movies and from the MIT dataset.*

1. Introduction

Video segmentation, in full generality, is the problem of partitioning a video into several spatio-temporal volumes, region chains, or tubes. While such a definition makes the connection to space-time clustering natural, it is less attractive when the objective is to identify the spatial support of the important static and moving objects of a scene, over time. Central to this relevant special case–with the promise to open path to semantic video analysis and categorization–is not so much just the capacity to partition the video somehow, without substantial control over the properties of the regions being generated, but being able to focus on important salient structures and persistently identify them over time. This becomes significantly harder than being successful in any single image or frame of the video (the special problem of image segmentation), as the probability of a 'temporal lucky strike' becomes diminishingly low. Scenes with not just one, but multiple static or moving objects, further drive down probabilities of generating good quality partitions, which are salient for human perception, and thus, for effective object learning and categorization. In this paper we propose a compositional approach to video partitioning within a framework that offers consistent inference for multiple video interpretations. Our segment and chain measures combine the notion of trained figural and motion Gestalts as well as the one of saliency based on human fixations. We show that such an approach generalizes well and offers a strong signal to separate and consistently rank spatial and temporal chains that are perceptually relevant (in their object coverage), from ones that are not. Technically our methodology relies on per-frame segment generation at multiple scales, using boundary and differential motion cues (optical flow), within a parametric max flow framework, followed by dense matching across frames by combining long-range search in order to compute stable and salient segment chains (SSC), and jointly refining such chains using appearance and location cues. Finally SSCs are combined into complete video interpretations based on techniques that rely on generating multiple cliques in a graph where the SSCs are nodes and the connections are drawn between spatially non-overlapping components. This approach produces state of the art results in the SegTrack dataset [28], for single object segmentation, and promising results in clips containing both static and moving objects, from the MIT dataset [16], and in Hollywood clips.

Related Work. There is substantial prior work for figure-ground video segmentation, as well as for generating spatio-

*The first two authors contributed equally

temporal superpixel partitions, or for motion segmentation. Segmenting a *single foreground object from a video* has been approached using initialization by a user [27, 8], or automatically, by producing a pool of video object hypotheses [12, 17, 10]. A possible pipeline includes a figure-model extraction using color, appearance, shape, and motion cues, derived from a prior clustering of segments over the entire video, and a final energy minimization in a binary MRF [12]. Alternatively a maximal weighted clique framework was used to optimally link segments in each frame [17], but the mutual exclusion constraint used allows only one segment to be selected in each timestep, and does not provide multiple scene layout interpretations. While these approaches can produce good quality video segmentations of single moving objects, computing a video segmentation for the multiple object case is not straightforward, as the generated segmentations are likely to overlap and vary in quality. Approaches to *multiple object video-segmentation* involve clustering at the level of pixels or superpixels, using color and motion cues [9, 29, 14] or long-term trajectories [15]. These encounter difficulties due to the small spatial support used to extract features, leading to over segmentations. To consider larger spatial support when extracting features, [23, 22] compute multiple segment hypotheses independently for each frame, then [23] solve for the final video segmentation directly. In contrast [22] compute single-object video segment hypotheses over a few frames and combine those into the final video segmentation using soft constraints as high order terms in an MRF. Both methods lack a notion of saliency to drive the selection process; while multiple hypotheses for image segmentation are obtained in each frame, a single video partition is generated. To avoid the non-overlap constraints between segments, but still have appropriate spatial support to build object models, [4] request hand-labeling the first and last frames, then used for the initial model extraction, and to propagate those labels in an MRF. Alternatively, [20] compute long term trajectories of detected feature points, cluster them into object hypotheses, and propagate the corresponding labels form pixels on the trajectory to the rest of the video.

Differently from existing work, our video segmentation framework is automatic and applies to both static and moving objects. We focus on multiple figure-ground hypotheses (as opposed to superpixels), extend CPMC [6] with optical flow and (if available) depth, and integrate appearance, motion, and saliency information, over all processing steps. Our SSC generation approach tracks segments using dense matching and long-range constraints, rather than clustering segments over the entire video [12] or using a maximum weight clique (MWC) [17] to solve segment competition. We enforce the consistency (non-overlap condition) between the individual SSC hypotheses as hard constraints in a MWC framework, as opposed to soft-constraints in [22][1]. We also provide an inference process for multiple video segmentation hypotheses, as opposed to just one.

2. Salient Segment Chain Construction

Given a collection of frames $\{f_1, \ldots, f_N\}$ of a video, a *Salient Segment Chain (SSC)* is a set of $K > 0$ segments $c = \{s^{t_1}, ..., s^{t_K}\}$ corresponding to K consecutive frames $f_{t_1}, \ldots, f_{t_K}$, with each $s^t \in c$ a segment of frame f_t. To build SSCs we **(1) generate** for each frame f_t a pool of object segment hypotheses $\mathcal{S}_t$, **(2) match and link** the segments corresponding to all pairs of consecutive frames, to obtain an extended pool of *coarse SSCs* that use only segments from $\mathcal{S} = \cup \mathcal{S}_t$, **(3) rank** the coarse SSCs using low and mid-level features, retaining the top scoring, and finally (4) **refine** each of the retained SSCs using appearance and location cues.

