Supervoxel-Consistent Foreground Propagation in Video

Suyog Dutt Jain and Kristen Grauman

University of Texas at Austin

Abstract. A major challenge in video segmentation is that the foreground object may move quickly in the scene at the same time its appearance and shape evolves over time. While pairwise potentials used in graph-based algorithms help smooth labels between neighboring (super)pixels in space and time, they offer only a myopic view of consistency and can be misled by inter-frame optical flow errors. We propose a higher order *supervoxel label consistency* potential for semi-supervised foreground segmentation. Given an initial frame with manual annotation for the foreground object, our approach propagates the foreground region through time, leveraging bottom-up supervoxels to guide its estimates towards long-range coherent regions. We validate our approach on three challenging datasets and achieve state-of-the-art results.

1 Introduction

In video, the *foreground object segmentation* problem consists of identifying those pixels that belong to the primary object(s) in every frame. A resulting foreground object segment is a space-time "tube" whose shape may deform as the object moves over time. The problem has an array of potential applications, including activity recognition, object recognition, video summarization, and post-production video editing.

Recent algorithms for video segmentation can be organized by the amount of manual annotation they assume. At one extreme, there are purely unsupervised methods that produce coherent space-time regions from the bottom up, without any video-specific labels [8, 12, 14, 17, 19, 21, 36, 38, 39]. At the other extreme, there are strongly supervised interactive methods, which require a human in the loop to correct the system's errors [4, 10, 20, 25, 34, 35]. Between either extreme, there are semi-supervised approaches that require a limited amount of direct supervision—an outline of the foreground in the first frame—which is then propagated automatically to the rest of the video [2, 3, 10, 27, 31, 33].

We are interested in the latter semi-supervised task: the goal is to take the foreground object segmentation drawn on an initial frame and accurately propagate it to the remainder of the frames. The propagation paradigm is a compelling middle ground. First, it removes ambiguity about what object is of interest, which, despite impressive advances [17, 19, 21, 39], remains an inherent pitfall for unsupervised methods. Accordingly, the propagation setting can accommodate a broader class of videos, e.g., those in which the object does not move much,

In Proceedings of the European Conference on Computer Vision (ECCV), 2014

or shares appearance with the background. Second, propagation from just one human-labeled frame can be substantially less burdensome than human-in-the-loop systems that require constant user interaction, making it a promising tool for gathering object tubes at a large scale. While heavier supervision is warranted in some domains (e.g., perfect rotoscoping for graphics), in many applications it is worthwhile to trade pixel-perfection for data volume (e.g., for learning object models from video, or assisting biologists with data collection).

Recent work shows that graph-based methods are a promising framework for propagating foreground regions in video [3, 10, 27, 31, 33]. The general idea is to decompose each frame into spatial nodes for a Markov Random Field (MRF), and seek the foreground-background (fg-bg) label assignment that maximizes both appearance consistency with the supplied labeled frame(s) as well as label smoothness in space and (optionally) time.

Despite encouraging results, these methods face an important technical challenge. In video, reliable foreground segmentation requires capturing *long-range* connections as an object moves and evolves in shape over time. However, current methods restrict the graph connectivity to local cliques in space and time. These local connections can be noisy: frame-to-frame optical flow is imperfect, and spatial adjacency can be a weak metric of "neighborliness" for irregularly shaped superpixels [1]. The failure to capture long-range connections is only aggravated by the fact that propagation models receive very limited supervision, i.e., the true foreground region annotated on the first frame of the video.

We propose a foreground propagation approach using *supervoxel* higher order potentials. Supervoxels—the space-time analog of spatial superpixels—provide a bottom-up volumetric segmentation that tends to preserve object boundaries [8,12,14,36,38]. To leverage their broader structure in a graph-based propagation algorithm, we augment the usual adjacency-based cliques with potentials for supervoxel-based cliques. These new cliques specify soft preferences to assign the same label (fg or bg) to superpixel nodes that occupy the same supervoxel. Whereas existing models are restricted to adjacency or flow-based links, supervoxels offer valuable longer-term temporal constraints.

We validate our approach on three challenging datasets, SegTrack [31], YouTube Objects [23], and Weizmann [13], and compare to state-of-the-art propagation methods. Our approach outperforms existing techniques overall, with particular advantage when foreground and background look similar, inter-frame motion is high, or the target changes shape between frames.

2 Related Work

Unsupervised video segmentation Unsupervised video segmentation methods efficiently extract coherent groups of voxels. Hierarchical graph-based methods use appearance and flow to group voxels [14, 38], while others group superpixels using spectral clustering [12] or novel tracking techniques [5, 32]. Distinct from the region-based methods, tracking methods use point trajectories to detect cohesive moving object parts [7,18]. Any such bottom-up method tends to preserve object boundaries, but "oversegment" them into multiple parts. As such, they are not intended as object segmentations; rather, they provide a mid-level space-time grouping useful for downstream tasks.

Several recent algorithms aim to upgrade bottom-up video segmentation to *object-level* segments [17, 19, 21, 22, 39]. While the details vary, the main idea is to generate foreground object hypotheses per frame using learned models of "object-like" regions (e.g., salient, convex, distinct motion from background), and then optimize their temporal connections to generate space-time tubes. While a promising way to reduce oversegmentation, these models remain fully unsupervised, inheriting the limitations discussed above. Furthermore, none incorporates higher order volumetric potentials, as we propose.

Interactive video segmentation At the other end of the spectrum are interactive methods that assume a human annotator is in the loop to correct the algorithm's mistakes [4,20,25,35], either by monitoring the results closely, or by responding to active queries by the system [10, 33, 34]. While such intensive supervision is warranted for some applications, particularly in graphics [4, 20, 25, 35], it may be overkill for others. We focus on the foreground propagation problem, which assumes supervision in the form of a single labeled frame. Regardless, improvements due to our supervoxel idea could also benefit the interactive methods, some of which start with a similar MRF graph structure [10,20,25,33] (but lack the proposed higher order potentials).

