Motion Coherent Tracking with Multi-label MRF optimization

David Tsai
caihsiaoster@gatech.edu
Matthew Flagg
mflagg@cc.gatech.edu
James M. Rehg
rehg@cc.gatech.edu

Computational Perception Laboratory
School of Interactive Computing
Georgia Institute of Technology
Atlanta, USA

Abstract

We present a novel off-line algorithm for target segmentation and tracking in video. In our approach, video data is represented by a multi-label Markov Random Field model, and segmentation is accomplished by finding the minimum energy label assignment. We propose a novel energy formulation which incorporates both segmentation and motion estimation in a single framework. Our energy functions enforce motion coherence both within and across frames. We utilize state-of-the-art methods to efficiently optimize over a large number of discrete labels. In addition, we introduce a new ground-truth dataset, called SegTrack, for the evaluation of segmentation accuracy in video tracking. We compare our method with two recent on-line tracking algorithms and provide quantitative and qualitative performance comparisons.

1 Introduction

Recent work in visual target tracking has explored the interplay between state estimation and target segmentation [1, 6, 15]. In the case of active contour trackers and level set methods, for example, the state model of an evolving contour corresponds to a segmentation of target pixels in each frame. One key distinction, however, between tracking and segmentation is that tracking systems are designed to operate automatically once the target has been identified, while systems for video object segmentation [18] are usually interactive, and incorporate guidance from the user throughout the analysis process. A second distinction is that tracking systems are often designed for on-line, real-time use, while segmentation systems can work off-line and operate at interactive speeds.

Several recent works have demonstrated excellent results for on-line tracking in real-time [1, 6]. However, the quality of the segmentations produced by on-line trackers is in general not competitive with those produced by systems for interactive segmentation [13, 14, 18], even in cases where the user intervention is limited. One reason is that segmentation-based methods often adopt a global optimization method (e.g. graphcut) and explicitly search a large fine-grained space of potential segmentations. In contrast, for tracking-based methods the space of possible segmentations is usually defined implicitly via the parameterization of the target model, and segmentation accuracy may be traded for computational efficiency.

BMVC 2010 doi:10.5244/C.24.56

Our work is motivated by applications in biotracking, where there is a need for a general purpose tool for tracking a wide range of animals with different morphologies. In these applications, an off-line batch formulation of video analysis is acceptable, but the need for guidance by the user must be minimized in order for the tool to be useful to biologists. Furthermore, while it is highly-desirable to be able to reliably segment the limbs of a target animal, in order to analyze its behavior, it is usually not necessary to obtain the pixel-accurate segmentations that are needed in video post-production and special effects domains.

This paper describes a new method for automatic target segmentation and tracking which uses a multi-label Markov Random Field (MRF) formulation to sequentially "carve" a target of interest out of a video volume. Our goal is to obtain higher-quality segmentations than existing on-line methods, without requiring significant user interaction. The primary novelty of our approach is our treatment of the inter-related tasks of segmenting the target and estimating its motion as a single global multi-label assignment problem. Energy functions enforce the *temporal coherence* of the solution, both spatially and across time. The result is a clean problem formulation based on global energy minimization. In contrast, on-line tracking methods can employ a diverse set of techniques to achieve good performance, including adaptive cue combination [1], spatially-varying appearance models [6], and shape priors [4]. We demonstrate experimentally that our approach can yield higher-quality segmentations than these previous methods, at the cost of greater computational requirements within a batch formulation.

A second goal of this work is to support the quantitative assessment of segmentation quality in tracking, through the development of a standardized database of videos with ground-truth segmentations. There has been no systematic quantitative or comparative evaluation of segmentation quality within the visual tracking literature.[1] We identify three properties of video sequences that can hamper segmentation: color overlap between target and background appearance, interframe motion, and change in target shape. We have developed a quantitative measure for each of these properties, and have assembled an evaluation dataset, called *SegTrack*, which spans the space defined by these challenges. We provide quantitative and qualitative evaluation of our method and compare it to two recent on-line contour-based trackers [1, 6].

In summary, this paper makes three contributions:

- We introduce a novel multi-label MRF formulation of video tracking which provides high-quality target segmentations and can handle extended video sequences.
- We propose an energy function that can enforce motion coherence between spatial neighbors and across the temporal dimension.
- We present a novel database that supports systematic quantification of segmentation quality with respect to three types of challenges found in real-world video footage.

