2007

Learning to Find Object Boundaries Using Motion Cues

Andrew Stein
Carnegie Mellon University

Derek Hoiem
Carnegie Mellon University

Martial Hebert
Carnegie Mellon University

Published In
IEEE International Conference on Computer Vision (ICCV).

Learning to Find Object Boundaries Using Motion Cues

Andrew Stein* Derek Hoiem Martial Hebert
The Robotics Institute, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213
anstein@cmu.edu

Abstract

*While great strides have been made in detecting and localizing specific objects in natural images, the bottom-up segmentation of unknown, generic objects remains a difficult challenge. We believe that occlusion can provide a strong cue for object segmentation and "pop-out", but detecting an object's occlusion boundaries using appearance alone is a difficult problem in itself. If the camera or the scene is moving, however, that motion provides an additional powerful indicator of occlusion. Thus, we use standard appearance cues (*e.g. *brightness/color gradient) in addition to motion cues that capture subtle differences in the relative surface motion (*i.e. *parallax) on either side of an occlusion boundary. We describe a learned local classifier and global inference approach which provide a framework for combining and reasoning about these appearance and motion cues to estimate which region boundaries of an initial over-segmentation correspond to object/occlusion boundaries in the scene. Through results on a dataset which contains short videos with labeled boundaries, we demonstrate the effectiveness of motion cues for this task.*

1. Introduction

There has been great progress in the last few years in *recognizing* specific objects in images, but the more general problem of *detecting* general, never-before-seen objects remains a challenge. For example, how may we determine a telephone sitting on our desk is an object separate from its surroundings, without already knowing what a telephone is? Or as Adelson and Bergen put it [2], how do we distinguish the "things" from the "stuff"?

This problem is variously known as object segmentation, *pop-out*, or figure-ground labeling. The basic problem is to extract contours that delineate scene objects or, alternatively, to extract regions corresponding to objects, based on a variety of visual cues estimated from the input image. Many cues have been proposed in relation to this problem, most notably those based on the Gestalt principles. Though largely a product of human studies, many of these cues have received significant attention in computer vision, *e.g.* [3, 17, 22, 47]. Most of these cues can be exploited in a single image, but ours is a temporal experience, and increasingly image sequences and video are becoming commonplace. Thus another potentially powerful cue, and one which is well-known in psychophysics, is that of motion – specifically relative motion discrepancies at depth discontinuities. Traditionally studied in controlled laboratory experiments [4, 8, 26, 42], such motion cues have received comparatively little attention for the purpose of object pop-out in computer vision, *e.g.* [3, 7, 29, 34].

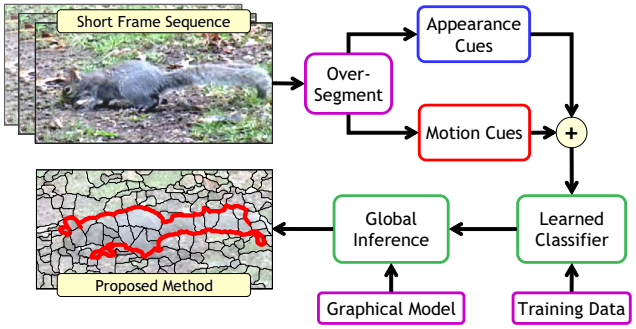

Figure 1. We develop a framework for object/occlusion boundary detection combining appearance cues, subtle, instantaneous motion cues, learned classifiers, and a global inference technique.

We propose to revisit the use of motion cues in extracting occluding contours as a step toward identifying object boundaries in a scene, with the hope of eventually enabling complete segmentation of those objects. We use subtle motion cues, such as parallax induced by a moving camera, in reasoning about these crucial boundaries which separate "things" in a scene. Here, we will begin with an over-segmentation of the image, with the assumption that the true object boundaries of interest are a subset of the fragmented boundaries formed by the regions (or *segments*) in that over-segmentation. Next we will extract a combination of appearance and motion cues [39] for the segments *and the contour fragments that separate them*. These cues will in turn generate features for a classifier trained

*Partial support provided by a National Science Foundation Graduate Fellowship, in addition to the Intelligent Robotics Development Program, funded by the Korean Ministry of Commerce, Industry, and Energy.

to distinguish fragments that are merely surface markings from those that are object/occlusion boundaries. Finally, by learning a notion of fragment connectivity and constructing a factor graph to model fragment and junction interdependencies, we will perform global inference to find the optimal labeling of the fragments jointly. Using this approach, we will demonstrate improved object boundary labeling when (a) using motion information and (b) additionally using global inference.

2. Related Work

Prior attempts to use motion cues to extract object contours (or object segmentations implying the contours) can be divided roughly into two groups: those that segment regions directly from the motion input, and those that detect contours via some local computation on the motion data. The first category includes approaches that attempt to infer segmentation or scene structure directly from reasoning about large-scale occlusions observed through dynamic object motion [6, 30] and/or the use of multiple, calibrated cameras for obtaining silhouettes [15].