Weakly supervised video cosegmentation An alternative way to supervise video segmentation is to provide the algorithm with a *batch* of videos, all known to contain the same object or object category of interest as foreground. Methods for this "weakly supervised" setting attempt to learn an object model from ambiguously labeled exemplars [15, 23, 28, 30]. This is very different from the propagation problem we tackle; our method gets only one video at a time and cannot benefit from cross-video appearance sharing.

Semi-supervised foreground propagation Most relevant to our work are methods that accept a frame labeled manually with the foreground region and propagate it to the remaining clip [3, 10, 27, 31, 33]. While differing in their optimization strategies, most prior methods use the core MRF structure described above, with i) unary potentials determined by the labeled foreground's appearance/motion and ii) pairwise potentials determined by nodes' temporal or spatial adjacency. Pixel-based graphs can maintain very fine boundaries, but suffer from high computational cost and noisy temporal links due to unreliable flow [3,33]. Superpixel-based graphs form nodes by segmenting each frame independently [10, 27, 31]. Compared to their pixel counterparts, they are much more efficient, less prone to optical flow drift, and can estimate neighbors' similarities more robustly due to their greater spatial extent. Nonetheless, their use of per-frame segments and frame-to-frame flow links limits them to short range interactions. In contrast, our key idea is to impose a supervoxel potential to encourage consistent labels across broad spatio-temporal regions.

Higher order potentials for segmentation Our approach is inspired by higher order potentials (HOP) for multi-class static image segmentation [16]. There, multiple over-segmentations are used to define large spatial cliques in the Robust P^n model, capturing a label consistency preference for each image segment's component pixels. We extend this idea to handle video foreground propagation with supervoxel label consistency.

Two existing unsupervised methods also incorporate the Robust P^n model to improve video segmentation, but with important differences from our approach. In [8], the spatial cliques of [16] are adopted for each frame, and 3-frame temporal cliques are formed via optical flow. The empirical impact is shown for the former but not the latter, making its benefit unclear. In [32], the Robust P^n model is used to prefer consistent labels in temporally adjacent superpixels within 5-frame subsequences. Both prior methods [8, 32] rely on traditional adjacency criteria among spatial superpixel nodes to define HOP cliques, and they restrict temporal connections to a short manually fixed window (3 or 5 frames). In contrast, we propose *supervoxel* cliques and HOPs that span space-time regions of variable length. The proposed cliques often span broader areas in space-time—at times the entire video length—making them better equipped to capture an object's long term evolution in appearance and shape. Ours is the first video segmentation approach (unsupervised or semi-supervised) to incorporate label consistency over supervoxels.

3 Approach

The input to our approach is a video clip and one labeled frame in which an annotator has outlined the foreground object of interest. The output is a space-time segmentation that propagates the foreground (fg) or background (bg) label to every pixel in every frame. While the foreground object must be present in the labeled frame, it may leave and re-enter the scene at other times.

3.1 Motivation and approach overview

Our main objective is to define a space-time graph and energy function that respect the "big picture" of how objects move and evolve throughout the clip. Key to our idea is the use of *supervoxels.* Supervoxels are space-time regions computed with a bottom-up unsupervised video segmentation algorithm [14, 36, 38]. They typically oversegment—meaning that objects may be parcelled into many supervoxels—but the object boundaries remain visible among the supervoxel boundaries. They vary in shape and size, and will typically be larger and longer for content more uniform in its color or motion. Though a given object part's supervoxel is unlikely to remain stable through the entire length of a video, thanks to temporal continuity, it will often persist for a series of frames. For example, in Figure 1, we see a number of larger supervoxels remain steady in early frames, then some split/merge as the dog's pose changes, then a revised set again stabilizes for the latter chunk of frames. As we will see below, our approach exploits the partial stability of the supervoxels but also acknowledges their noisy imperfections.

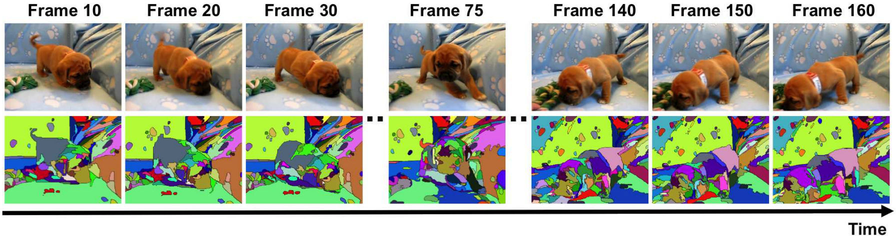


Fig. 1: Example supervoxels, using [14]. Unique colors are unique supervoxels, and repeated colors in adjacent frames refer to the same supervoxel. Best viewed in color.

While a number of supervoxel algorithms could be used, we choose the method of Grundmann et al. [14] due to its efficiency and object boundary-preserving properties [36]. The method uses appearance and motion cues to produce a hierarchy of supervoxels, and as such it can detect long-term coherence. To be concrete, whereas flat pixel-level approaches typically return regions on the order of ∼5 frames, the Grundmann approach yields voxels lasting up to 400 frames for some videos. We take all supervoxels at the 15-th level of the tree, which based on preliminary visual inspection was found to be a good middle ground between very fine and coarse voxels.[1]

How should supervoxels be leveraged for propagation? To motivate our solution, first consider an analog in the *static* image segmentation domain, which is currently much more mature than video segmentation. It is now standard in static segmentation to construct MRF/CRF models using superpixel nodes rather than pixel nodes, e.g., [29]. Superpixels [11, 26] are local oversegmented spatial regions with coherent color or texture. MRF segmentations on a superpixel graph are not only faster to compute, but they also enable broader spatial connections and richer unary potentials.