2 Related Work

There are two bodies of previous work which are related to our method. The first are techniques for video object segmentation and layer segmentation which also make use of an MRF formulation. The second are tracking methods which make use of the segmentation of the target object.

[1] In contrast, there has been extensive work on comparing the state estimation performance of standard state-based trackers. Some representative examples are [9] and the VS-PETS workshops.

Video object segmentation is usually formulated as a binary labeling problem in an MRF and solved using graphcut. In the formulation from [3], the MRF is instantiated in the temporal dimension by linking corresponding pixel sites in adjacent frames, and the solution is given by a volume graphcut.[2] This approach was improved by Li et. al. [13], by creating links between superpixels in adjacent frames. In contrast to these earlier works, we incorporate motion and segmentation constraints into a single unified multi-label formulation.

There are many alternative ways to enforce temporal coherence in video analysis, using techniques like KLT Tracking [14, 19], SIFT matching [18] and optical flow. These methods rely heavily on the quality of the motion estimates and may fail in challenging sequences. Furthermore, the flow in these works is primarily calculated between pairs of frames, and does not exploit coherence over larger time windows. Other works which address the joint computation of optical flow and segmentation [5, 21] are based on iterative estimation methods which do not provide any global guarantees on solution quality.

Recently, there have been significant advances in discrete optimization methods for large label spaces. Komodakis et. al. proposed a discrete optimization algorithm called Fast-PD [10], which provides an efficient approach to minimizing the discrete MRF energy. It has been used in image registration [7], stereo disparity estimation [10], and optical flow estimation [8]. In these latter applications it is sufficient to analyze pairs of frames, while our case requires the analysis of the entire video volume.

A large number of on-line tracking methods can produce object segmentations (representative examples are [1, 6, 17]). Since these methods are fully-automatic, they represent an interesting point of comparison. Bibby and Reid describe an impressive tracking system in [1], which demonstrates adaptation of the target model and integration of multiple cues so as to track a wide range of challenging targets. A level-set based system, described in [6], uses a combination of spatially-constrained appearance modeling and motion estimation to achieve good segmentation performance. In comparison to these works, we employ a volumetric multi-label MRF formulation. In addition, we conduct the first quantitative and qualitative comparisons between these existing methods using a standardized testing set with ground-truth.

3 Multi-Label MRF Framework

Given a video sequence and manual initialization of a target of interest in the first frame, our goal is to carve the moving target out of the video volume, yielding a target segmentation at every frame. We adopt the volumetric MRF formulation, in which the video volume is represented as a multi-label MRF with hidden nodes corresponding to the unknown labels. The resulting optimization problem is to find the joint label assignment L for all pixel sites in the video volume that minimizes

$$E(L) \quad = \quad \sum_{p \in G} V_p(l_p) + \sum_{p \in G} \sum_{q \in N(p)} V_{pq}(l_p, l_q), \tag{1}$$

where $L = \{l_p\}_{p \in G}$, $V_p(\cdot)$ are the unary potentials representing the data term, $V_{pq}(\cdot,\cdot)$ are the pairwise potentials representing the smoothness term, G represents the set of pixel sites (nodes) in the video volume, and N represents the neighborhood system of the nodes, both spatially and temporally. In this section, we define the label space and energy terms used in Equation 1.

[2] Volume graphcuts have also been employed in medical image segmentation [2, 12].

l_p	1	2	3	4	5	6	7	8	9	10
Attr(l_p)	0	0	0	0	0	1	1	1	1	1
D(l_p)	-2	-1	0	1	2	-2	-1	0	1	2

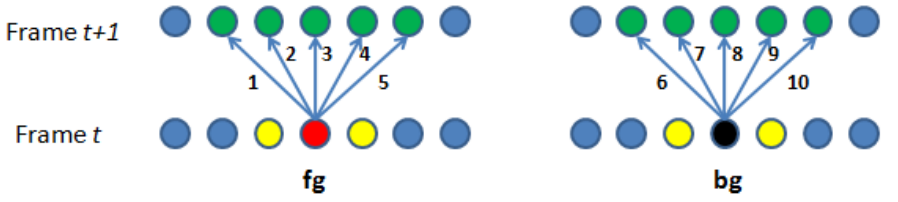

Figure 1: **Illustration of label definition**. We illustrate the label space for a center pixel in frame t. If the maximum displacement in the x direction is 2, then there are 5 possible displacements ranging from (-2,0) to (2,0). In each case, the pixel can also be labeled either foreground (red, lower left figure) or background (black, lower right figure), resulting in 10 possible labels per pixel.

