Video Segmentation with Background Motion Models

Scott Wehrwein
swehrwein@cs.cornell.edu
Cornell University
Ithaca, NY
Richard Szeliski
szeliski@fb.com
Facebook Inc.
Seattle, WA

Abstract

Many of today's most successful video segmentation methods use long-term feature trajectories as their first processing step. Such methods typically use spectral clustering to segment these trajectories, implicitly assuming that motion is translational in image space. In this paper, we explore the idea of explicitly fitting more general motion models in order to classify trajectories as foreground or background. We find that homographies are sufficient to model a wide variety of background motions found in real-world videos. Our simple approach achieves competitive performance on the DAVIS benchmark, while using techniques complementary to state-of-the-art approaches.

1 Introduction

Video segmentation is a fundamental problems in computer vision. While unsupervised segmentation of general videos remains challenging, many successful techniques use features based on motion, saliency, or a combination of both. In particular, a number of successful motion-based techniques [5, 10, 17, 18] compute long-term feature trajectories (tracks), cluster the tracks into different groups using motion similarity, and then produce a dense segmentation by interpolating segment labels for pixels that are not part of tracks. Long-term tracks provide temporal coherence, which has been a significant challenge for bottom-up approaches such as those using superpixels.

Figure 1: Example frame from our results. Left: input frame. Center: tracked features are color-coded by model fit (inliers are green, outliers are blue) with lines showing the difference to the background model's prediction. Right: dense segmentation.

However, these clustering approaches must specify pairwise affinities between tracks, which are only accurate for 2D translational motion. Affine models have been proposed [17], but at the cost of significantly increased complexity. In this work, we explicitly model background motions using feature tracks. We find that simple parametric models are sufficient to model a surprisingly common class of background motions in videos.

We have a range of choices for how to model backgrounds. At the lowest level, optical flow is flexible and detailed, but difficult to estimate and lacks predictive power. At the other extreme, 3D reconstruction techniques accurately model camera motion in a rigid scene, but are brittle in the absence of translational motions. We use simple parametric models, namely homographies, to model motion of a scene's background.

Figure 1 shows an example result from our method. After computing tracks, we fit a background model and classify each track according to its fit with the model (top row). We then use an unsupervised variant of Bilateral Video Segmentation [15] to produce a dense, edge-aware smooth segmentation that respects the track classifications.

We find that rigid backgrounds whose motion can be represented using a homography are quite common in practice. Our method outperforms other motion-based segmentation methods on the DAVIS 2016 benchmark [20]. Our method is both complementary to and competitive with the state-of-the-art, a saliency-based technique [4].

2 Related Work

The precise definition of video segmentation depends on the setting and application. Although the ideal is a complete segmentation of all objects regardless of quantity, size, and motion, many settings are more constrained. In this work, we focus on segmenting moving objects, where motion across multiple frames provides important cues about which pixels should be grouped together.

The DAVIS 2016 dataset and benchmark by Perazzi *et al*. [20] provides an overview and detailed quantitative evaluation of the large body of work related to this task, which they refer to as *video object segmentation*. They categorize video segmentation techniques as supervised (active user in the loop), semi-supervised (some ground truth, *e.g*. the first frame, is given as input), or unsupervised (no ground truth is used).

This work proposes an *unsupervised* method that makes no use of ground truth. Such methods must use other cues to determine both where to place segment boundaries and which segments belong to which category (foreground or background). The majority of unsupervised methods use some combination of *motion* and *saliency* cues. NLC, a saliency-based approach by Faktor *et al*. [4], is currently the highest-performing unsupervised approach on the DAVIS 2016 benchmark. The method begins by computing superpixels and then uses an iterative consensus scheme to group them by similarity, using saliency measures to identify a foreground cluster.