1. Segment Pool Generation, $\mathcal{S}_t$. To generate the segment pool $\mathcal{S}_t$ for frame f_t we extend the Constrained Parametric Min Cuts method (CPMC) [6]. CPMC generates a large pool of figure-ground segmentations by applying constraints at different locations and scales in the image (frame), scores them using mid-level properties and returns the top ranked following diversification. Segments are generated by solving a family of optimization problems for energies of the form:

$$E^{\lambda}(L) = \sum_{l_x \in L} D_{\lambda}(l_x) + \sum_{l_x \in L, l_y \in \mathcal{N}(l_x)} V_{xy}(l_x, l_y) \quad (1)$$

where L is a labeling of the pixels of the image into foreground or background, $\lambda \in \mathbb{R}$ selects the problem instance to be solved, the unary term D_{λ} defines the cost of assigning a particular pixel to the foreground or the background, and the pairwise term V_{ub} penalizes the assignment of different labels to similar neighboring pixels. To leverage motion and learned saliency cues, we extend CPMC in multiple ways: we add motion seeds to constrain certain pixel-labels to foreground, augment the smoothness constraint in eq. 1 with optical flow, and extend the features used for scoring and ranking segments with Gestalt features extracted on optical flow.

CPMC uses simple, content independent strategies to place foreground and background seeds. For the segmentation of moving objects, we cluster the optical flow vectors computed using [3], and place additional foreground seeds for each connected component corresponding to a cluster. The original pairwise term V_{xy} in eq. 1 adds the penalty $g(x, y) = \exp\left[-\frac{\max(B_{\mathcal{I}}(x), B_{\mathcal{I}}(y))}{\sigma^2}\right]$ if two neighboring

[1] Hard constraints have been previously used for tracking [1], with focus on object trajectories, where they were enforced on bounding-boxes, between consecutive frames only, rather than on segments and over an entire video.

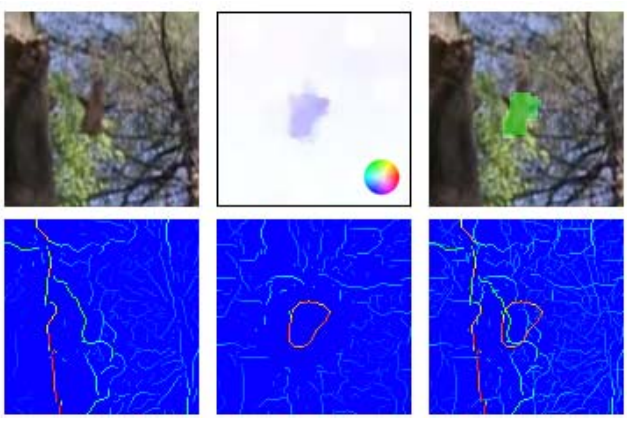

Figure 1: Impact of adding motion information to the smoothness term in CPMC. Detail from the first frame of the *birdfall* sequence used in our experiments. First row, left to right: original image, optical flow, and *best* segment obtained for the ground-truth object, a falling bird. Second row: contour detector applied on the original image and on the optical flow representation, followed by their combination using max. Strong edges are shown in red. Notice that the strong contours detected directly on the image do not highlight the ground-truth object, while the motion-based contours provide a good estimate of its boundary.

pixels x, y are assigned different labels, where $B_{\mathcal{I}}$ is the output of a trained contour detector [13, 18] computed for the image $\mathcal{I}$ at a given pixel. We modify this term to use motion information and define the augmented penalty $g'(x,y) = \exp\left[-\frac{\max(B_{\mathcal{I}}(x),B_{\mathcal{I}}(y),B_{\mathcal{OF}}(x),B_{\mathcal{OF}}(y))}{\sigma^2}\right]$, where $B_{\mathcal{OF}}$ is the output of a global contour detector [13, 18] computed on the HSV representation of the optical flow (see also fig. 1). Another extension that we propose here is to use depth data when available, which may be acquired through devices such as Kinect. In order to take advantage of depth information we maximize in the formula for $g'(x,y)$ over two additional terms: $B_{\mathcal{D}}(x)$ and $B_{\mathcal{D}}(y)$, where $B_{\mathcal{D}}$ represents the output of the contour detector computed on the depth image. Fig. 2 illustrates how better segment pools are obtained using **CPMC-3D**, our CPMC extension which uses depth information.

Segment scores are originally computed based on a combination of graph partition, region and Gestalt properties, with corresponding parameters trained, category independently, on object segments computed from static images. We adjust these scores using two features that are sensitive to video inputs. The first one uses a saliency detector that predicts human fixations [19], trained on the Hollywood dataset. The feature value for each segment is set to the average detector responses for all pixels in that segment.