A naive generalization to video would build a graph with supervoxels as nodes, connecting adjacent supervoxels in space and time. The problem is the irregular shape of supervoxels—and their widely varying temporal extents—lead to brittle graphs. As we will see in the results, the pairwise potentials in such an approach lead to frequent bleeding across object boundaries.

Instead, we propose to leverage supervoxels in two ways. First, for each supervoxel, we project it into each of its child frames to obtain spatial superpixel nodes. These nodes have sufficient spatial extent to compute rich visual features. Plus, compared to standard superpixel nodes computed independently per frame [3, 8, 10, 12, 25, 27, 31], they benefit from the broader perspective provided by the hierarchical space-time segment that generates the supervoxels. For example, optical flow similarity of voxels on the dog's textured collar may preserve it as one node, whereas per-frame segments may break it into many. Secondly, we leverage supervoxels as a higher-order potential. Augmenting the usual unary and pairwise terms, we enforce a soft label consistency constraint among nodes originating from the same supervoxel. Again, this provides broader context to the propagation engine.

[1] This choice could possibly be eliminated by incorporating a "flattening" stage [37].

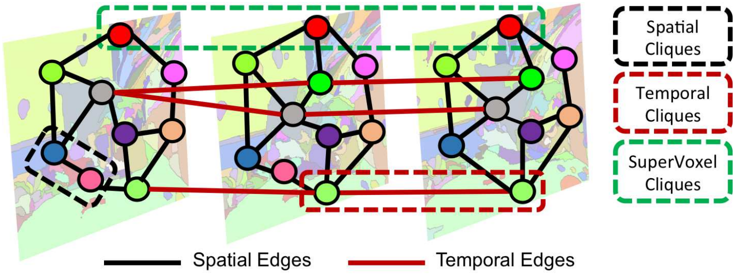


Fig. 2: Proposed spatio-temporal graph. Nodes are superpixels (projected from supervoxels) in every frame. Spatial edges exist if the superpixels have boundary overlap (black); temporal edges are computed using optical flow (red). Higher order cliques are defined by supervoxel membership (dotted green). For legibility, only a small subset of nodes and connections are depicted. Best viewed in color.

In the following, we describe the three main stages of our approach: 1) we construct a spatio-temporal graph from the video sequence using optical flow and supervoxel segmentation (Sec. 3.2); 2) we define a Markov Random Field over this graph with suitable unary potentials, pairwise potentials, and higher order potentials (Sec 3.3); and 3) we minimize the energy of this MRF by iteratively updating the likelihood functions using label estimates (Sec 3.4).

3.2 Space-time MRF graph structure

We first formally define the proposed spatio-temporal Markov Random Field (MRF) graph structure G consisting of nodes $\mathcal{X}$ and edges $\mathcal{E}$. Let $\mathcal{X} = \{X_t\}_{t=1}^{T}$ be the set of superpixels[2] over the entire video volume, where T refers to the number of frames in the video. X_t is a subset of $\mathcal{X}$ and contains superpixels belonging only to the t-th frame. Therefore each X_t is a collection of superpixel nodes $\{x_t^i\}_{i=1}^{K_t}$, where K_t is the number of superpixels in the t-th frame.

We associate a random variable $y_t^i \in \{+1, -1\}$ with every node to represent the label it may take, which can be either object (+1) or background (-1). Our goal is to obtain a labeling $\mathcal{Y} = \{Y_t\}_{t=1}^{T}$ over the entire video. Here, $Y_t = \{y_t^i\}_{i=1}^{K_t}$ represents the labels of superpixels belonging only to the t-th frame. Below, (t, i) indexes a superpixel node at position i and time t.

We define an edge set $\mathcal{E} = \{\mathcal{E}_s, \mathcal{E}_t\}$ for the video. $\mathcal{E}_s$ is the set of spatial edges between superpixel nodes. A spatial edge exists between a pair of superpixel nodes (x_t^i, x_t^j) in a given frame if their boundaries overlap (black lines in Figure 2). $\mathcal{E}_t$ is the set of temporal edges. A temporal edge exists between a pair of superpixels (x_t^i, x_{t+1}^j) in adjacent frames if any pixel from x_t^i tracks into x_{t+1}^j using optical flow (red lines in Figure 2). We use the algorithm of [6] to compute dense flow between consecutive frames. Let $[(t, i), (t', j)]$ index an edge between two nodes. For spatial edges, $t' = t$; for temporal edges, $t' = t + 1$.

Finally we use $\mathcal{S}$ to denote the set of supervoxels. Each element $v \in \mathcal{S}$ represents a higher order clique (one is shown with a green dashed box in Fig. 2)

[2] Throughout, we use “superpixel” to refer to a supervoxel projection into the frame.

over all the superpixel nodes which are a part of that supervoxel. Let y_v denote the set of labels assigned to the superpixel nodes belonging to the supervoxel v.

For each superpixel node x_t^i, we compute two image features using all its pixels: 1) an RGB color histogram with 33 bins (11 bins per channel), and 2) a histogram of optical flow, which bins the flow orientations into 9 uniform bins. We concatenate the two descriptors and compute the visual dissimilarity between two superpixels $\mathcal{D}(x_t^i, x_{t'}^j)$ as the Euclidean distance in this feature space.