More closely related to our method are the track-based methods [5, 10, 13, 16, 17, 18], which typically compute long-term feature tracks, then use clustering to group the tracks into different objects by motion similarity. One important weakness of these methods is their reliance on pairwise track affinities, which can only represent translational motion similarity. Ochs and Brox [17] used three-way affinities to fit affine motion models, necessitating a triplet sampling step and projection from a hypergraph to a graph before clustering can be performed. Our work also begins with trajectories, but allows for more flexible homography-based motions by explicitly modeling the background motion without the need for clustering.

The idea of modeling motion with homographies has been applied to a number of previous problems. Szeliski and Shum [25] use homographies to align images for panoramic stitching; this technique was used by Chuang *et al*. [3] to create a background clean plate for video matting. Sivic *et al*. [22] used feature grouping and motion models to track and re-detect (but not segment) objects in videos. Wang *et al*. [8] use homographies to stabilize features and prune trajectories for action recognition. Closely related to our approach, Yang and Li [28] model a two-frame optical flow field using piecewise homography motions. Although this approach could be applied to videos, it was not designed with temporal stability and efficiency in mind. By using modern tracking techniques and focusing on robustly modeling simple background motion in the presence of arbitrary foreground motion, our method efficiently produces video object segmentation results competitive with the state of the art.

3 Overview

Like many other motion-based video segmentation methods, we begin by computing long-term feature tracks throughout the video. Tracks provide useful motion information that includes temporal coherence across frames. Our goal is to distinguish tracks that can be explained by the motion of a rigid background from those that belong to a moving foreground object. We first fit a motion model to the background, initializing with RANSAC and refining a nonlinear least squares objective with a per-track residual (Section 4). Using the final residuals to indicate whether each track fits the background motion, we construct a novel data term for the efficient bilateral-space video segmentation technique of [15], adapting it to the unsupervised setting (Section 5).

3.1 Modeling Assumptions

Unsupervised video segmentation techniques make modeling assumptions to help scope the problem being solved. Motion-based techniques such as ours assume that the foreground undergoes motion separate from the background. We also focus on producing a binary foreground-background segmentation, without separating multiple foreground objects.

In this work, we use homographies to model background motion, meaning that our method is designed to work well on a restricted class of scenes where this model applies. In particular, the scene's background must be rigid and satisfy *one* of the following criteria:

- the background is far from the camera, or
- the background is mostly planar, or
- the camera's 3D motion involves primarily rotation.

Although these criteria seem restrictive, videos that satisfy them are ubiquitous. For example, 42 of the 50 sequences in the DAVIS 2016 dataset—which was designed with variety in mind—satisfy one of these additional criteria; 30 of those 42 are also not labeled with the "Dynamic Background" attribute. We argue that this is a useful class of sequences to be able to segment, and that more sophisticated motion models could be developed in the future to handle more general classes of scenes.

4 Motion Model Estimation

For a sequence of input frames $I_1 \dots I_F$, our goal is to fit a set of motion models describing the background motion. Our model is defined by the homographies $H_1 \dots H_{F-1}$, where H_i explains the dominant (hopefully background) motion between I_i and I_{i+1}. Given a model, we can evaluate how well each track's motion fits the model to determine whether it is part of the background or foreground.

We begin by computing feature tracks. We used the GPU implementation of the flow-based tracker by Sundaram *et al.* [23] with a pixel spacing of 8 to produce a fairly dense set of tracks. All tracks that span less than 5 frames are eliminated to reduce noise. From these tracks, we calculate an initial homography using a standard per-frame RANSAC fit to all the observed correspondences in each pair of frames. This initialization provides a good initial guess, but does not leverage multiframe information provided by the tracks. We therefore refine the model by minimizing a robust nonlinear least squares cost ψ that considers all data but limits the penalty for outliers.

We measure each track t's fit to the model by calculating its reprojection in each frame f and taking the maximum reprojection error frames as the track's residual:

$$r_t = \max_f \|H_f p_{t,f} - p_{t,f+1}\|, \tag{1}$$

where $p_{t,f}$ is the 2D position (in homogeneous coordinates) of feature track t in frame f.