Also in this category is layered motion segmentation, in which regions are segmented from an input image sequence based on the consistency of motion within each region, *e.g*. [30, 38, 45]. Most of these techniques use a parametric motion model for each layer, and employ various techniques for estimating those models and for assigning pixels to the correct layer/model. Typical models are restricted to near-planar, rigidly-moving regions. In addition, many approaches assume a known, fixed number of layers in the scene and/or do not scale well as that number increases. We argue that attempting to explain the scene in terms of a specific number of motion-consistent connected regions may not be necessary, and instead we propose to detect a large fraction of the objects' boundaries by estimating local motion cues and using them in a statistical classifier combined with a mechanism to enforce global consistency. Quite recently, a method for *binary* segmentation of video was presented which combats some of the difficulties of layered motion segmentation methods by combining clustered motion features (akin to the textons popularized by recognition research) with a boosted tree-based classifier [46].

In the second category, techniques have been developed based on the observation that occluding contours can be defined as extremal boundaries, where the viewing ray is tangent to the object's surface. This led to the development of algorithms that rely on an explicit geometric model of the motion of occluding contours [20, 36, 37, 43]. These approaches are appealing because they rely on well-defined, mathematically correct, geometric models. However, one drawback is their sensitivity to deviations of the actual data from the model. An alternative is to use an implicit model, either learned from local motion cues estimated from training data or based on some fixed model of the distribution of motion cues in the vicinity of occluding boundaries [5, 14, 28, 40]. Our work falls in this general category in that we do not attempt to precisely *model* the motion of occlusion boundaries directly. Instead we rely on the statistical modeling of (relative) local motion cues at those boundaries.

Finally, although we focus in this paper on the use of motion cues, considerable prior work exists in extracting boundaries from a single image. Two major threads emerge from this line of work. The first one is the idea of combining multiple cues into a single boundary classifier [19, 24, 11]. The second key idea, largely due to Ren [32], is to use the region boundaries of an image's (over-)segmentation as initial candidates to be labeled as occluding/non-occluding, thereby inducing a labeling of the regions as figure/ground. We build upon each of these ideas in our work. We combine many local cues into a single classifier, with the difference that we use motion cues in addition to appearance cues. We also start with an over-segmentation of the image, with the goal of filtering it to retain only those region boundaries that correspond to physical object boundaries. In addition, we use a novel model for inferring a globally consistent labeling of the boundary fragments.

Figure 2. Example input sequence (left), with initial contour fragments and junctions from over-segmentation of center frame (right).

3. Initial Segments and Fragments

We initially oversegment the image in order to generate candidate boundary fragments and associated regions of support for our motion and appearance features. Another option could be to start from edge detections and then perform an edge chaining procedure, as in [21, 38] for example. But edge chaining is inherently brittle in natural cluttered scenes, and perhaps more importantly, the over-segmentation approach offers two distinct advantages for later higher-level reasoning. First, by construction, fragments come together whenever regions meet, and thus closed contours are immediately available. This produces a natural graph structure suitable for global inference without the need to impose artificially such structure later (*e.g*. with Constrained Delaunay Triangulation [31]). Second, a direct link is established between fragments and segments. It is clear that a set of segments in an image imply a set of boundaries, but working in the opposite direction to obtain a segmentation from a set of disconnected boundary fragments is non-trivial.

We use a watershed segmentation driven by the output of the Pb detector [24] after non-local maxima suppression. We chose the watershed approach for its more regularly-shaped segments as compared to other methods [10, 13], and its speed compared to methods relying on normalized cuts [27, 33]. An example over-segmentation can be seen in Figure 2.

From the over-segmentation, we construct a contour graph by chaining together a set of fragments along the boundaries of each segment, starting and stopping at junctions with other segments (see Figure 2). Rather than operating at the level of pixels when chaining, however, we instead use the "cracks" between the pixels. These cracks naturally form a graph and offer a very simple, efficient domain on which to chain. In addition, a maximum of four fragments can meet at a junction, limiting the number of junction labeling cases we must consider when doing the global inference described later.

4. Computing Cues

Given segments and fragments from the segmentation, the next step is to compute a set of cues associated with each fragment. The cues should be chosen for their potential utility in estimating the likelihood that a fragment is on an occluding contour.