3.3 Energy function with supervoxel label consistency

Having defined the graph structure, we can now explain the proposed segmentation pipeline. We define an energy function over $G = (\mathcal{X}, \mathcal{E})$ that enforces long range temporal coherence through higher order potentials derived from supervoxels $\mathcal{S}$:

$$E(\mathcal{Y}) = \underbrace{\sum_{(t,i)\in\mathcal{X}} \Phi_t^i(y_t^i)}_{Unary\ potential} + \underbrace{\sum_{\substack{[(t,i),(t',j)]\in\mathcal{E}\\ t'\in\{t,t+1\}}} \Phi_{t,t'}^{i,j}\left(y_t^i, y_{t'}^j\right)}_{Pairwise\ potential} + \underbrace{\sum_{v\in\mathcal{S}} \Phi_v\left(y_v\right)}_{Higher\ order\ potential} \quad . \tag{1}$$

The goal is to obtain the video's optimal object segmentation by minimizing Eqn. 1: $\mathcal{Y}^* = \operatorname{argmin}_{\mathcal{Y}} E(\mathcal{Y})$. The unary potential accounts for the cost of assigning each node the object or background label, as determined by appearance models and spatial priors learned from the labeled frame. The pairwise potential promotes smooth segmentations by penalizing neighboring nodes taking different labels. The higher order potential, key to our approach, ensures long term consistency in the segmentation. It can offset the errors introduced by weak or incorrect temporal connections in the adjacent frames.

Next we give the details for each of the potential functions.

Unary potential: The unary potential in Eqn. 1 has two components, an appearance model and a spatial prior:

$$\Phi_t^i(y_t^i) = \underbrace{\lambda_{app}\, A_t^i(y_t^i)}_{Appearance\ prior} + \underbrace{\lambda_{loc}\, L_t^i(y_t^i)}_{Spatial\ prior}, \tag{2}$$

where λ_{app} and λ_{loc} are scalar weights reflecting the two components' influence.

To obtain the appearance prior $A_t^i(y_t^i)$, we use the human-labeled frame to learn Gaussian mixture models (GMM) to distinguish object vs. background. Specifically, all the pixels inside and outside the supplied object mask are used to construct the foreground G_{+1} and background G_{-1} GMM distributions, respectively, based on RGB values. To compute the likelihood that a superpixel x_t^i is object or background, we use the mean likelihood over all pixels within the superpixel:

$$A_t^i(y_t^i) = -\log \frac{1}{|x_t^i|} \sum_{p\in x_t^i} P(F_p|G_{y_t^i}), \tag{3}$$

where F_p is the RGB color value for pixel p and $|x_t^i|$ is the pixel count within the superpixel node x_t^i.

The spatial prior $L_t^i(y_t^i)$ penalizes label assignments that deviate from an approximate expected spatial location for the object:

$$L_t^i(y_t^i) = -\log P(y_t^i|(t,i)), \tag{4}$$

where (t,i) denotes the location of a superpixel node. To compute this prior, we start with the human-labeled object mask in the first frame and propagate that region to subsequent frames using both optical flow and supervoxels.[3] In particular, we define:

$$P(y_{t+1}^k|(t+1,k)) = \sum_{(i,t)\in\mathcal{B}_k} \psi\left(x_{t+1}^k, x_t^i\right)\ \delta\left(P(y_t^i|(t,i)) > \tau\right), \tag{5}$$

where $\mathcal{B}_k$ is the set of superpixel nodes tracked backwards from x_{t+1}^k using optical flow, and δ denotes the delta function. The δ term ensures that we transfer only from the most confident superpixels, as determined in the prior frame of propagation. In particular, we ignore the contribution of any x_t^i with confidence lower than $\tau = 0.5$.

The term $\psi(x_{t+1}^k, x_t^i)$ in Eqn. 5 estimates the likelihood of a successful label transfer from frame t to frame $t+1$ at the site x^k. If, via the flow, we find the transfer takes place between superpixels belonging to the same supervoxels, then we predict the transfer succeeds to the extent the corresponding superpixels overlap in pixel area, $\rho = \frac{|x_t^i|}{|x_{t+1}^k|}$. Otherwise, we further scale that overlap by the superpixels' feature distance:

$$\psi(x_{t+1}^k, x_t^i) = \begin{cases} \rho & \text{if } (x_{t+1}^k, x_t^i) \in v \text{ (same supervoxel)} \\ \rho \exp\left(-\beta_u \mathcal{D}(x_{t+1}^k, x_t^i)\right) & \text{otherwise,} \end{cases}$$

where β_u is a scaling constant for visual dissimilarity.

Pairwise potential: In order to ensure that the output segmentation is smooth in both space and time, we use standard pairwise terms for both spatial and temporal edges:

$$\Phi_{t,t'}^{i,j}(y_t^i, y_{t'}^j) = \delta(y_t^i \neq y_{t'}^j) \exp\left(-\beta_p \mathcal{D}(x_t^i, x_{t'}^j)\right), \tag{6}$$

where β_p is a scaling parameter for visual dissimilarity. The penalty for adjacent nodes having different labels is contrast-sensitive, meaning we modulate it by the visual feature distance $\mathcal{D}(x_t^i, x_{t'}^j)$ between the neighboring nodes. For temporal edges, we further weigh this potential by ρ, the pixel overlap between the two nodes computed above with optical flow. Both types of edges encourage output segmentations that are consistent between nearby frames.

[3] If a frame other than the first is chosen for labeling, we propagate from that frame out in both directions. See Sec. 4.3 for extension handling multiple labeled frames.

Higher order potential: Finally, we define the supervoxel label consistency potential, which is crucial to our method. While the temporal smoothness potential helps enforce segmentation coherence in time, it suffers from certain limitations. Temporal edges are largely based on optical flow, hence they can only connect nodes in adjacent frames. This inhibits long-term coherence in the segmentation. In addition, the edges themselves can be noisy due to errors in flow.

Therefore, we propose to use higher order potentials derived from the supervoxel structure. As discussed above, the supervoxels group spatio-temporal regions which are similar in color and flow. Using the method of [14], this grouping is a result of long-term analysis of regions, and thus can overcome some of the errors introduced from optical flow tracking. For instance, in the datasets we use below, supervoxels can be up to 400 frames long and occupy up to 70% of the frame. At the same time, the supervoxels themselves are not perfect—otherwise we'd be done! Thus, we use them to define a soft preference for label consistency among superpixel nodes within the same supervoxel.