Use of the max per-frame residual is common in the track clustering literature, which faces a similar problem when defining track affinities across a sequence of frames.[1] The maximum is an effective choice here because a foreground object (*e.g.*, a walking bear's foot) may be stationary in some frames. By considering residuals per-track and using the maximum error over all frames, the bear's foot is determined to be an outlier in all frames.

To refine the model, we use a smooth truncated quadratic cost given by Zach [29]:

$$\psi(r_t) = \frac{1}{2} \min_w \left\{ w^2 \|r_t\|^2 + \frac{\tau^2}{2}(1-w^2)^2 \right\}, \tag{2}$$

where τ is a parameter that determines how large the residual can get before its cost saturates. This cost limits the penalty for outliers by using an auxiliary variable w, which appears as a weight in the first term, and as a regularizer in the second term that prevents all weights from tending to zero. The weights that minimize ψ in can be computed from the residuals as

$$w^2(r_t) = 1 - \frac{\|r_t\|^2}{\tau^2}, \tag{3}$$

and the cost as a function of only the residual is

$$\psi(r_t) = \left\{ \begin{array}{ll} \frac{1}{2}\|r_t\|^2 \left(1 - \frac{\|r\|^2}{2\tau^2}\right) & \text{if } \|r\|^2 \le \tau^2 \\ \frac{\tau^2}{4} & \text{otherwise} \end{array} \right\} \tag{4}$$

Figure 2 plots the cost (blue) and weight (red) as a function of residual for our empirically chosen value of $\tau = 4$.

[1] We also experimented with other p norms in addition to maximum but did not find it improved performance.

We optimize the model, minimizing the sum of all per-track residuals using iteratively reweighted least squares (IRLS) with a dynamic step size. At each iteration, the algorithm computes weights from the current residuals, fits a new model using weighted least squares, then recomputes the residuals given the new model. Details of the algorithm are given in the supplementary material.

This objective could also be minimized using other nonlinear least squares solvers, but we found IRLS worked well in practice.

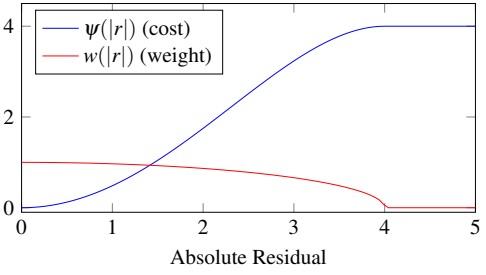

Figure 2: Cost (red) and weight (blue) as a function of absolute residual.

We now have a sequence of homographies describing the dominant motion for each frame and a single inlier weight in the range [0,1] for each track describing how well it fits the model. In the next section, we use these inlier weights as the data terms in an MRF formulation to generate a dense per-pixel segmentation over the video.

5 Dense Segmentation

The recently proposed Bilateral Video Segmentation method by Märki *et al.* [15] has shown promising results on semi-supervised video segmentation, where ground truth is given for the first frame. We adapt their method for the unsupervised setting to produce a dense, edge-aware segmentation that respects the tracks' inlier weights. Our approach closely follows Märki *et al.*, with the key difference being in the data term, which in our case is derived from the inlier weights (Equation 5). We briefly summarize the approach here for completeness.

The method begins by constructing a "lifted" bilateral representation of each pixel in the input video and each track observation's inlier confidence. The pixels and confidences are then "splatted" into a discretized 6D grid; the vertices of the grid are then segmented using graph cuts, and finally a foreground/background segment label is "sliced" out of the grid at each original input pixel location.

Each pixel is represented as a point in $\mathbb{R}^6$ where the dimensions represent position in spatiotemporal and Luv color space: (x,y,t,L,u,v). The 6D bilateral space is quantized into a grid and each lifted pixel is scaled according to the desired level of quantization, then "splatted" into the grid using some interpolation scheme. We use the adjacent interpolation scheme introduced by [15] which balances efficiency and accuracy by only interpolating onto the neighbors adjacent to the nearest neighbor.