3.1 Definition of Label Sets

In contrast to the standard approach to MRF-based segmentation, our label set *augments* the usual foreground/background binary attribute with a discrete representation of the flow between frames. Associated with each label is a quantized motion field $\{d^1,\cdots,d^i\}$, such that the label assignment l_p to pixel site p is associated with displacing that node by the corresponding vector d^{l_p}. If the maximum possible spatial displacement in x or y is M, and all integer displacements are allowed, then there will be a total of $(2M+1)^2$ (including zero displacement) flow possibilities for a single pixel in a 2D image. In addition, each pixel can be either foreground or background, leading to a total of $2\times(2M+1)^2$ labels per pixel. Figure. 1 illustrates these combinations for a simple 1D example. Note that we take the Cartesian product of attributes and flows (rather than their sum) because the interaction between these two components is a key element in enforcing temporal coherence between frames.

3.2 Data Term

The data term in Equation 1 is defined as follows:

$$V_p(l_p) = \underbrace{\int_\Omega w(x,p)\cdot\rho(I(x),I(x+D(l_p)))dx}_{\text{Appearance Similarity}} + \underbrace{U_p(l_p)}_{\text{Appearance Model}} \tag{2}$$

The first term in Equation 2 measures the appearance similarity across the temporal dimension. Ω represents the nodes in the local patch, $I(\cdot)$ is the intensity of the pixel, and $\rho(\cdot,\cdot)$ is the similarity measurement. Our implementation uses the Gaussian weighted Sum of Absolute Differences between the two patches centered by control points. This is very similar to the pixel-based matching costs used in stereo matching. Other more sophisticated measures such as normalized cross correlation, Rank Filter, or mutual information, could be used instead.

The second term measures pixel appearance relative to the foreground/background color

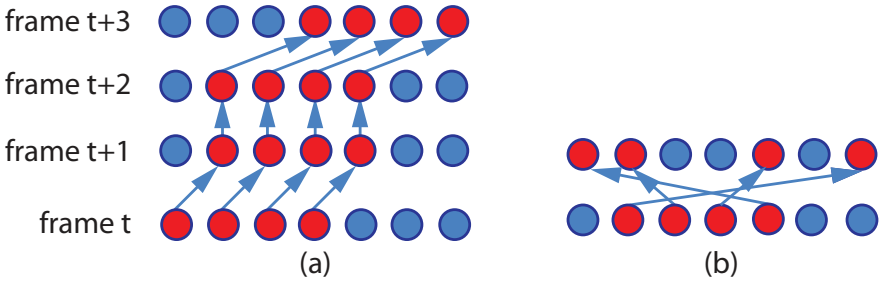

Figure 2: **Example of Motion Coherence**. (a) The movement is both spatially and temporally coherent (b) The movement is spatially and temporally incoherent.

models, and is defined as:

$$U_p(l_p) = \begin{cases} -\log \Pr^{fg}(p) & \text{Attr}(l_p) = fg \\ -\log \Pr^{bg}(p) & \text{Attr}(l_p) = bg \end{cases} \tag{3}$$

The appearance model term helps to decide whether one pixel is more likely to be foreground or background. We employ Gaussian Mixture Models for both target and background in RGB color space. These models are used to compute the pixel likelihoods in Equation 3.