We adopt the Robust P^n model [16] to define these potentials. It consists of a higher order potential defined over supervoxel cliques:

$$\Phi_v(y_v) = \begin{cases} N(y_v)\frac{1}{Q}\gamma_{\max}(v) & \text{if } N(y_v) \leq Q \\ \gamma_{\max}(v) & \text{otherwise,} \end{cases} \tag{7}$$

where y_v denotes the labels of all the superpixel nodes within the supervoxel $v \in \mathcal{S}$, and $N(y_v)$ is the number of nodes within the supervoxel v that do not take the dominant label. That is, $N(y_v) = \min(|y_v = -1|, |y_v = +1|)$. Following [16], Q is a truncation parameter that controls how rigidly we want to enforce the consistency within the supervoxels. Intuitively, the more confident we are the supervoxels are strictly an oversegmentation, the higher Q should be.

The penalty $\gamma_{\max}(v)$ is a function of the supervoxel's size and color diversity, reflecting that those supervoxels that are inherently less uniform should incur lesser penalty for label inconsistencies. Specifically, $\gamma_{\max}(v) = |y_v| \exp(-\beta_h \sigma_v)$, where σ_v is the total RGB variance in supervoxel v.

3.4 Energy minimization and parameters

The energy function defined in Eqn. 1 can be efficiently minimized using the α-expansion algorithm [16]. The optimal labeling corresponding to the minimum energy yields our initial fg-bg estimate. We iteratively refine that output by re-estimating the appearance model—using only the most confident samples based on the current unary potentials—then solving the energy function again. We perform three such iterations to obtain the final output.

The only three parameters that must be set are λ_{app} and λ_{loc}, the weights in the appearance potential, and the truncation parameter Q. We determined reasonable values ($\lambda_{app} = 100$, $\lambda_{loc} = 40, Q = 0.2\ |y_v|$) by visual inspection of a couple outputs, then fixed them for all videos and datasets. (This is minimal effort for a user of the system. It could also be done with cross-validation, when sufficient pixel-level ground truth is available for training.) The remaining

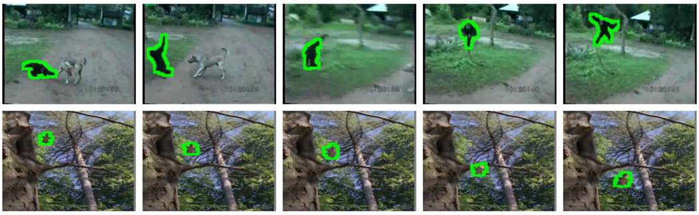


Fig. 3: Example results on SegTrack. Best viewed in color.

parameters β_u, β_p, and β_h, which scale the visual dissimilarity for the unary, pairwise, and higher order potentials, respectively, are all set automatically as the inverse of the mean of all individual distance terms.

4 Results

Datasets and metrics: We evaluate on 3 publicly available datasets: SegTrack [31], YouTube-Objects [24], and Weizmann [13]. For SegTrack and YouTube, the true object region in the first frame is supplied to all methods. We use standard evaluation metrics: average pixel label error and intersection-over-union overlap.

Methods compared: We compare to five state-of-the-art methods: four for semi-supervised foreground label propagation [9, 10, 31, 33], plus the state-of-the-art higher order potential method of [8]. Note that unsupervised multiple-hypothesis methods [17, 19, 21, 39] are not comparable in this semi-supervised single-hypothesis setting. We also test the following baselines:

- **SVX-MRF:** an MRF comprised of supervoxel nodes. The unary potentials are initialized through the labeled frame, and the smoothness terms are defined using spatio-temporal adjacency between supervoxels. It highlights the importance of the design choices in the proposed graph structure.
- **SVX-Prop:** a simple propagation scheme using supervoxels. Starting from the labeled frame, the propagation of foreground labels progresses through temporally linked (using optical flow) supervoxels. It illustrates that it's non-trivial to directly extract foreground from supervoxels.
- **PF-MRF:** the existing algorithm of [33], which uses a pixel-flow (PF) MRF for propagation. This is the only video segmentation propagation algorithm with publicly available code.[4] Note that the authors also propose a method to actively select frames for labeling, which we do not employ here.
- **Ours w/o HOP:** a simplified version of our method that lacks higher order potentials (Eqn. 7), to isolate the impact of supervoxel label consistency.

4.1 SegTrack Dataset Results

SegTrack [31] was designed to evaluate object segmentation in videos. It consists of six videos, 21-71 frames each, with various challenges like color overlap in objects, large inter-frame motion, and shape changes. Pixel-level ground truth

[4] `http://vision.cs.utexas.edu/projects/active_frame_selection/`

	Ours	PF-MRF [33]	Fathi [10]	Tsai [31]	Chockalingam [9]
birdfall	**189**	405	342	252	454
cheetah	1170	1288	**711**	1142	1217
girl	2883	8575	**1206**	1304	1755
monkeydog	**333**	1225	598	563	683
parachute	**228**	1042	251	235	502
penguin	**443**	482	1367	1705	6627

Table 1: Average pixel errors for all existing propagation methods on SegTrack.

	Ours	Ours w/o HOP	SVX-MRF	SVX-Prop
birdfall	**189**	246	299	453
cheetah	**1170**	1287	1202	1832
girl	**2883**	3286	3950	5402
monkeydog	**333**	389	737	1283
parachute	**228**	258	420	1480
penguin	**443**	497	491	541

Table 2: Average pixel errors (lower is better) for other baselines on SegTrack.

is provided, and the standard metric is the average number of mislabeled pixels over all frames, per video. The creators also provide difficulty ratings per video with respect to appearance, shape, and motion.