At each grid vertex $\mathbf{v} = (v_x, v_y, v_t, v_L, v_u, v_v)$, we store a 3-vector $(\mathbf{v}_{\text{fg}}, \mathbf{v}_{\text{bg}}, \mathbf{v}_{\text{m}})$ representing foreground evidence, background evidence, and pixel mass. The value $(0,0,1)$ is splatted into the grid at each input pixel's location to distribute the pixel mass. Each track's inlier weight w from the model fit is converted into a foreground and background confidence value between 0 and 1:

$$c_{\text{fg}} = 2\max(0, 0.5 - w) \tag{5}$$

$$c_{\text{bg}} = 2\max(0, w - 0.5) \tag{6}$$

For each observation of each track, we splat $(c_{\text{fg}}, c_{\text{bg}}, 0)$ at the lifted coordinates of the observation.

Each grid vertex $\mathbf{v}$ now contains a value $(\mathbf{v}_{\text{fg}}, \mathbf{v}_{\text{bg}}, \mathbf{v}_{\text{m}})$, corresponding to the accumulated foreground evidence, background evidence, and pixel mass accumulated at that vertex. We construct the 6D grid graph $(\Gamma, \mathcal{E})$ given by the bilateral vertices for which $\mathbf{v}_{\text{m}} > 0$. We then segment the bilateral graph using graph cuts to minimize the energy function

$$E(\alpha) = \lambda_{\text{u}} \sum_{\mathbf{v}\in\Gamma} \begin{cases} \mathbf{v}_{\text{fg}} & \text{if } \alpha_{\mathbf{v}} \text{ is bg} \\ \mathbf{v}_{\text{bg}} & \text{if } \alpha_{\mathbf{v}} \text{ is fg} \end{cases} + \lambda_{\text{s}} \sum_{\mathbf{u},\mathbf{v}\in\mathcal{E}} \begin{cases} \mathbf{u}_{\text{m}} * \mathbf{v}_{\text{m}} e^{\frac{1}{2}\|W(\mathbf{u}-\mathbf{v})\|} & \text{if } \alpha_{\mathbf{v}} \neq \alpha_{\mathbf{u}} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where $\alpha_{\mathbf{v}}$ denotes the label assigned to vertex $\mathbf{v}$ and $W = \text{diag}(w_x y, w_x y, w_t, w_L, w_u v, w_u v)$ scales the effective distance between vertices in each dimension.

Intuitively, each vertex pays a cost equal to its evidence for the opposite label, and each pair of neighbors with differing labels pays a cost proportional to the product of their accumulated pixel masses; see [15] for more details and intuition.

We minimize this energy by solving a single max-flow problem [1]. We then use adjacent interpolation to reverse the splatting operation, slicing out a foreground/background value at the lifted coordinates of each input pixel. As in [15], we apply a 3x3 median filter and threshold the result to produce a final binary segmentation.

Textureless Regions Our method relies on accurate tracks, which are difficult to estimate in large textureless regions. To avoid mislabeling large textureless areas—which almost universally appear as part of backgrounds—in the absence of any data term signal, we add a weak background prior in such regions. In each frame, we add a synthetic "background observation", splatting the value $(0, w_{\text{textureless}} = 0.05, 0)$ at any points on an 8x8 pixel grid that lie more than a radius of $r_{\text{textureless}} = 32$ pixels from an actual track observation.

Parameters Our method has a small number of tunable parameters, which were set empirically. We tried using Bayesian optimization to improve certain subsets of the parameters (*e.g.*, bilateral dimension scales), but found that performance was quite stable and did not improve significantly. The supplementary material includes a table detailing all of the parameters. We plan to make our code available upon publication.

6 Results and Discussion

Sample frames from four representative sequences in the DAVIS benchmark are shown in Figure 3. For each sequence, we show two frames with the overlaid tracks color-coded by inlier weight (blue=0, green=1). For each track, we also display a vector from the track's observed location to its predicted location. In the second row, we show the final dense segmentation, shaded in red. The segmentations in these examples are quite accurate because the camera is primarily panning, or in the case of TRAIN, the background is largely planar. In some cases, our method accurately segments dynamic scene elements, such as the splashing water at the surfer's feet, although the ground truth labels this background. More results, including full videos, are included in the supplemental material.