In a single image, many low-level appearance cues are available to indicate boundaries, including differing texture, color, or brightness. Motion cues are necessary because many appearance cues prominent at physical boundaries are also produced by simple surface markings. We use two main motion cues in this work. The first one is based on the relative motion of patches extracted on either side of a fragment [40]. At occlusion boundaries, there may exist an inconsistency in motion due to parallax induced by the observer's motion, dynamic objects in the scene, or both. The second motion cue is based on the observation that we can compare the segment motions not only to each other but *to the motion of the fragments which they neighbor*. Consider the common case in which the foreground side of a boundary is nearly texture-less and is moving against a cluttered background. The foreground patch motion may be difficult to estimate accurately due to the lack of texture, but we can still use the fact that the occlusion boundary is "owned" by the foreground surface and should move consistently with it [38]. More practically, it should move *inconsistently* with the background patch. By recognizing this discrepancy, we can still detect the occlusion.

In this work, we are interested in the estimation and analysis of the instantaneous motions of fragments and segments. We do not explicitly track either over long periods of time. Instead we consider only a few nearby frames in a short temporal window around the reference frame under consideration. Operating on multiple frames simultaneously also results in more stable motion estimates via more extended temporal integration of information.

4.1. Motion Cue 1: Segment Motion

The segments from Section 3 naturally specify the spatial support for estimating left- and right-side motions for each boundary fragment. Furthermore, multiple fragments bound each segment, meaning we can reuse each segment's motion estimate for reasoning about several fragments.

We assume a local translational model of motion for each segment and we use the standard Lucas-Kanade optical flow method for estimating the motion based on local spatio-temporal derivatives. We assume that our motion model is constant within the short temporal window and thus estimate a single motion for each segment using its entire spatio-temporal extent at once. Thus we have a set of equations to be solved by least squares:

$$\begin{bmatrix} tI_x & tI_y \\ \vdots & \vdots \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} I_t - I_0 \\ \vdots \end{bmatrix}, \qquad (1)$$

where $t \in \{-T_{win}, ..., 0, ..., T_{win}\}$ represents the frame number relative to I_0 (the reference frame), I_x and I_y represent spatial derivatives within the window in I_0, $I_t - I_0$ are the temporal differences between frame t and the I_0, and (u, v) are the components of the estimated translational motion.

We employ three techniques to reduce the errors in motion estimation due to pixels close to occlusion boundaries. First, pixels near the boundary of a segment are weighted such that they contribute less to the solution. Second, we use robust (*i.e*. iteratively re-weighted) least squares to solve (1). Finally, we place a weak prior on small motions, since the relative motions we seek are quite subtle.

4.2. Motion Cue 2: Fragment Motion

A second cue that will contribute to the classification of a fragment is the motion of the fragment itself. Since we do not yet know whether a fragment lies on an occlusion boundary (that is precisely what we hope to establish), it would be dangerous to employ local patches of appearance data (*i.e*. the neighboring segments) to estimate its motion by simple tracking. Instead, we need to estimate the motion of each fragment *independently* from the motion of neighboring segments. This can be accomplished by taking advantage of the fact that moving edges sweep out spatio-temporal surfaces over time [1, 7, 16, 29, 41]. Specifically, at each position along a fragment, we align the axes of a spatio-temporal cylindrical detector to the local orientation of the edge. By comparing the distributions of intensity and color within each half of the cylinder at various *temporal* orientations, we can find the speed of the moving edge in the direction normal to its spatial orientation. This approach is a temporal extension of the "compass" filter [25, 35], also used by the Pb detector [24]. It is thus quite similar to the recent approach of [41], but since spatial orientation is specified, only a cylindrical (one-DOF) speed detector is needed

rather than a spherical (two-DOF) speed-plus-orientation detector.

This approach only offers (1D) estimates of normal motions at each edge pixel due to the aperture problem. We can, however, combine the normal estimates along the fragment to get a full 2D motion estimate of the whole fragment [12, 44][1]. In our approach, we again use robust least squares to solve a linear system of equations:

$$\begin{bmatrix} n_{x,i} & n_{y,i} \\ \vdots & \vdots \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} w_i \\ \vdots \end{bmatrix} \quad (2)$$

where $n_{x,i}$ and $n_{y,i}$ are the components of the unit normal at point i on the fragment, and w_i is the corresponding speed from the spatio-temporal detector.

5. A Global Boundary Model

Given appearance information and motion estimates along fragments and within the segments they separate, our goal is to classify which fragments are occlusion boundaries. While we hope that local appearance and motion cues will provide strong evidence for this classification, it is unreasonable to hope that a *purely* local solution will suffice. To capture the structure of our problem and facilitate global reasoning and propagation of local estimates, we define a global model utilizing the graph of contour fragments and junctions implied by the reference frame's over-segmentation (Section 3). In the following, we refer to edge fragments (or *edgelets*), e_i, with labels "on" and "off" indicating occlusion and non-occlusion, respectively.