3.3 Smoothness Term

The smoothness term is an important part of our formulation. It incorporates coherence in attributes over time as well as spatial and temporal motion coherence. In Equation 1, for each pixel p, we use $N(p)$ to denote its neighbors, which includes both spatially and temporally-adjacent pixel sites. An example is given in Figure 1. The yellow pixels are the spatial neighbors of the target pixel, while the green pixels are the temporal neighbors. We compute pairwise energies for every pair of neighbors. So for each pixel, there are a total of 4 spatial neighbors and $(2M+1)^2$ temporal neighbors. The basic coherence function is given in Equation 4. It is evaluated directly for spatial neighbors. For temporal neighbors, we must ensure that it is only applied to corresponding pixels. Let $\text{Corr}(l_p)$ for a site p at time t denote the corresponding site at time $t+1$ that p maps to under the displacement $D(l_p)$. The coherence term for temporal neighbors (p,q) is then given by $V_{pq}(l_p,l_q)\delta(q,\text{Corr}(l_p))$, where δ denotes the indicator function which is one when its two arguments are equal and zero otherwise.

$$V_{pq}(l_p,l_q) = \underbrace{\lambda|D(l_p)-D(l_q)|}_{\text{Motion Coherence}} + \underbrace{U_{pq}(l_p,l_q)}_{\text{Attribute Coherence}} \tag{4}$$

The first term of Equation 4 captures the property of motion coherence, and has both spatial and temporal components. In the spatial dimension, the intuition is that points which are close to one another will move coherently. In the temporal dimension, the intuition is that the object should maintain a coherent movement across frames. This is illustrated in Figure 2.

Returning to Equation 4, its second term captures the property of attribute coherence as

$$U_{pq}(l_p,l_q) = \begin{cases} E_c & \text{Attr}(l_p) \neq \text{Attr}(l_q) \\ 0 & \text{Attr}(l_p) = \text{Attr}(l_q) \end{cases} \tag{5}$$

This term models the interaction between segmentation and estimated motion, and is the key benefit of the joint label space illustrated in Figure. 1. It penalizes labellings in which

spatial and temporal neighbors receive different segmentations (i.e. pixel attributes). In the spatial domain, it enforces the constraint that adjacent pixels have the same attributes. This is identical to the spatial smoothness constraint used in the standard binary label MRF formulation. In the temporal domain, it enforces the constraint that the two pixels connected by a flow vector (i.e. temporal neighbors) should have the same attribute label.

3.4 Optimization and Propagation

In order to optimize the energy function in Equation 1, we adopt the Fast-PD method of Komodakis et. al. [10, 11]. This discrete optimization technique takes advantage of the primal-dual principle, which can be stated as a relationship between two problem formulations:

$$\begin{aligned} \text{Primal: } & \min \ \mathbf{c^T x} & \qquad \text{Dual: } & \max \ \mathbf{b^T y} \\ & \text{s.t. } \mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} & & \text{s.t. } \mathbf{A^T y} \leq \mathbf{c}. \end{aligned} \tag{6}$$

Let x and y be integral-primal and dual feasible solutions having a primal-dual gap less than f, which can be written:

$$\mathbf{c^T x} \leq \mathbf{f} \cdot \mathbf{b^T y}. \tag{7}$$

Then x is an f-approximation to the optimal integral solution x^*: $\mathbf{c^T x^*} \leq \mathbf{c^T x} \leq \mathbf{f} \cdot \mathbf{c^T x^*}$.

Fast-PD has demonstrated impressive performance in multi-label MRF optimization. The generated solution is guaranteed to be be an f-approximation to the true optimum, and in practice the per-instance approximation factor often drops quickly to 1 [10]. Fast-PD can provide substantial speed-ups over conventional graphcut methods such as alpha-expansion, which would be unacceptably slow for a large label space such as ours. In our experiments, we use the library described in [11].

In our implementation, we use a multi-grid sliding window approach to address the practical infeasibility of storing the entire video volume graph in memory. We perform a global optimization on a window of n frames and infer motion and segmentation variables simultaneously. Within each window, we use down-sampled control points as nodes to reduce the spatial resolution, increasing computational efficiency. We then use the standard graph cut to interpolate the down-sampled segmentation result to the original size. For each sliding window position, the first frame of the current window is overlapped with the last frame of the previous window. Hard constraints are established for the first frame using labels obtained from the previous volumetric label assignment (or from the initialization frame in the beginning.) This enforces the continuity of the solution between window evaluations.