Table 1 shows our results, compared to all existing propagation results in the literature. We outperform the state-of-the-art in 4 of the 6 videos. Especially notable are our substantial gains on the challenging “monkeydog” and “birdfall” sequences. Figure 3 (top row) shows examples from “monkeydog” (challenging w.r.t shape & motion [31]). Our method successfully propagates the foreground, despite considerable motion and deformation. Figure 3 (bottom row) is from “birdfall” (challenging w.r.t motion & appearance [31]). Our method propagates the foreground well in spite of significant fg/bg appearance overlap.

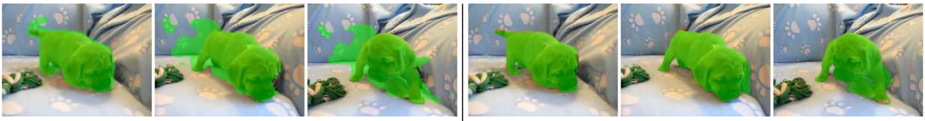
Our weaker performance on “cheetah” and “girl” is due to undersegmentation in the supervoxels, which hurts the quality of our supervoxel cliques and the projected superpixels. In particular, “cheetah” is low resolution and fg/bg appearance strongly overlap, making it more difficult for [14] (or any supervoxel algorithm) to oversegment. This suggests a hierarchical approach that considers fine to coarse supervoxels could be beneficial, which we leave as future work.

PF-MRF [33], which propagates based on flow links, suffers in several videos due to errors and drift in optical flow. This highlights the advantages of our broader scale nodes formed from supervoxels: our graph is not only more efficient (it requires 2-3 minutes per video, while PF-MRF requires 8-10 minutes), but it also is robust to flow errors. The prior superpixel graph methods [10, 31] use larger nodes, but only consider temporal links between adjacent frames. Thus, our gains confirm that long-range label consistency constraints are important for successful propagation.

Table 2 compares our method to the other baselines on SegTrack. SVX-Prop performs poorly, showing that tracking supervoxels alone is insufficient. SVX-MRF performs better but still is much worse than our method, which shows that it’s best to enforce supervoxel constraints in a soft manner. We see that the higher order potentials (HOP) help our method in all cases (compare cols 1 and 2 in Table 2). To do a deeper analysis of the impact of HOPs, we consider the

obj (#vid)	Ours	Ours w/o HOP	SVX-MRF	SVX-Prop	PF-MRF [33]
aeroplne (6)	**86.27**	79.86	77.36	51.43	84.9
bird (6)	**81.04**	78.43	70.29	55.23	76.3
boat (15)	**68.59**	60.12	52.26	48.70	62.44
car (7)	**69.36**	64.42	65.82	50.53	61.35
cat (16)	**58.89**	50.36	52.9	36.25	52.61
cow (20)	**68.56**	65.65	64.66	51.43	58.97
dog (27)	**61.78**	54.17	53.57	39.10	57.22
horse (14)	**53.96**	50.76	47.91	28.92	43.85
mbike (10)	60.87	58.31	45.23	42.23	**62.6**
train (5)	66.33	62.43	47.26	55.33	**72.32**

Table 3: Average accuracy per class on YouTube-Objects (higher is better). Numbers in parens denote the number of videos for that class.



Propagation result using PF-MRF [33]

Propagation result with our method

Fig. 4: Our method resolves dragging errors common in flow-based MRFs.

sequences rated as difficult in terms of motion and shape by [31], "monkeydog" and "birdfall". On their top 10% most difficult frames, the relative gain of HOPs is substantially higher. On "birdfall" HOPs yield a 40% gain on the most difficult frames (as opposed to 23% over all frames). On "monkeydog" the gain is 18% (compared to 13% on all frames).

4.2 YouTube-Objects Dataset Results

Next we evaluate on the YouTube-Objects [24]. We use the subset defined by [30], who provide segmentation ground truth. However, that ground truth is approximate—and even biased in our favor—since annotators marked supervoxels computed with [14], not individual pixels. Hence, we collected fine-grained pixel-level masks of the foreground object in every 10-th frame for each video using MTurk. In all, this yields 126 web videos with 10 object classes and more than 20,000 frames.[5] To our knowledge, these experiments are the first time such a large-scale evaluation is being done for the task of foreground label propagation; prior work has limited its validation to the smaller SegTrack.

Table 3 shows the results in terms of overlap accuracy. Our method outperforms all the baselines in 8 out of 10 classes, with gains up to 8 points over the best competing baseline. Note that each row corresponds to multiple videos for the named class; our method is best on average for over 100 sequences.

On YouTube, PF-MRF [33] again suffers from optical flow errors, which introduce a "dragging effect". For example, Figure 4 shows the PF-MRF pixel flow drags as the dog moves on the sofa (left), accumulating errors. In contrast, our method propagates the fg and bg more cleanly (right). The SVX-MRF baseline is on average 10 points worse than ours, and only 25 seconds faster.

Comparing the first two columns in Table 3, we see our supervoxel HOPs have the most impact on "boat", "dog", and "cat" videos. They tend to have