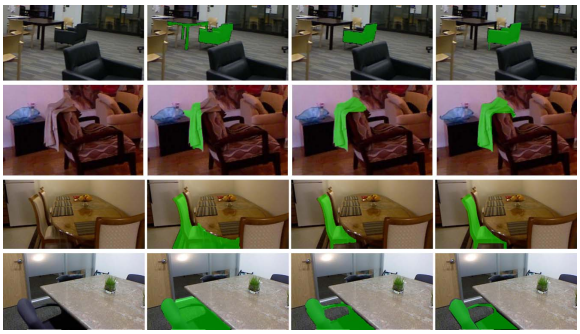

Figure 2: Using depth cues notably improves CPMC segment quality. Left to right: original image, best segment obtained from CPMC, best segment from **CPMC-3D**, ground truth. The images are from the NYU Depth V2 dataset [24].

The second feature is obtained by computing Gestalt features in [6] on the HSV optical flow representation. These features highlight regions with salient motion, and penalize segments with parts that move similarly to the background.

The final segment score is a linear combination of the original image, saliency and flow-based scores:

$$q_s(s) = \alpha_I \cdot I(s) + \alpha_S \cdot S(s) + \alpha_F \cdot F(s) \qquad (2)$$

Running the modified CPMC model, which we denote **CPMC-OF** (or **CPMC-3D** and **CPMC-OF-3D** if depth is available), we obtain for each frame f_t, a set $\mathcal{S}_t$ of segment hypotheses with scores $q_s(s)$.

2. Segment Modeling, Matching and Linking. We generate a large pool of coarse SSCs, each constructed using only segments from $\mathcal{S} = \cup\mathcal{S}_t$. A total of $|\mathcal{S}|$ SSCs are produced, one corresponding to each segment in $\mathcal{S}$. The term *coarse* is used to differentiate from the *refined* SSCs, computed later, which are not restricted to segments in $\mathcal{S}$. Coarse SSCs correspond to paths in a *trellis*, where each node (segment) at time t is connected with (all) nodes at times $t-1$ and $t+1$. We generate SSCs by starting at each segment s_i^t [2] and growing in both directions as described below.

Each segment is modeled using its position (1st order moment), size and LAB color histogram. The pairwise distance $d(s_i, s_j)$ between two segments s_i, s_j is computed as:

$$d(s_i^{t_k}, s_j^{t_l}) = d_r(s_i^{t_k}, s_j^{t_l}) + d_c(s_i^{t_k}, s_j^{t_l}) \qquad (3)$$

$$d_r(s_i^{t_k}, s_j^{t_l}) = \sum_p \lambda_p \cdot \sigma(|p(s_i^{t_k}, s_j^{t_l})|, \tau^p_{|t_k - t_l|}) \qquad (4)$$

$$\sigma(x,\tau) = \frac{1}{1+\exp(-\frac{x-\tau}{\tau})} \qquad (5)$$

[2] The superscript t is used to denote the frame number, and the subscript the index in $\mathcal{S}_t$. We might omit any of them, if clear from context.

where $d_c(s_i^{t_k}, s_j^{t_l})$ is the Euclidean distance between the normalized LAB histograms of the two segments, and d_r sums over normalized region properties p (area, position of the centroid, height and width of the bounding box of the segment) a sigmoid function on the difference between the properties of the two segments. This sigmoid is used to heavily penalize changes in shape or position above a threshold $\tau^p_{|F_i - F_j|}$ that depends on the distance between the frames of the segments and on the considered property. Overall, the distance defined in eq. 3-5 is small for segments with similar LAB color histograms, whose size and position did not change too much over a few frames.

Given an SSC $c = \{s^{t_1}, \ldots, s^{t_K}\}$ the distance of a segment $s_j \in f_{t_K+1}$ to c is given by:

$$D(s_j, c) = \texttt{rmean}\left(d(s_j, s^{t_K-m+1}), \ldots, d(s_j, s^{t_K})\right) \quad (6)$$

where rmean stands for *robust-mean* and equals the mean, after removing the largest and smallest values in the list. For $s_j \in f_{t_1-1}$, $D(s_j, c) = \text{rmean}\,(d(s_j, s^{t_1}), \ldots, d(s_j, s^{t_m}))$. Using segments from more than one time-step and the rmean function increases robustness against segmentation errors in hard-to-segment frames where the set $\mathcal{S}$ might not contain a segment to match.

To build an SSC, we initialize with a segment s_j^t and grow at both ends by greedily adding the segments in f_{t_1-1}, f_{t_K+1} to minimize D. The indexes t_1, t_k are updated to always match the first and last frame indexes in the SSC. If the minimizing value is above a threshold d_0, growing in that direction is abandoned and the segmented object is assumed to have disappeared from the scene or become fully occluded.