4 SegTrack Database

In addition to developing an effective motion coherent tracker, the second goal of this work is to facilitate a deeper understanding of the trade-offs and issues involved in on-line and off-line formulations of video segmentation and tracking, via a standardized database of videos with ground-truth segmentations. There has been very little comparative work addressing the segmentation performance of tracking methods. Our starting point was to identify three properties of video sequences that pose challenges for segmentation quality: color overlap between target and background appearance, interframe motion, and change in target shape. We developed a quantitative measure for each these phenomena, described below, and we systematically assembled an evaluation dataset, called *SegTrack*, which spans the space of challenges. We also provide direct comparison between our batch tracking method and two state-of-the-art on-line contour-based algorithms [1, 6].

In order to obtain a set of sequences which adequately cover the space of challenge properties, we went through the following selection procedure. First, a set of 11 image sequences were manually identified as potentially containing the desired challenge combinations. Each sequence was manually rated as being either high or low with respect to each challenge type. The sequences were assigned to one of eight combination bins (high/low per challenge for 3 challenges). Next, the sequences were manually segmented and the challenge measures were computed for each one. Finally, using the computed measures we selected six image sequences, ranging in length from 21 to 70 frames, that maximally cover the challenge space. We can see from Table 1 that with respect to the challenge measures (color, motion, and shape), the difficulty of the sequences can be characterized as: *parachute*: low-low-low, *girl*: low-low-high, *monkeydog*: low-high-high, *penguin*: high-low-low, *bird*: high-high-low, and *cheetah*: high-high-high. We now describe the three challenge metrics.

Target-background color overlap: An accurate segmentation of the target, provided by the user, is commonly used to estimate a color model for the target and non-target pixels. Unfortunately, the discriminative power of such models is inversely proportional to the degree of overlap between the figure-ground color distributions. Numerous trackers and interactive segmentation systems evaluate color overlap to decide how and when to lessen the importance of color and increase reliance on other models of the target (e.g. a locally modeled shape prior as in [18]). We chose to model target and ground colors with GMMs containing 5 Gaussians. Equation 8 gives a formula for evaluating color overlap on a per-frame basis. High C_1 values correspond to large target-background overlap, which makes segmentation and tracking more difficult. The average measure per sequence is given in Table 1.

$$C_1 = \frac{\int_{X \in fg} p(X|bg)}{\int_{X \in fg} p(X|fg)} + \frac{\int_{X \in bg} p(X|fg)}{\int_{X \in bg} p(X|bg)} \tag{8}$$

Interframe target motion: Many tracking systems rely on the matching of discriminative local features to maintain temporal coherence. Large target motions result in an expanded search space for registration, which can result in poor matching performance. From ground truth segmentation, we measure interframe motion as the foreground XOR intersection area normalized by the mean object bounding box area. The per-frame average motion is reported in Table 1.

Target shape change: Shape priors constructed from target initialization, keyframes (as obtained automatically in [20]) and previously-segmented frames are often adaptively applied when other appearance models (e.g. color) are predicted to have small discriminative power. When target shape is relatively constant and motion estimation is reliable, shape priors can be used to track reliably in sequences with large figure-ground color overlap and occlusions [18]. However, when motion estimation is unreliable or shape change is drastic, this strategy can fail for obvious reasons. The SegTrack database contains such challenging scenarios. The measurement of shape change is similar to that of target motion: it is given by the foreground XOR intersection area normalized by the mean object bounding box area after compensating for translational motion estimated from centroid differences. Table 1 reports the mean shape change for each sequence.

5 Experiments

In this section, we provide experimental evidence for the benefits of our approach. First, we provide quantitative comparisons to the method described in [6], using the SegTrack Database. Second, we provide qualitative performance comparisons to [1] and [6], demonstrating our method's ability to generate more accurate segmentations in many cases. Finally,

sequence	color	motion	shape	Our score	[6] score
parachute	.038	.119	.024	235	502
girl	.205	.145	.147	1304	1755
monkeydog	.299	.243	.132	563	683
penguin	1.02	.016	.013	1705	6627
birdfall	.466	.283	.070	252	454
cheetah	.760	.273	.187	1142	1217

Table 1: **SegTrack database metrics and scores**: Challenge measures and scores for each of the six SegTrack sequences. Scores correspond to average number of error pixels per frame. Select frames from *parachute*, *girl*, *monkeydog* and *birdfall* are illustrated in Figure 4, while frames from *penguin* are displayed in Figure 3.

we assess our system's performance in tracking longer sequences. We use a single set of manually-specified parameters in all of our experiments.