We also found that our method performs well on many videos from the BVSD (aka VSB100) [6, 24], FBMS [18], and SegTrackv2 [14] datasets. Qualitative results on some of these sequences are given in the supplemental material. We did not compute quantitative performance scores for these datasets because their sequences and metrics are designed to evaluate different variants of video segmentation.

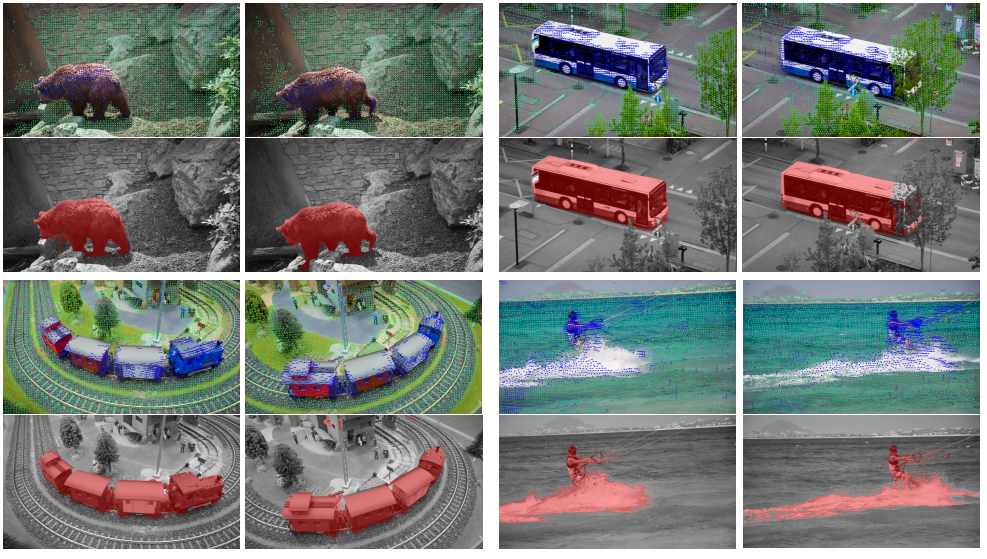

Figure 3: Sample results on BEAR, BUS, TRAIN, and KITE-SURF. For each dataset, the 20th and 40th frames are shown. The top row shows the tracks colormapped by inlier weight from blue (0) to green (1), with vectors from the track's observed location to its predicted location according to the background model. The bottom row shows the resulting segmentation.

6.1 Quantitative Performance

Table 1 (top) gives quantitative results of our method evaluated on the complete DAVIS benchmark. Although some of the sequences in the benchmark are not well-modeled by our method, we still achieve performance competitive with the current state of the art method (NLC, by Faktor *et al*. [4]). Furthermore, it is important to note that [4] is a saliency-based method, whereas ours is based on motion cues, making the two techniques complementary.

Table 1 (bottom) shows performance metrics on the subset of 30 sequences our method is designed to handle well—videos with backgrounds that can be modeled by homographies (see supplemental for a list). Our method performs competitively on average on the full dataset, while significantly outperforming other unsupervised motion-based methods (MSG [16], CVOS [26], FST [19], and TRC [5]).

Full per-sequence results are included in the supplementary material. Our method achieves the best performance on more videos (i.e., 18) than any other method (the next best method, NLC, performs best on 13 sequences).