Given our trellis structure, other inference strategies for generating SSCs, can also be envisioned. For $m = 1$ our model has only pairwise terms and the Viterbi algorithm can efficiently find a global optimum for each initialization segment s_j^t. However, most of the times this optimal solution contains high scoring sub-chains corresponding to different objects, which are connected by a few weak links. This leads to poor segmentation results and very similar SSCs, each one also very close to the globally optimal SSC obtained without imposing a particular segment s_j^t. Increasing the penalty incurred when selecting a poor link can partly mitigate this problem, but choosing an appropriate threshold can be problematic. The reason for choosing a greedy strategy is because of low computational complexity when the number m of past frames considered is larger than 1. The model can also be made more powerful by including long-range terms.

3. Coarse SSC Scoring and Ranking. Each coarse SSC c is associated a score given by:

$$q_c(c, \alpha) = \sum_{s^t \in c} \alpha_u q_u(s^t) + \sum_{s^t, s^{t+1} \in c} \alpha_p q_p(s^t, s^{t+1}) \quad (7)$$

where the unary term $q_u(s^t)$ is the segment score given by eq. 2, and the pairwise term q_p is the matching score in eq. 6. To make the matching scores in q_p independent of the order in which segments where initially added to c, this is recomputed for all segments in c as if they were all added in chronological order.

To select a set of high quality SSCs, we first rank all coarse SSCs in decreasing order of their score q_c in eq. 7. However, if they exist, very similar SSCs are assigned similar scores by q_c and end up in consecutive or close positions in the ranked SSC list. Our aim is to obtain a set of high quality and diverse SSCs, and address this issue by re-ranking them using a Maximal Marginal Relevance (MMR) measure [5], using per frame average segment overlaps as the redundancy measure. This leads to an increase of the highest quality when considering top k ranked SSCs, *e.g.* for $k = 200$ by 50% on the 'cheetah' video we used in experiments. Finally, we retain the top scoring 150 SSCs, as ranked by the MMR measure.

4. SSC Refinement. The retained coarse SSCs cover the objects in the video reasonably well, but may be sensitive to the quality of the initial segments in $\mathcal{S}$. However, even for cases where in certain frames no appropriate segment is generated for an object, the matching strategy with $m > 1$ could in principle overcome this issue, to produce high quality SSCs. The segmentation in difficult frames can then be improved by propagating information from neighboring frames in the SSC. Therefore, for each retained coarse SSC c_c we compute a refined SSC c_r over the same video frames as c_c, but not constrained to include the original segments in $\mathcal{S}$. Each segment $s'^t \in c_r$ is obtained from s'^{t-1} and s^t by solving a binary labeling problem:

$$L^* = \underset{L}{\text{argmax}} \sum_{x \in f_t} \phi_u(l_x) + \sum_{x \in f_t, y \in \mathcal{N}(x)} \phi_p(l_x, l_y) \quad (8)$$

where $L \in \{0,1\}^{|f_t|}$ is a labeling of the pixels of f_t, ϕ_u, ϕ_p are unary and pairwise potentials, and the neighborhood $\mathcal{N}(x)$ contains the 4-connected neighbors of x. The unary potential $\phi_u(l_x)$ is computed from appearance (color) and location priors. The color prior is obtained using Gaussian mixture models. The pixels from the refined segment in the previous frame are used as foreground samples. The location prior is generated using an Euclidean distance transform on the union of s^t and the projection of s'^{t-1} in frame t. To refine segments in the first frame we define $s'^1 = s^1$ and consider s^1 as its projection. The pairwise potential $\phi_p(l_x, l_y)$ is a contrast dependent smoothing term similar to $V_{xy}(l_x, l_y)$ in eq. 1, but using the union between the boundaries given by a global contour detector [13, 18] and the optical flow boundaries. Fig. 3 shows examples of these potentials and the segments that generated them, together with the final optimization result. s'^t is taken to be the pixels in L^* that have been assigned the foreground label 1.

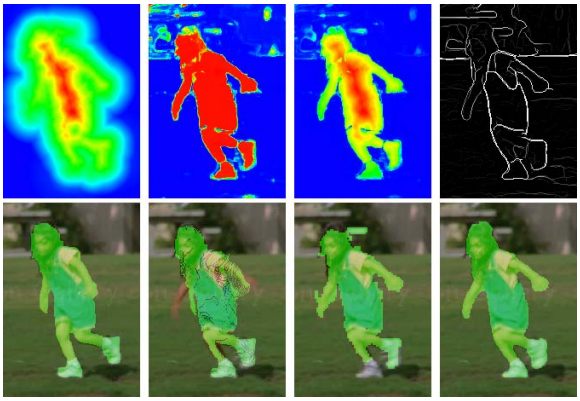

Figure 3: Visualization of our SSC refinement process. First row, left to right: location prior, color prior, foreground prior, and boundaries used in defining pairwise terms. Second row: segment in frame $t-1$ used to generate the color model, projection of this segment to frame t, the segment from the coarse SSC corresponding to frame t, and the final refined segment in frame t.