Our tracker is initialized by a segmentation of the first frame into foreground and background pixels, provided by the user. This is similar to [6], where the user specifies an initial contour in the first frame.

5.1 Quantitative Comparison

In order to perform a quantitative performance comparison using SegTrack, we carefully tuned and benchmarked a state-of-the-art level set-based tracker [6], using code provided by the authors. Our system is suited to offline batch processing while the system of [6] is an on-line method. The quantitative comparison is provided in Table 1. Our per-pixel segmentation accuracy is better than that of [6] across all sequences. The large difference in the score for *penguin* was caused by tracker failure (the target contour vanished completely). For the *cheetah* case, neither of the trackers perform well, as it is the most difficult sequence according to the three challenge measures.

5.2 Qualitative Comparison

Figure 3 shows a comparison between our method and that of [1] and [6] for selected frames in 5 different test sequences. To compare our output with [1], we identified the source video clips used by the authors and initialized our tracker by labeling the first frame. Our method was able to track the target and provide more accurate segmentations. In the comparison to [6], we used the testing video clips provided by the authors.

In Figure 4, we show a greater variety of example tracker outputs. The upper four sequences are a selection of frames from our SegTrack database while the two bottom clips were long sequences from BBC's Planet Earth. Full length outputs are provided in the supplementary video and on our project website.

6 Conclusion

We have described an off-line method for target tracking through the sequential segmentation of the video volume. Our formulation uses multi-label MRF optimization with an energy function that enforces spatio-temporal coherence. We present a ground-truth dataset for target tracking, called SegTrack, which is based on a systematic assessment of the sources of difficulty in accurate segmentation. We compare our method to two recent on-line trackers, and demonstrate improved performance. Our results suggest that it is possible to obtain more accurate segmentations using an off-line approach, at the cost of increased computation. Our dataset and software are available from our project website.[3]

[3] http://cpl.cc.gatech.edu/projects/SegTrack

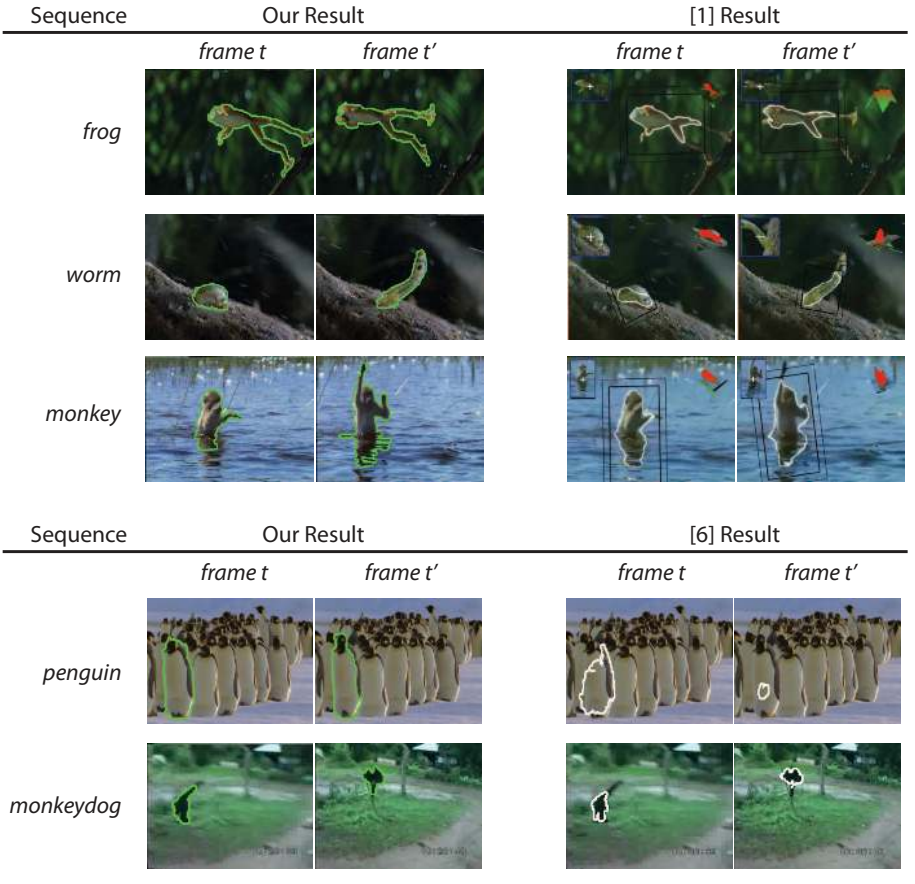

Figure 3: **Comparative results**: Tracking results are illustrated for selected frames from five sequences, comparing our method to that of [1] and [6].