Our objective is to maximize $\Pr(e|x)$, the probability of all the edgelet labels given the cues x extracted from the data. Given the structure defined by the graph induced by the over-segmentation, this probability can be written as a product of factors. Recalling that the use of pixel cracks advantageously limits the number of fragments meeting at any given junction to either three or four, each factor corresponds either to an individual edgelet or to a junction of edgelets,

$$\Pr(e|x) \propto \prod_i \psi(e_i) \prod_k \phi_k \quad (3)$$

where $\psi(e_i)$ represents the potential function for an individual edgelet e_i and ϕ_k represents the potential function for the set of edgelets meeting at junction k, $\{e_j\}_{j=1\ldots N_k}$, where $N_k \in \{3, 4\}$.

The unary potential $\psi(e_i)$ is defined such that it only contributes to (3) if e_i is labeled "off,"

$$\psi(e_i) = \begin{cases} 1 & e_i = \mathsf{on} \\ \Pr(e_i = \mathsf{off}|x) & e_i = \mathsf{off} \end{cases} \quad (4)$$

[1]Note that perfectly straight fragments will still only permit estimation of purely normal motion, but we provide a measure of fragment curvature to the classifier to help capture this uncertainty (see Section 6).

The junction potential ϕ is evaluated for all junctions depending on the labeling of their constituent fragments, although we will only specify the three-edgelet junctions here for the sake of brevity.

Though three edgelets, each with three possible labels, would imply 27 total possible label combinations for a given junction, there are in fact only five possible configurations (up to circular permutations of the edgelets) once we rule out impossible cases. These are shown in Figure 3, with the shaded regions indicating foreground, and the darkest region being the closest one.

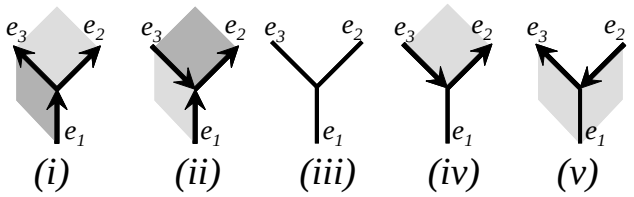

Figure 3. The five types of three-edgelet junctions. The shaded regions are the foreground regions, with darker being closer. By convention, the foreground region (shaded) is to the left of the direction indicated by the arrow for those edges that are "on". Edgelets with no arrow are "off".

For illustration, let us consider the first configuration shown in the figure, type (i). There, the factor ϕ must combine two types of information. First, e_1 and e_3 are part of the physical boundary of the occluding region on the left, and e_1 precedes e_3 when walking along this boundary. Therefore, the likelihood of e_3 being labeled "on" is conditioned on e_1 also being labeled "on" (and on the cues from the data). We denote this by $\Pr(e_3 = \mathsf{on}|e_1 = \mathsf{on}, x)$. The second term in ϕ measures how likely it is that e_2 is an occluding edge given local evidence, or $\Pr(e_2 = \mathsf{on}|x)$. The product of these two terms results in the expression for a factor of junction type (i), as indicated in Table 1.

A similar reasoning yields the expressions of ϕ for the other junctions types. Note that $\phi = 1$ for junction type (iii) in which none of the edgelets are labeled as occluding edges, indicating it does not convey any information in the original probability model. Furthermore, in types (iv) and (v), the "off" edgelets' probabilities are not factored into the potentials. For all these cases, the *unary* factors will express the likelihood of these edgelets being off, so we need not (double-)count this information in the junction factors.

Type	Potential (ϕ)
(i)	$\Pr(e_3 = \mathsf{on}\|e_1 = \mathsf{on}, x) \Pr(e_2 = \mathsf{on}\|x)$
(ii)	$\Pr(e_2 = \mathsf{on}\|e_3 = \mathsf{on}, x) \Pr(e_1 = \mathsf{on}\|x)$
(iii)	1
(iv)	$\Pr(e_2 = \mathsf{on}\|e_3 = \mathsf{on}, x)$
(v)	$\Pr(e_3 = \mathsf{on}\|e_2 = \mathsf{on}, x)$

Table 1. Junction potentials corresponding to junction types in Figure 3.

$\Pr(e|x)$ can now be computed for any assignment of labels to edgelets: for each junction we find the junction type induced by the labeling and we use the corresponding expression of ϕ to compute the contribution of the junction. The inference problem (finding the assignment of labels e that maximizes $\Pr(e|x)$) is intractable in its exact form. However, an approximation of the MAP solution can be found by combining the sum-product algorithm of Heskes *et al.* [18] with the mean field approximation suggested by Yuille [48].

6. Fragment Feature Classification

Despite the many complex junction cases described in the previous section, note that all the potentials in our model are defined in terms of just two probabilities: a unary probability that an edgelet is off and a pairwise probability that an edgelet is on given that the preceding edgelet is also on. ($\Pr(e_i = \mathsf{on}|e_j = \mathsf{on}, x)$) (Though we have not listed them here, we also define the four-edgelet junctions in terms of these same probabilities.) We will now describe a classifier used to estimate these probabilities, as functions of features (x) extracted from labeled training data.