3. Segment Chain Composition

At this point we have computed a pool of refined SSCs, each corresponding to an object from the video, throughout its lifespan. We define a *Video Partition (VP)* as the set v of refined SSCs such that (i) no two SSCs in v overlap, and (ii) v cannot be extended using SSCs from the pool. We further define a potential function over the set $\mathcal{V}$ of all possible VPs as:

$$\Gamma_\beta(v) = \sum_{c \in v} \beta_u^\top \gamma_u(c) + \sum_{c \in v, c' \in \mathcal{N}(c)} \beta_p^\top \gamma_p(c, c') \qquad (9)$$

where $\beta = [\beta_u^\top \beta_p^\top]^\top$ are the parameters of the model, the neighborhood $\mathcal{N}(c)$ contains all refined SSCs that have neighboring segments with c in at least one frame (i.e. segments that would overlap when dilated a few pixels), and γ_u, γ_p are unary and pairwise potentials, respectively. The unary potentials capture the individual qualities of the SSCs, $\beta_u^\top \gamma_u(c) = q_c(c, \alpha)$ (see eq. 7). The pairwise potentials capture the affinity of different SSCs and make use of per frame individual segment affinities, using the measure defined in [11]. This measure takes into account joint region properties such as relative area, position and orientation, as well as features to signal occlusion boundaries. Inference is cast as optimization over maximal cliques, in a graph connecting all non-overlapping refined SSCs. Multiple solutions are computed and ranked by their scores $\Gamma_\beta(v)$. See [11] for details on the inference procedure.

4. Experiments

We evaluate the proposed framework on videos from the SegTrack database [28], the MIT dataset [16], and Hollywood movies. SegTrack contains videos with a single moving foreground-object annotated, and is used to quantitatively evaluate our SSC generation. No existing dataset provides annotation for several moving and static objects, as appropriate for the computed VPs [3]. We thus use clips with multiple static and moving objects from the MIT dataset [16] (only a few objects are annotated) and Hollywood movies, to report quantitative and qualitative results for the multi-object case. To sum up, we use 6 videos from the SegTrack dataset [4], 9 videos from the MIT dataset, and 2 short clips from Hollywood movies, more than 600 frames in total.

The validated parameter values are: in eq. 2, $\alpha_I = 1$, $\alpha_S = 1.2$, $\alpha_F = 3$; in eq. 4, $\tau_f^p = 0.1$, $\lambda_p = 10$ for p = normalized area, respectively $\tau_f^p = 0.1 \cdot f$, $\lambda_p = 5$, for the other normalized region properties p (centroid, width and height); when constructing SSCs, in eq. 6, we consider $m = 5$ segments from nearby frames; a large threshold d_0 was used, imposing for the SSCs to span over the entire video-sequence. For eq. 7 and eq. 9, $\alpha_u = 1$, $\alpha_p = -0.5$, $\beta_u = [\alpha_u \alpha_p]^\top$, $\beta_p = 1$.

In the following we discuss the main steps of our framework (see sec. 2 and 3 for details). We report results using pixel-error (number of pixels that have been wrongly labeled as foreground/background, smaller is better), overlap (intersection over union, larger is better) and RandIndex [21]. When selecting from several candidates, *first* corresponds to quality of the highest ranked configuration that has an overlap with the ground-truth larger than 0.5 (the same methodology as [12]), *best* corresponds to the highest quality configuration.

Segment generation with CPMC-OF. Table 1 shows results when including motion cues (CPMC-OF), compared to using only per frame static appearance information (CPMC). *E.g.* for the difficult *birdfall* video from SegTrack (see also fig.1), the *Best* segment computed for the bird, without motion cues (CPMC), has a rather small overlap score of 0.31, because of the small foreground object without a strong boundary. Adding motion information leads to a significant improvement.

Ranking segments and SSCs. Fig. 4 shows segment ranking performance, when using only static, motion, saliency, as well as all possible features. The latter achieve superior results. Table 2 shows results for *first* coarse and refined SSCs, using only one of the three types of cues (static, motion, and saliency), as well as all of them. When using

[3] The Chen Xiph.org dataset [7] provides only semantic ground-truth, with neighboring objects of the same class indistinguishable. The Berkeley Motion Segmentation dataset [2] considers only moving objects. The MIT dataset [16] focuses on objects with distinct motion patterns, having annotations only for the static objects that are distinguishable due to camera motion.

[4] Following the standard practice, we do not compare on 'penguin'.

Sequence	birdfall	cheetah	girl	monkeydog	parachute	penguin
CPMC	0.31(229)	0.54(28)	0.77(151)	0.59(60)	0.90(60)	0.58(232)
CPMC OF	0.61(909)	0.64(298)	0.83(651)	0.64(415)	0.90(315)	0.70(672)

Table 1: Results of CPMC vs. **CPMC-OF** on the SegTrack dataset. For both methods we give averages over all frames, of the overlap score of our *Best* segment in each frame (higher is better). In brackets we give the number of segments in the generated pools. The increase in seeds and boundaries makes the parametric energy in eq. 1 have more breakpoints, which results in a larger number of segments produced by **CPMC-OF**. See the text for descriptions of measures.