[5] Available at `http://vision.cs.utexas.edu/projects/videoseg/`

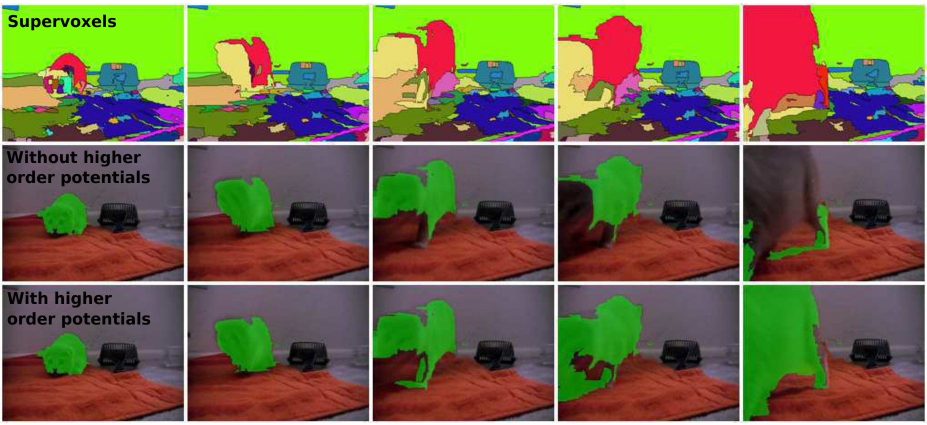
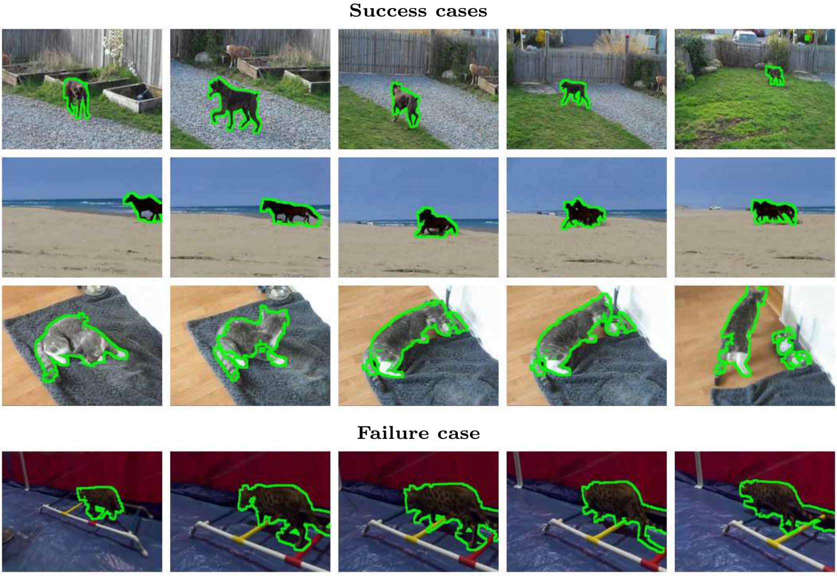


Fig. 5: Label propagation with and without HOPs (frames 31, 39, 42, 43, 51).


Success cases

Failure case

Fig. 6: Qualitative results highlighting our performance under fast motion, shape changes, and complex appearance. The first image in each row shows the human-labeled first frame of the video. See text for details.

substantial camera and object motion. Thus, often, the temporal links based on optical flow are unreliable. In contrast, the supervoxels, which depend on not only motion but also object appearance, are more robust. For example, Figure 5 shows a challenging case where the cat suddenly jumps forward. Without the HOP, optical flow connections alone are insufficient to track the object (middle row). However, the supervoxels are still persistent (top row), and so the HOP propagates the object properly (bottom row).

Figure 6 shows more qualitative results. Our method performs well even in the cases where there is significant object or camera motions. The cat (third row)

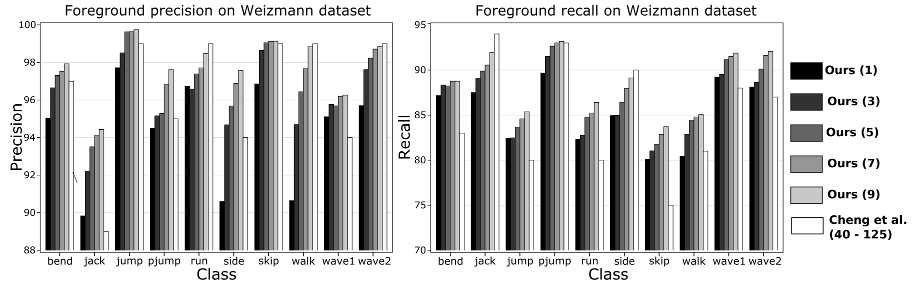


Fig. 7: Foreground precision (left) and recall (right) on Weizmann. Legend shows number of labeled frames used per result (1 to 9 for our method, 40-125 for [8]).

also shows our robustness to fg-bg appearance overlap. In the failure case (last row), we intially track the cat well, but later incorrectly merge the foreground and ladder due to supervoxel undersegmentations.

4.3 Weizmann Dataset Results

Lastly, we use the Weizmann dataset [13] to compare to [8], which uses higher order spatial cliques and short temporal cliques found with flow (see Sec. 2). The dataset consists of 90 videos, from 10 activities with 9 actors each.

Figure 7 shows the results in terms of foreground precision and recall, following [8]. Whereas we output a single fg-bg estimate (2 segments), the method of [8] outputs an oversegmentation with about 25 segments per video. Thus, the authors use the ground truth on each frame to map their outputs to fg and bg labels, based on majority overlap; this is equivalent to obtaining on the order of 25 manual clicks per frame to label the output. In contrast, our propagation method uses just 1 labeled frame to generate a complete fg-bg segmentation. Therefore, we show our results for increasing numbers of labeled frames, spread uniformly through the sequence. This requires a multi-frame extension of our method—namely, we take the appearance model G_{y_t} from the labeled frame nearest to t, and re-initialize the spatial prior $L_t^i(y_t^i)$ at every labeled frame.

With just 5 labeled frames (compared to the 40-125 labeled frames used in [8]), our results are better in nearly all cases. Even with a single labeled frame, our performance is competitive. This result gives strong support for our formulation of a long-range HOP via supervoxels. Essentially, the method of [8] achieves a good oversegmentation, whereas our method achieves accurate object tubes with long range persistence.

5 Conclusions

We introduced a new semi-supervised approach to propagate object regions in video. Due to its higher order supervoxel potential, it outperforms the state-of-the-art on over 200 sequences from 3 distinct datasets. In future work, we plan to extend the idea to accommodate multiple and/or hierarchical supervoxel inputs, and to explore shape descriptors to augment the foreground models.