We use the logistic regression form of Adaboost [9] for classification, where the weak learners are decision trees. Since boosted decision trees are well-suited to feature selection, we provide a variety of appearance and motion features, based on the cues described above, and allow the classifier to choose the most suitable ones. We discussed various appearance and motion cues in Section 4, but here we explicitly list the *features* derived from those cues which are provided to the classifier (let s_L and s_R denote the neighboring segments for a given edge fragment e_i):

- **Appearance Features:** average Pb-strength along e_i, length of e_i, ratio of e_i's length to the longer perimeter of s_L and s_R, difference in area between the s_L and s_R, difference in average color (in LAB space) between s_L and s_R, and χ^2-distance between color distributions estimated within s_L and s_R (*i.e.* a Pb-like operation on segments).
- **Motion Features:** absolute differences between individual u and v motion components, simple Euclidean distance between motion vectors, confidences of motion estimates (derived from the amount of gradient within a segment or the curvature of a fragment), and the Mahalanobis-like motion consistency score defined in [40]. Each of these features is computed from comparisons between e_i and each of s_L and s_R as well as comparisons between s_L and s_R themselves.

Given a set of training data, we apply the oversegmentation and the fragment-chaining approaches described in Section 3 and compute each of the above features. We then train a unary classifier directly from the individual ground truth labels and the features as listed.

For the pairwise classifier, we first extract pairs of fragments in the ground truth for which e_i follows e_j and both are labeled "on". These pairs are our positive examples while negative examples are collected from those pairs for which e_i is off but is connected via the graph to an e_j that is on. The feature vector for an example pair includes the unary features for each fragment, as listed above, augmented by (1) the relative angle between the two fragments (to capture a notion of continuity), (2) the difference between the motions of the two fragments, and (3) the motion and color differences between the two fragments' foreground-side segments. From these examples and augmented features, we learn a second, pairwise classifier. For all our experiments, we allow ten iterations of boosting with ten-node decision trees.

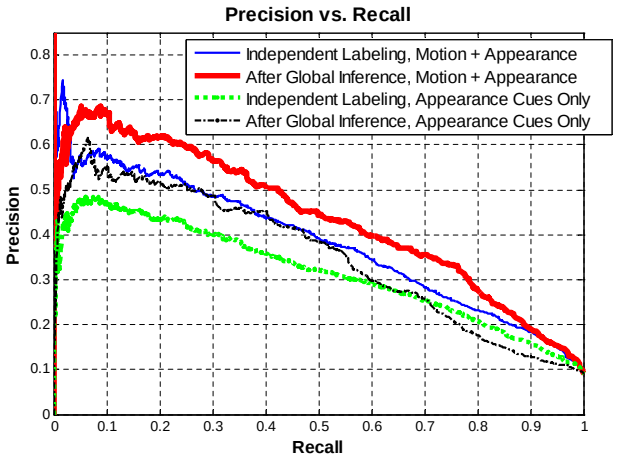

Figure 4. Precision vs. Recall for the entire dataset, showing that using motion and global inference results in the most accurate identification of those edge fragments which are occlusion/object boundaries.

7. Results

To train and evaluate our approach, we require sequences of images. Thus, existing databases for evaluating object segmentation and boundary detection, most notably the popular Berkeley Segmentation Dataset (BSDS) [23], are inappropriate for our task [39]. We have therefore created a dataset[2] consisting of 30 short video sequences (approximately 10-20 frames each) with a wide variety of content [39]: indoor and outdoor scenes, uncontrolled variable lighting, a range of scene depths, *etc*. Each exhibits very brief camera motion, instantaneous motion of objects in the scene, or a combination of the two. The dataset has not been selected specifically to suit our task and contains some very difficult cases. Our task is to detect occlusion boundaries for the middle (reference) frame of each sequence, for which we have labeled ground truth regions to indicate object/occlusion boundaries as well as the side of each boundary which is the foreground. By using this dataset, we intend to provide quantitative results in addition to the anecdotal examples often presented in motion segmentation