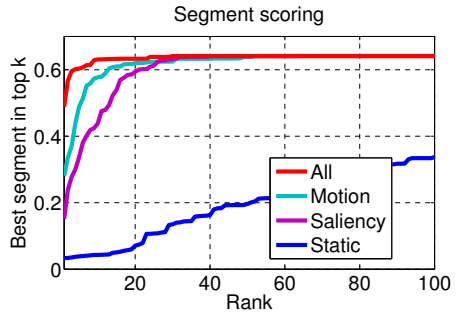

Figure 4: Average overlap score for the *Best* segment, when retaining the top k ranked segments, on the "cheetah" video. Saliency and optical-flow features are informative for scoring and ranking segments. Including all three types of cues induces the *best* ranking.

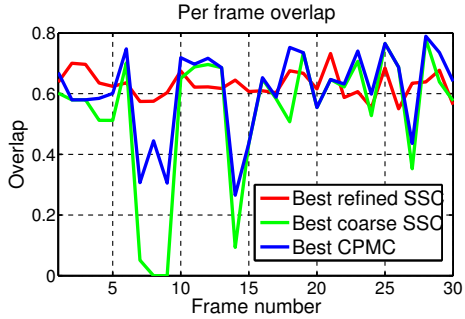

Figure 5: For each frame of the "birdfall" sequence, the overlap score of the *Best* CPMC segment, the CPMC segment included in the *best* coarse SSC (Best coarse SSC), and that segment after refinement in the corresponding refined SSC (Best refined SSC).

all three ranking features the quality of the predicted *first* coarse SSCs is very close to the quality of the *best* existing SSCs. This evaluation uses the entire, coarse SSCs pool (thousands of chains), and the pool of refined SSCs, generated from the reduced set of 150 high scoring coarse SSCs.

SSC generation and refinement. Fig. 5 shows for the 'birdfall' sequence, the overlap score of the generated *best* segment in each frame, and the overlap scores of the segments included in the *best* coarse SSC, and refined in the corresponding refined SSC. Considering $m = 5$ neighboring frames when growing SSCs helps deal with frames for which CPMC-OF fails to generate good quality segments. Based on the predicted *best* coarse SSC, the refinement improves the SSC beyond the segments in the pool $\mathcal{S}$.

Single-object segmentation. The computed refined SSCs correspond to segmentation hypotheses for a single object, and can be evaluated as such. Table 3 shows results for the SegTrack single-object video segmentation dataset. The proposed method achieves state-of-the art results in 4 out of 5 videos and is competitive on the fifth.

Video Partitions for Static and Moving Objects. Table 4 gives quantitative results obtained on the MIT dataset [16]. The video partition inference mechanism encourages selecting SSCs with good mutual affinity (*e.g.* occlusion boundaries), which also have object-like characteristics, as estimated using mid-level static descriptors as well as dynamic features. Fig. 6 shows visual results for multi-object video segmentation, in several sequences taken from the Hollywood-2 and MIT datasets. Notice that our method can handle *both* static *and* moving objects.

5. Conclusion

We have presented a compositional approach to video segmentation. We have argued that in order for video segmentation to become useful for semantic video analysis, learning and object categorization, it has to be lifted from the status of a purely video clustering approach. Along these lines, we have introduced methodology that incorporates a learned notion of object saliency, both static and dynamic, propagated over time, in order to maximize the probability of identifying persistent, accurate segments, over many frames, and for both static and moving objects. The inference process in this framework provides multiple hypotheses at all processing stages, and leads to state of the art results not only in the competitive benchmark SegTrack[28], but also in complex clips extracted from Hollywood films, and the MIT dataset[16].

	Coarse SSCs					Refined SSCs
	Static	**Saliency**	**Motion**	**All**	**Best**	**All**
birdfall	0.55 (129)	0.54 (4)	0.54 (2)	0.57 (**1**)	0.57	0.69 (1)
cheetah	0.51 (30)	0.57 (50)	0.53 (2)	0.53 (**1**)	0.59	0.60 (1)
girl	0.79 (3)	0.76 (2)	0.81 (1)	0.81 (**1**)	0.81	0.87 (1)
monkeydog	0.52 (496)	0.52 (376)	0.52 (473)	0.52 (**373**)	0.52	0.72 (10)
parachute	0.86 (2)	0.50 (7)	0.81 (1)	0.80 (**1**)	0.86	0.94 (1)

Table 2: The quality of *first* SSCs when ranking using one of static, saliency, and motion cues, or using all (All), for computed coarse and refined SSCs, on the SegTrack dataset. The numbers in brackets correspond to the actual position of the *first* SSC among the ranked SSCs. The scores for coarse SSCs use the large pool, before reduction, whereas the refined SSCs use the reduced pool of maximum 150. To show the advantage of using all cues, the score of the *Best* coarse SSCs is also shown.