7 Acknowledgements

Portions of this research were supported in part by NSF Grant 0916687 and by Google Research. We would like to thank the authors of [6] for sharing their code with us. We would also like to thank Shaunak Vaidya and Vinit Patankar for their help with the SegTrack database, and Atsushi Nakazawa for many useful discussions.

References

[1] Charles Bibby and Ian Reid. Robust real-time visual tracking using pixel-wise posteriors. In *ECCV*, 2008.

[2] Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient n-d image segmentation. *International Journal of Computer Vision (IJCV)*, 70(2):109–131, 2006.

[3] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[4] Vicent Caselles, Ron Kimmel, and Guillermo Sapiro. Geodesic active contours. *IJCV*, 22(1): 61–79, 1997.

[5] Michael M. Chang, A. Murat Tekalp, and M. Ibrahim Sezan. Simultaneous motion estimation and segmentation. *IEEE Transactions on Image Processing*, 6(9):1326–1333, 1997.

[6] P. Chockalingam, N. Pradeep, and S. T. Birchfield. Adaptive fragments-based tracking of non-rigid objects using level sets. In *ICCV*, 2009.

[7] Ben Glocker, Nikos Paragios, Nikos Komodakis, Georgios Tziritas, and Nassir Navab. Inter and intra-modal deformable registration: continuous deformations meet efficient optimal linear programming. In *IPMI*, 2007.

[8] Ben Glocker, Nikos Paragios, Nikos Komodakis, Georgios Tziritas, and Nassir Navab. Optical flow estimation with uncertainties through dynamic MRFs. In *CVPR*, 2008.

[9] Edward K. Kao, Matthew P. Daggett, and Michael B. Hurley. An information theoretic approach for tracker performance evaluation. In *ICCV*, 2009.

[10] N. Komodakis, N. Paragios, and G. Tziritas. MRF optimization via dual decomposition: Message-passing revisited. In *ICCV*, 2007.

[11] Nikos Komodakis and Georgios Tziritas. A new framework for approximate labeling via graph cuts. In *ICCV*, 2005.

[12] V. Lempitsky and Y. Boykov. Global optimization for shape fitting. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.

[13] Yin Li, Jian Sun, and Heung-Yeung Shum. Video object cut and paste. *ACM Trans. Graph.*, 24 (3):595–600, 2005.

[14] Brian L. Price, Bryan S. Morse, and Scott Cohen. Livecut: Learning-based interactive video segmentation by evaluation of multiple propagated cues. In *ICCV*, 2009.

[15] Xiaofeng Ren and Jitendra Malik. Tracking as repeated figure/ground segmentation. In *CVPR*, 2007.

[16] M.D. Rodriguez, J. Ahmed, and M. Shah. Action mach a spatio-temporal maximum average correlation height filter for action recognition. In *CVPR*, 2008.

[17] N. Vaswani, A. Tannenbaum, and A. Yezzi. Tracking deforming objects using particle filtering for geometric active contours. *IEEE Trans. PAMI*, 29(8):1470–1475, 2007.

[18] X.Bai, J.Wang, D.Simons, and G.Sapiro. Video snapcut: Robust video object cutout using localized classifiers. In *SIGGRAPH*, 2009.

[19] Jiangjian Xiao and Mubarak Shah. Motion layer extraction in the presence of occlusion using graph cuts. *PAMI*, 27(10):1644–1659, 2005.

[20] Y. Zhaozheng and R. Collins. Shape constrained figure-ground segmentation and tracking. In *CVPR*, 2009.

[21] C. Lawrence Zitnick, Nebojsa Jojic, and Sing Bing Kang. Consistent segmentation for optical flow estimation. In *ICCV*, 2005.