[2] http://www.cs.cmu.edu/∼stein/occlusion_data/

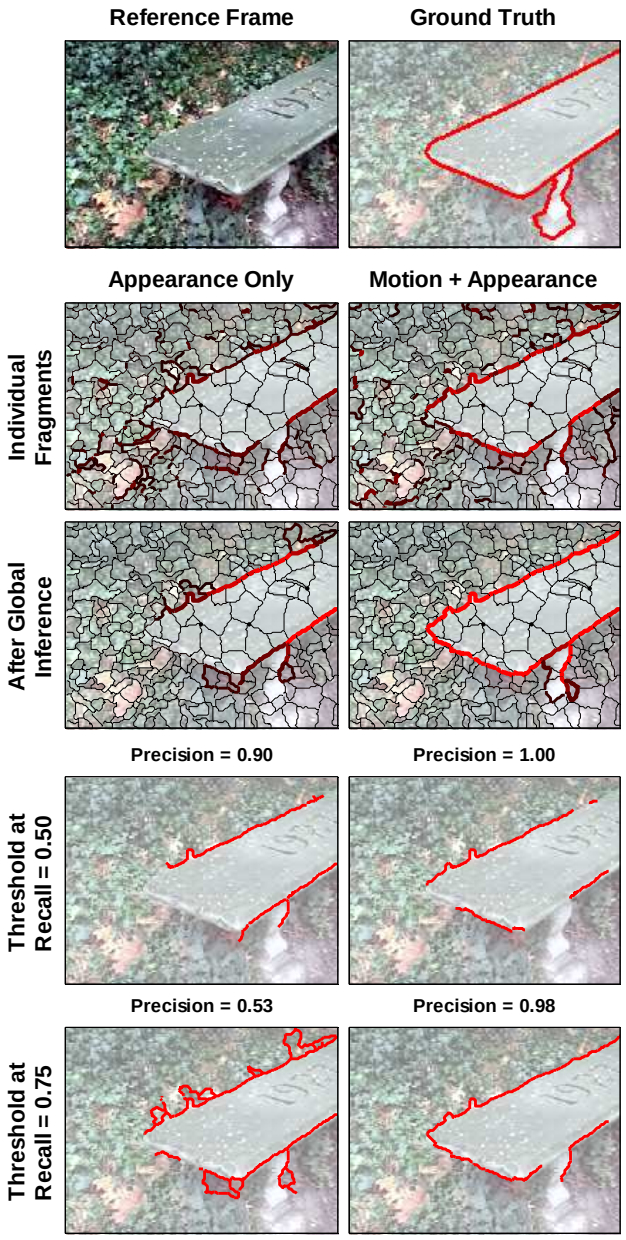

Figure 5. Example result: The appearance-only classifier's lack of confidence becomes obvious when we use use a higher-recall operating point. With the addition of motion, very high precision is maintained. (Static scene, handheld camera motion: several centimeters.) [Best viewed in color.]

work.

After over-segmentation, there are approximately 20,000 total fragments available for testing and training, on which we perform a correspondence procedure to extract ground truth labels.

We would like to verify that both global inference and motion information result in improved performance overall. We see in Figure 4 that this is indeed the case by plotting precision vs. recall for final fragment labeling of the entire dataset in aggregate. The parameter varied in creating each plot is the threshold on the likelihood ratio of each fragment being on or off. We see that using appearance cues alone results in the worst performance. In fact, note that reasoning with motion cues on individual fragments, but *without* global inference, offers equivalent or superior results than global inference used with appearance cues alone. Finally, global inference with *combined* motion and appearance information consistently yields the highest precision. Also note that the low precision at 100% recall (corresponding to the trivial solution of labeling *all* fragments as occlusion boundaries) provides some indication of the difficulty of our task and our dataset.

While aggregate statistics captured by the precision recall plots are useful for understanding quantitative performance in general, they do hide important semantic measures of quality which can only be understood by looking at individual results. In Figures 5-7, we provide a few such examples out of the 30 in our database. In each, the reference image of the sequence and the ground truth labeling are provided in the top row. In the remaining rows we compare the use of appearance only (left column) to that of appearance and motion combined (right column). The second row displays fragments overlaid on the image with brightness and line width proportional to the confidence that they are occlusion boundaries according to their *independent* classification results (*i.e*. before global inference). Thus, the brighter red and thicker a fragment is, the more the system believes it to be an occluding boundary. The next row displays the same type of result but *after* performing global inference on the initial classifications. The final two rows show these global inference results thresholded at equal recall rates for fair comparison. It is interesting to note that the motion+appearance approach sustains higher precision (*i.e*. fewer false positives) even as the recall is increased. This indicates that the motion adds significant confidence to the classifier's decision.

It is not surprising that there are several scenes in our database for which motion does not help because some objects simply *are* well-segmented by basic appearance cues, such as color, or the scenes may lack enough texture (or depth variation) to provide the necessary relative motion cues. However, it is also very rare that using motion *hurts* performance, and in those cases where appearance information alone does *not* capture the properties of occlusion boundaries well, motion cues often provide substantial improvement. Much of this improvement is due to reduced false positives, since motion information may allow the system to recognize and filter out high-contrast surface markings which confuse an appearance-only approach.