6.2 Limitations

Figure 4 shows a few sequences that illustrate the limitations of our approach, most of which arise from scenes with backgrounds not well modeled with homographies. PARKOUR is a cinematic shot with dramatic parallax due to camera motion, and therefore a single homography is insufficient. Water ripples produce nonrigid motion, causing our method to fail on BLACKSWAN and MALLARD-WATER. In BMX-JUMP, a few frames are corrupted by track-

	Metric	NLC [4]	CVOS [26]	TRC [5]	MSG [16]	KEY [11]	SAL [27]	FST [19]	Ours
Full	J(M)	**0.641**	0.514	0.501	0.543	0.569	0.426	0.575	0.625
	J(O)	**0.731**	0.581	0.560	0.636	0.671	0.386	0.652	0.700
	F(M)	**0.593**	0.490	0.478	0.525	0.503	0.383	0.536	**0.593**
	F(O)	0.658	0.578	0.519	0.613	0.534	0.264	0.579	**0.662**
	T*	0.356	0.242	0.329	0.242	**0.189**	0.600	0.276	0.264
Subset	J(M)	0.694	0.602	0.551	0.622	0.587	0.477	0.601	**0.757**
	J(O)	0.807	0.704	0.625	0.728	0.672	0.482	0.686	**0.894**
	F(M)	0.631	0.552	0.514	0.582	0.529	0.422	0.566	**0.691**
	F(O)	0.692	0.663	0.573	0.696	0.558	0.339	0.648	**0.842**
	T*	0.306	**0.187**	0.304	0.229	0.197	0.550	0.283	0.198

Table 1: Average Jaccard (J), boundary (F), and temporal stability (T) scores unsupervised methods on the DAVIS 2016 dataset. For J and F, we give mean (M) and recall (O) (fraction of frames better than 0.5). Top: metrics for the complete DAVIS 2016. Bottom: metrics for the subset of videos with rigid backgrounds that can be modeled with homographies. *For the T metric, smaller scores are better; the metric can produce NaNs, which are ignored when computing the mean.

ing failures due to motion blur, although the method quickly recovers once the background stabilizes. In a few frames of BMX-JUMP, the rigid foreground occupies most of the frame, violating our assumption that the dominant homography models the background.

The tracking method we use [23] links pairwise flow estimates, and sometimes produces tracks that jump from background to foreground elements (and vice versa). Such tracks are considered outliers by our model, introducing spurious observations into our data term. Breaking up such tracks into more useful components could improve our results.

Finally, solutions sliced from the coarse bilateral grid often produce small, discontinuous regions with incorrect labels (for example, under the legs of BEAR). The choice of dimension scales before splatting determines both the "reach" of smoothness edges in the graph cut and the level of quantization. Efficient methods that allow independent tuning of smoothness radius and grid quantization are needed. Preliminary experiments postprocessing the results by computing features on connected components (e.g. area and distance to the largest foreground component) did not yield a significant performance improvement.

Runtime Our algorithm is quite fast: on a workstation, each 480p frame took 5.587s for tracking, 0.266s for model fitting, and 0.320s for the densification stage, totaling 6.17 seconds. More efficient trackers could improve performance significantly.

7 Future Work

In this paper, we have demonstrated that simple parametric models are effective at modeling background motion for video segmentation. These motion models can be estimated from long-term trajectories and combined with a spatiotemporal segmentation method to achieve useful video segmentation results on a large and common class of sequences. While the proposed technique is simple, it takes a different, and complementary, approach to existing approaches. Two directions for future work show promise: developing more flexible background models, and incorporating background models into other techniques.

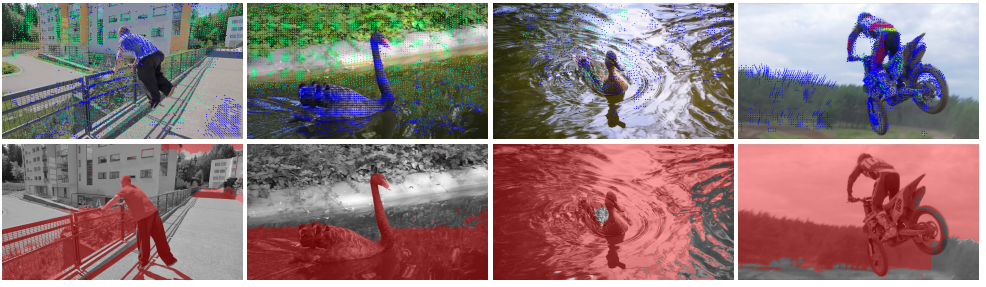

Figure 4: Frames from four sequences that illustrate our algorithm's limitations: 3D scene structure with camera translation, nonrigid backgrounds, and tracking failures.