	First	**First Coarse**	**Best**	**Best Coarse**	[17]	[12]	[28]
birdfall	**166**(1)	312	166	312	189	288	252
cheetah	**661**(1)	960	629	827	806	905	1142
girl	**1214**(1)	1714	1182	1714	1698	1785	1304
monkeydog	**394**(10)	1264	394	1264	472	521	563
parachute	218(1)	902	218	537	221	**201**	235

Table 3: Single-object video segmentation results on the SegTrack dataset [28]: our *First* and *Best* SSCs, as well as state-of-the art single-object video segmentation methods Key-seg [12], MWC [17], and MCT [28]. For the comparison we used the quality of our *First* segmentation. *First* selects one of the multiple segmentations using the predicted ranking following the methodology of [12]. The predicted rank is shown in parentheses. The used measure is pixel error (smaller is better), with the smallest score in each video marked in bold. The proposed method has the smallest error, with a margin of over 10%, in 4 out of 5 videos, and is competitive in the fifth. We also give the scores of the computed SSCs before refinement (prefix 'Coarse').

Acknowledgements This work was supported, in part, by CNCS-UEFISCDI, under PCE-2011-3-0438.

References

[1] W. Brendel, M. R. Amer, and S. Todorovic. Multiobject tracking as maximum weight independent set. In *CVPR*, 2011. 2

[2] T. Brox and J. Malik. Object segmentation by long term analysis of point trajectories. In *ECCV*, 2010. 5

[3] T. Brox and J. Malik. Large displacement optical flow: Descriptor matching in variational motion estimation. *PAMI*, 33(3), 2011. 2

[4] I. Budvytis, V. Badrinarayanan, and R. Cipolla. Semi-supervised video segmentation using tree structured graphical models. In *CVPR*, 2011. 2

[5] J. G. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, 1998. 4

[6] J. Carreira and C. Sminchisescu. Cpmc: Automatic object segmentation using constrained parametric min-cuts. *PAMI*, 34(7), 2012. 2, 3

[7] A. Y. Chen and J. J. Corso. Propagating multi-class pixel labels throughout video frames. In *WNYIPW*. IEEE, 2010. 5

[8] A. Fathi, M. F. Balcan, X. Ren, and J. M. Rehg. Combining self training and active learning for video segmentation. In *BMVC*, 2011. 2

[9] M. Grundmann, V. Kwatra, M. Han, and I. A. Essa. Efficient hierarchical graph-based video segmentation. In *CVPR*, 2010. 2, 8

[10] G. Hartmann, M. Grundmann, J. Hoffman, D. Tsai, V. Kwatra, O. Madani, S. Vijayanarasimhan, I. A. Essa, J. M. Rehg, and R. Sukthankar. Weakly supervised learning of object segmentations from web-scale video. In *ECCV Wks*, 2012. 2

[11] A. Ion, J. Carreira, and C. Sminchisescu. Image segmentation by figure-ground composition into maximal cliques. In *ICCV*, November 2011. 5

[12] Y. J. Lee, J. Kim, and K. Grauman. Key-segments for video object segmentation. In *ICCV*, 2011. 2, 5, 7

[13] M. Leordeanu, R. Sukthankar, and C. Sminchisescu. Efficient Closed-Form Solution to Generalized Boundary Detection. In *ECCV*, 2012. 3, 4

[14] A. Levinshtein, C. Sminchisescu, and S. Dickinson. Spatiotemporal Closure. In *ACCV*, 2010. 2

[15] J. Lezama, K. Alahari, J. Sivic, and I. Laptev. Track to the future: Spatio-temporal video segmentation with long-range motion cues. In *CVPR*, 2011. 2

[16] C. Liu, W. Freeman, E. Adelson, and Y. Weiss. Human-assisted motion-annotation. In *CVPR*, 2008. 1, 5, 6, 8

[17] T. Ma and L. J. Latecki. Maximum weight cliques with mutex constraints for video object segmentation. In *CVPR*, 2012. 2, 7

	car1	car2	car3	dog	phone	table	toy	hand	person
HGVS [9]	0.602	0.401	0.689	0.260	0.493	0.766	0.809	0.499	0.430
Layers++ [25]	0.612	0.512	0.778	0.964	0.567	0.909	0.832	0.814	0.986
nLayers [26]	0.836	0.589	0.766	0.974	0.578	0.979	0.858	0.881	0.944
Our VP	0.823	0.972	0.771	0.884	0.846	0.919	0.896	0.990	0.930

Table 4: Quantitative evaluation of our video partitions obtained on the MIT dataset [16], using the RandIndex measure [21]. Motion segmentation or layer-based methods such as [25] and [26] do not attempt to distinguish static objects in the background, which are also not annotated in the ground-truth of this dataset. Thus, before evaluation, we removed those SSCs that did not have overlaps larger than 0.5 with any annotated objects. *E.g.* only the printer and phone are annotated in the sequence shown in fig. 6, whereas our method segments both moving and static objects.