8. Conclusion

We have proposed a framework for introducing motion as a cue in detection and grouping of object/occlusion boundaries. The use of motion and occlusion for object

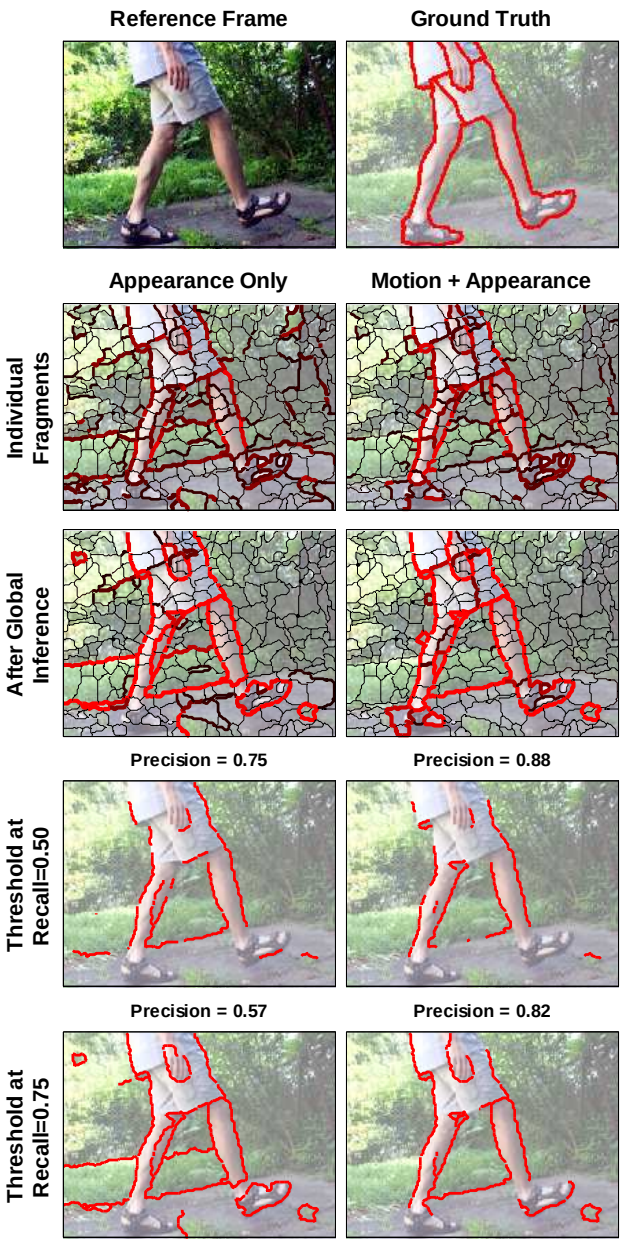

Figure 6. Example result: Using motion allows for sustained high precision, even at higher recall. (Dynamic scene, static camera.) [Best viewed in color.]

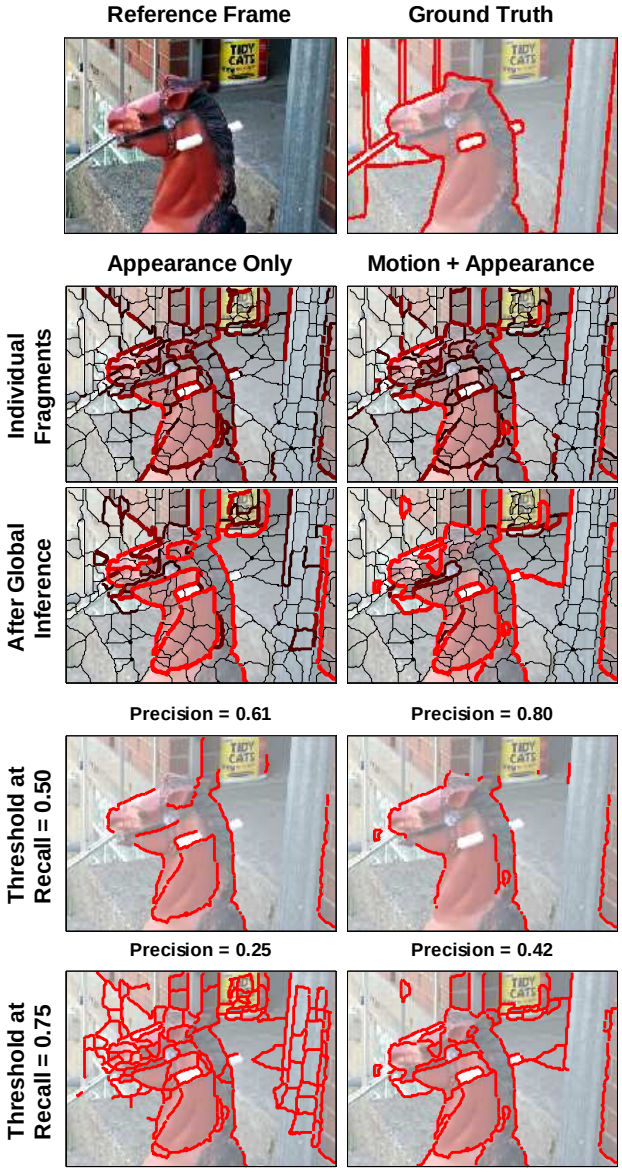

Figure 7. Example result: This more difficult example, the motion+appearance classifier still performs best. (Static scene, handheld camera motion: several centimeters.) [Best viewed in color.]