7.1 More Flexible Motion Models

We believe that our technique could be extended to include more flexible motion models, yielding improved performance on a more complete class of camera motions and scene content. We experimented with several more sophisticated motion models, but found that currently available datasets contain relatively few videos where they apply.

To handle arbitrary rigid backgrounds, we tried two approaches. Running structure-from-motion on the videos worked when the camera translation is significant, but currently available software breaks down in the common rotation-only case. Fitting pairwise fundamental matrices instead of homographies was somewhat effective, but suffered from two problems. First, like SfM, fundamental matrices are unstable in the rotation-only case. Second, any object motion along the epipolar line will erroneously fit the model because the track's residual can only be measured by distance to the epipolar line. Future work could explore ways to automatically select the appropriate model, fit models among non-sequential pairs of frames, or introduce projective depth constraints across time.

We also experimented with fitting multiple homography models in each frame. The second and third homographies often have many inliers, encompassing a significant fraction of the outliers to the first model. However, we were unable to find simple heuristics that determine whether each additional model constitutes foreground or background. Future work could explore the use of more sophisticated multi-model fitting techniques (e.g., [9]); learning-based approaches could be applied to classify models as foreground or background.

7.2 Incorporation with Other Techniques

Our technique takes a complementary approach to many existing techniques; future work could combine background modeling with other methods that leverage clustering [10, 18], saliency [4], and semantics [2, 21], to achieve improved results. Background models could isolate foreground objects which could then be segmented using clustering to handle multiple foreground objects. Saliency-based methods could use background models to differentiate salient foreground motion from rigid background motion.

Finally, in light of recent work on semi-supervised video segmentation [2, 21], a promising direction is to initialize these methods with segmentations derived from unsupervised techniques such as ours.

8 Conclusion

In this paper, we have demonstrated a simple technique for segmenting moving foreground objects in videos by explicitly modeling the motion of the background. Although our technique is designed to work well on only 30 of the 50 videos in the DAVIS dataset, we achieve performance on par with the state of the art, while significantly outperforming other methods that rely solely on motion cues. We believe that future work exploring more flexible background models and incorporating saliency cues show potential to make further progress towards robust and accurate video segmentation.

Acknowledgements

This work was supported by Facebook and by the National Science Foundation Graduate Research Fellowship under Grant No. DGE-1650441. The authors would like to thank the anonymous reviewers and the following colleagues for helpful comments and discussions: Matt Uyttendaele, Michael Cohen, Jan-Michael Frahm, Peter Hedman, Tianfan Xue, Mrinal Mohit, Noah Snavely, and Kavita Bala.

References

[1] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(9):1124–1137, September 2004.

[2] S. Caelles, K.-K. Maninis, J. Pont-Tuset, L. Leal-Taixé, D. Cremers, and L. Van Gool. One-shot video object segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[3] Yung-Yu Chuang, Aseem Agarwala, Brian Curless, David H. Salesin, and Richard Szeliski. Video matting of complex scenes. *ACM Transactions on Graphics*, 2002.

[4] Alon Faktor and Michal Irani. Video segmentation by non-local consensus voting. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2014. doi: http://dx.doi.org/10.5244/C.28.21.

[5] K. Fragkiadaki, G. Zhang, and J. Shi. Video segmentation by tracing discontinuities in a trajectory embedding. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012.

[6] Fabio Galasso, Naveen Shankar Nagaraja, Tatiana Jimenez Cardenas, Thomas Brox, and Bernt Schiele. A unified video segmentation benchmark: Annotation, metrics and analysis. In *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, 2013.