Acknowledgements This research is supported by ONR award N00014-12-1-0068.

References

1. Ahuja, N., Todorovic, S.: Connected segmentation tree: a joint representation of region layout and hierarchy. In: CVPR (2008)
2. Ali, K., Hasler, D., Fleuret, F.: Flowboost: Appearance learning from sparsely annotated video. In: CVPR (2011)
3. Badrinarayanan, V., Galasso, F., Cipolla, R.: Label propagation in video sequences. In: CVPR (2010)
4. Bai, X., Wang, J., Simons, D., Sapiro, G.: Video snapcut: Robust video object cutout using localized classifiers. In: SIGGRAPH (2009)
5. Brendel, W., Todorovic, S.: Video object segmentation by tracking regions. In: ICCV (2009)
6. Brox, T., Malik, J.: Large displacement optical flow: descriptor matching in variational motion estimation. PAMI 33(3), 500–513 (2011)
7. Brox, T., Malik, J.: Object Segmentation by Long Term Analysis of Point Trajectories. In: ECCV (2010)
8. Cheng, H.T., Ahuja, N.: Exploiting nonlocal spatiotemporal structure for video segmentation. In: CVPR (2012)
9. Chockalingam, P., Pradeep, S.N., Birchfield, S.: Adaptive fragments-based tracking of non-rigid objects using level sets. In: ICCV (2009)
10. Fathi, A., Balcan, M., Ren, X., Rehg, J.: Combining self training and active learning for video segmentation. In: BMVC (2011)
11. Felzenszwalb, P., Huttenlocher, D.: Efficient graph-based image segmentation. IJCV 59(2) (2004)
12. Galasso, F., Cipolla, R., Schiele, B.: Video segmentation with superpixels. In: ACCV (2012)
13. Gorelick, L., Blank, M., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. PAMI 29(12), 2247–2253 (2007)
14. Grundmann, M., Kwatra, V., Han, M., Essa, I.: Efficient hierarchical graph based video segmentation. In: CVPR (2010)
15. Hartmann, G., Grundmann, M., Hoffman, J., Tsai, D., Kwatra, V., Madani, O., Vijayanarasimhan, S., Essa, I., Rehg, J., Sukthankar, R.: Weakly supervised learning of object segmentations from web-scale video. In: ECCV Workshop on Vision in Web-Scale Media (2012)
16. Kohli, P., Ladicky, L., Torr, P.H.S.: Robust higher order potentials for enforcing label consistency. In: CVPR (2008)
17. Lee, Y.J., Kim, J., Grauman, K.: Key-segments for video object segmentation. In: ICCV (2011)
18. Lezama, J., Alahari, K., Sivic, J., Laptev, I.: Track to the future: Spatio-temporal video segmentation with long-range motion cues. In: CVPR (2011)
19. Li, F., Kim, T., Humayun, A., Tsai, D., Rehg, J.M.: Video Segmentation by Tracking Many Figure-Ground Segments. In: ICCV (2013)
20. Li, Y., Sun, J., Shum, H.Y.: Video object cut and paste. ACM Trans. Graph. 24(3), 595–600 (2005)
21. Ma, T., Latecki, L.: Maximum weight cliques with mutex constraints for video object segmentation. In: CVPR (2012)
22. Papazoglou, A., Ferrari, V.: Fast object segmentation in unconstrained video. In: ICCV (2013)
23. Prest, A., Leistner, C., Civera, J., Schmid, C., Ferrari, V.: Learning object class detectors from weakly annotated video. In: CVPR (2012)

24. Prest, A., Leistner, C., Civera, J., Schmid, C., Ferrari, V.: Learning object class detectors from weakly annotated video. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 3282–3289. Ieee (Jun 2012), http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6248065
25. Price, B.L., Morse, B.S., Cohen, S.: Livecut: Learning-based interactive video segmentation by evaluation of multiple propagated cues. In: ICCV (2009)
26. Ren, X., Malik, J.: Learning a classification model for segmentation. In: ICCV (2003)
27. Ren, X., Malik, J.: Tracking as repeated figure/ground segmentation. In: CVPR (2007)
28. Rubio, J.C., Serrat, J., Lopez, A.M.: Video co-segmentation. In: ACCV (2012)
29. Shotton, J., Winn, J., Rother, C., Criminisi, A.: Textonboost: joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: ECCV (2006)
30. Tang, K., Sukthankar, R., Yagnik, J., Fei-Fei, L.: Discriminative segment annotation in weakly labeled video. In: CVPR (2013)
31. Tsai, D., Flagg, M., Rehg, J.: Motion coherent tracking with multi-label mrf optimization. In: BMVC (2010)
32. Vazquez-Reina, A., Avidan, S., Pfister, H., Miller, E.: Multiple hypothesis video segmentation from superpixel flows. In: ECCV (2010)
33. Vijayanarasimhan, S., Grauman, K.: Active frame selection for label propagation in videos. In: ECCV (2012)
34. Vondrick, C., Ramanan, D.: Video annotation and tracking with active learning. In: NIPS (2011)
35. Wang, J., Bhat, P., Colburn, A., Agrawala, M., Cohen, M.F.: Interactive video cutout. ACM Trans. Graph. 24(3), 585–594 (2005)
36. Xu, C., Corso, J.: Evaluation of super-voxel methods for early video processing. In: CVPR (2012)
37. Xu, C., Whitt, S., Corso, J.: Flattening supervoxel hierarchies by the uniform entropy slice. In: ICCV (2013)
38. Xu, C., Xiong, C., Corso, J.J.: Streaming Hierarchical Video Segmentation. In: ECCV (2012)
39. Zhang, D., Javed, O., Shah, M.: Video object segmentation through spatially accurate and temporally dense extraction of primary object regions. In: CVPR (2013)