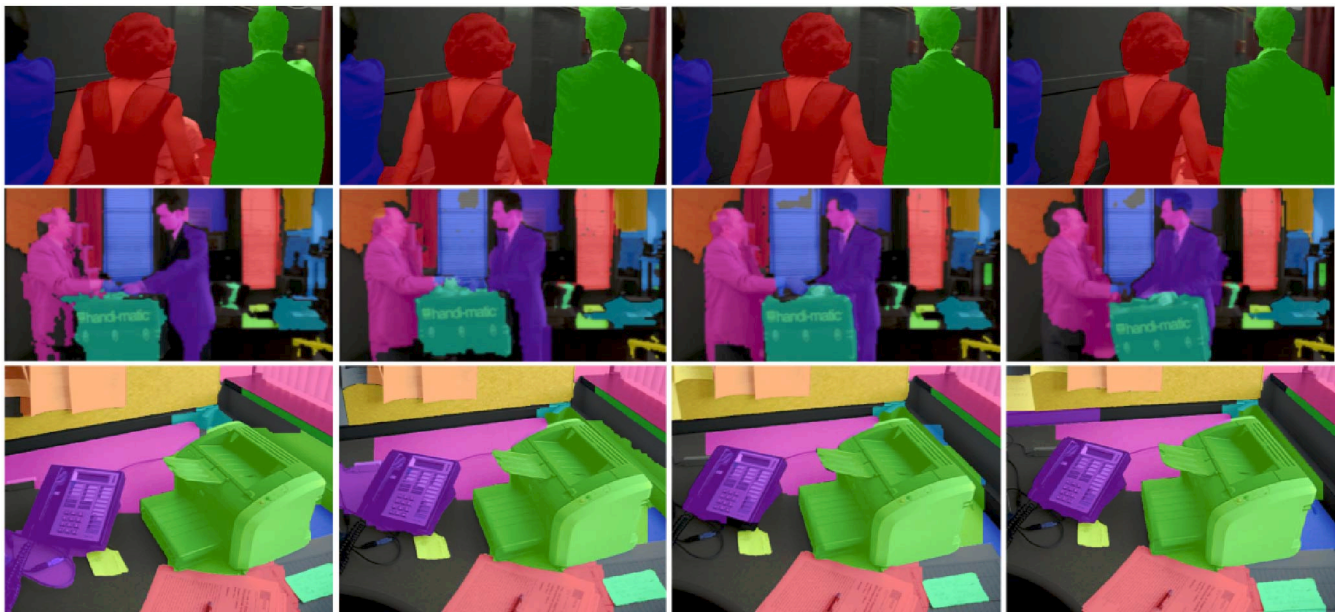

Figure 6: Visual results of our segmentations in scenes with multiple objects: top and middle row showing 'Goodfellas' and 'Big Fish' from Hollywood, bottom row 'phone' from the MIT dataset. The video partitioning mechanism correctly selected SSCs corresponding to different objects, both moving and static.

[18] M. Maire, P. Arbelaez, C. Fowlkes, and J. Malik. Using contours to detect and localize junctions in natural images. In *CVPR*, 2008. 3, 4

[19] S. Mathe and C. Sminchisescu. Dynamic Eye Movement Datasets and Learned Saliency Models for Visual Action Recognition. In *ECCV*, 2012. 3

[20] P. Ochs and T. Brox. Object segmentation in video: A hierarchical variational approach for turning point trajectories into dense regions. In *ICCV*, 2011. 2

[21] W. M. Rand. Objective criteria for the evaluation of clustering methods. *JAMSA*, 66(336):846–850, 1971. 5, 8

[22] A. V. Reina, S. Avidan, H. Pfister, and E. L. Miller. Multiple hypothesis video segmentation from superpixel flows. In *ECCV*, 2010. 2

[23] A. V. Reina, M. Gelbart, D. Huang, J. Lichtman, E. L. Miller, and H. Pfister. Segmentation fusion for connectomics. In *ICCV*, 2011. 2

[24] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 3

[25] D. Sun, E. B. Sudderth, and M. J. Black. Layered image motion with explicit occlusions, temporal consistency, and depth ordering. In *NIPS*, pages 2226–2234, 2010. 8

[26] D. Sun, E. B. Sudderth, and M. J. Black. Layered segmentation and optical flow estimation over time. In *CVPR*, pages 1768–1775. IEEE, 2012. 8

[27] D. Tsai, M. Flagg, A. Nakazawa, and J. M. Rehg. Motion coherent tracking using multi-label mrf optimization. *IJCV*, 100(2), 2012. 2

[28] D. Tsai, M. Flagg, and J. M. Rehg. Motion coherent tracking with multi-label mrf optimization. In *BMVC*, 2010. 1, 5, 6, 7

[29] C. Xu, C. Xiong, and J. J. Corso. Streaming hierarchical video segmentation. In *ECCV*. Springer, 2012. 2