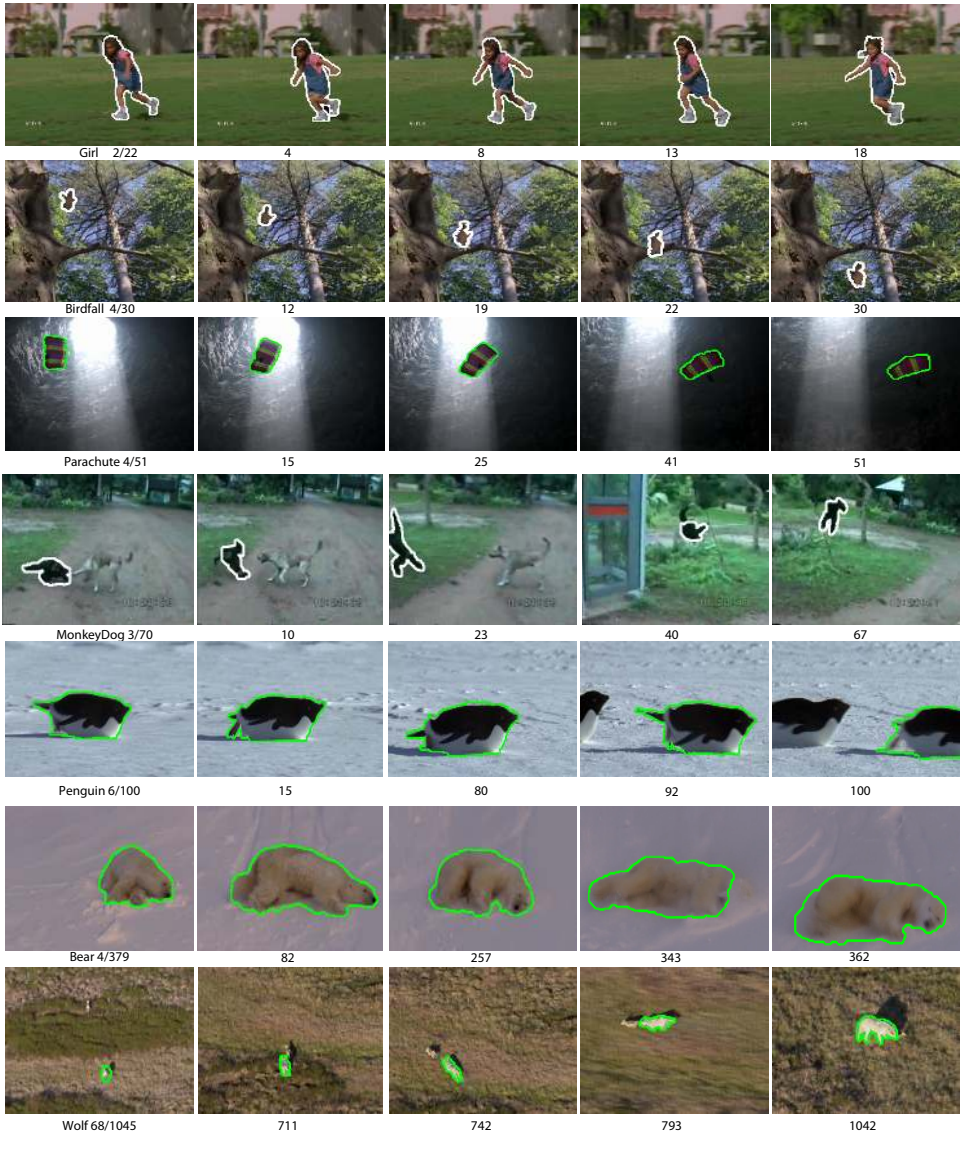

Figure 4: **Qualitative tracking results**: Top: *Girl* sequence[16] from the UCF action database, illustrating shape changes. Row 2: BirdFall sequence from SegTrack, exhibiting color overlap, large motion and small shape change, followed by Parachute, the easiest sequence in SegTrack. Row4: MonkeyDog sequence from Segtrack, showing large motion and significant shape change. Row5: One more penguin example. Rows 6 and 7: Two successfully tracked long sequences.