[7] Matthias Grundmann, Vivek Kwatra, Mei Han, and Irfan Essa. Efficient hierarchical graph based video segmentation. *IEEE CVPR*, 2010.

[8] Jakob Verbeek Cordelia Schmid Heng Wang, Dan Oneata. A robust and efficient video representation for action recognition. *International Journal of Computer Vision*, 2016.

[9] Hossam Isack and Yuri Boykov. Energy-based geometric multi-model fitting. *International Journal of Computer Vision*, 97(2):123–147, Apr 2012.

[10] M. Keuper, B. Andres, and T. Brox. Motion trajectory segmentation via minimum cost multicuts. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. URL http://lmb.informatik.uni-freiburg.de//Publications/2015/KB15b.

[11] Yong Jae Lee, Jaechul Kim, and Kristen Grauman. Key-segments for video object segmentation. In *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*, 2011.

[12] J. Lezama, K. Alahari, J. Sivic, and I. Laptev. Track to the future: Spatio-temporal video segmentation with long-range motion cues. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[13] J. Lezama, K. Alahari, J. Sivic, and I. Laptev. Track to the future: Spatio-temporal video segmentation with long-range motion cues. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[14] Fuxin Li, Taeyoung Kim, Ahmad Humayun, David Tsai, and James M. Rehg. Video segmentation by tracking many figure-ground segments. In *ICCV*, 2013.

[15] Nicolas Maerki, Federico Perazzi, Oliver Wang, and Alexander Sorkine-Hornung. Bilateral space video segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[16] P. Ochs and T. Brox. Object segmentation in video: a hierarchical variational approach for turning point trajectories into dense regions. In *IEEE International Conference on Computer Vision (ICCV)*, 2011. URL http://lmb.informatik.uni-freiburg.de//Publications/2011/OB11.

[17] P. Ochs and T. Brox. Higher order motion models and spectral clustering. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. URL http://lmb.informatik.uni-freiburg.de//Publications/2012/OB12.

[18] P. Ochs, J. Malik, and T. Brox. Segmentation of moving objects by long term video analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(6): 1187 – 1200, Jun 2014. URL http://lmb.informatik.uni-freiburg.de//Publications/2014/OB14b. Preprint.

[19] A. Papazoglou and V. Ferrari. Fast object segmentation in unconstrained video. In *2013 IEEE International Conference on Computer Vision*, 2013.

[20] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Computer Vision and Pattern Recognition*, 2016.

[21] F. Perazzi, A. Khoreva, R. Benenson, B. Schiele, and A.Sorkine-Hornung. Learning video object segmentation from static images. In *Computer Vision and Pattern Recognition*, 2017.

[22] J. Sivic, F. Schaffalitzky, and A. Zisserman. Object level grouping for video shots. *International Journal of Computer Vision*, 67(2):189–210, 2006.

[23] Narayanan Sundaram, Thomas Brox, and Kurt Keutzer. Dense point trajectories by gpu-accelerated large displacement optical flow. In *Proceedings of the 11th European Conference on Computer Vision: Part I*, ECCV'10, pages 438–451, Berlin, Heidelberg, 2010. Springer-Verlag. ISBN 3-642-15548-0, 978-3-642-15548-2.

[24] P. Sundberg, T. Brox, M. Maire, P. Arbelaez, and J. Malik. Occlusion boundary detection and figure/ground assignment from optical flow. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2011.

[25] Richard Szeliski and Heung-Yeung Shum. Creating full view panoramic image mosaics and environment maps. In *Proc. SIGGRAPH*, 1997.

[26] Brian Taylor, Vasiliy Karasev, and Stefano Soatto. Causal video object segmentation from persistence of occlusions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, 2015.

[27] Wenguan Wang, Jianbing Shen, and F. Porikli. Saliency-aware geodesic video object segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[28] J. Yang and H. Li. Dense, accurate optical flow estimation with piecewise parametric model. In *CVPR*, 2015.

[29] Christopher Zach. Robust bundle adjustment revisited. In *Proceedings of ECCV*, 2014.