segmentation, discovery, and "pop-out", which is fundamentally important to general scene understanding, is well-established in psychophysics and perception. Given the increasing availability of and interest in temporal data for computer vision applications, the use of motion cues will offer substantial performance gains over methods based on static appearance cues only. In our experiments, we have demonstrated that motion is indeed helpful in finding these boundaries, when used in a statistical classifier applied to contour fragments generated from an over-segmentation of the image. In our continuing work, we will use these detected boundaries to improve object segmentation. And because our labeled fragments are naturally linked to an initial over-segmentation, we can use feedback between the two processes (segmentation and boundary detection) in order to improve them both in an iterative fashion. Also, additional information can be extracted as to which of the two neighboring segments at a contour fragment is on the foreground object. Initial results indicate that the framework described here can also be used for this level of labeling.

References

[1] E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of motion. *Journal of the Optical So-*

ciety of America A, 2(2):284–299, February 1985.

[2] E. H. Adelson and J. R. Bergen. *The plenoptic function and the elements of early vision*, chapter 1. The MIT Press, 1991.

[3] N. Apostoloff and A. Fitzgibbon. Learning spatiotemporal T-junctions for occlusion detection. In *CVPR*, 2005.

[4] P. Bayerl and H. Neumann. Disambiguating visual motion by form-motion interaction – a computational model. *IJCV*, 72(1):27–45, April 2007.

[5] M. J. Black and D. J. Fleet. Probabilistic detection and tracking of motion discontinuities. *IJCV*, 38(3):231–245, 2000.

[6] G. Brostow and I. Essa. Motion based decompositing of video. In *ICCV*, 1999.

[7] G. T. Chou. A model of figure-ground segregation from kinetic occlusion. In *ICCV*, 1995.

[8] C. Chubb and G. Sperling. Second-order motion perception: Space/time separable mechanisms. In *Proc. Workshop on Visual Motion*, pages 126–138, March 1989.

[9] M. Collins, R. Schapire, and Y. Singer. Logistic regression, adaboost and bregman distances. *Machine Learning*, 48(1-3), 2002.

[10] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *PAMI*, 24(5):603–614, May 2002.

[11] P. Dollár, Z. Tu, and S. Belongie. Supervised learning of edges and objects boundaries. In *CVPR*, 2006.

[12] T. Drummond and R. Cipolla. Application of lie algebras to visual servoing. *IJCV*, 37(1):21–41, 2000.

[13] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59(2):167–181, 2004.

[14] D. J. Fleet, M. J. Black, and O. Nestares. Bayesian inference of visual motion boundaries. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, July 2002.

[15] L. Guan, J.-S. Franco, and M. Pollefeys. 3D occlusion inference from silhouette cues. In *CVPR*, 2007.

[16] D. J. Heeger. Optical flow using spatiotemporal filters. *IJCV*, 1:270–302, 1988.

[17] L. Herault and R. Horaud. Figure-ground discrimination: A combinatorial optimization approach. *PAMI*, 15(9):899–914, September 1993.

[18] T. Heskes, K. Albers, and B. Kappen. Approximate inference and constrained optimization. In *UAI*, pages 313–320, 2003.

[19] S. Konishi, A. L. Yuille, J. M. Coughlan, and S. C. Zhu. Statistical edge detection: Learning and evaluating edge cues. *PAMI*, 25(1):57–74, January 2003.

[20] S. Lazebnik and J. Ponce. The local projective shape of smooth surfaces and their outlines. *IJCV*, 63(1):65–83, 2005.

[21] C. Liu, W. T. Freeman, and E. H. Adelson. Analysis of contour motions. In *NIPS*, 2006.

[22] S. Mahamud, L. R. Williams, K. K. Thornber, and K. Xu. Segmentation of multiple salient closed contours from real images. *PAMI*, 25(4):433–444, April 2003.

[23] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, volume 2, pages 416–423, July 2001.

[24] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *PAMI*, 26(5):530–549, May 2004.

[25] B. A. Maxwell and S. J. Brubaker. Texture edge detection using the compass operator. In *BMVC*, volume II, pages 549–558, September 2003.

[26] J. McDermott and E. H. Adelson. The geometry of the occluding contour and its effect on motion interpretation. *Journal of Vision*, 4(10):944–954, 2004.

[27] G. Mori. Guiding model search using segmentation. In *ICCV*, 2005.

[28] O. Nestares and D. J. Fleet. Probabilistic tracking of motion boundaries with spatiotemporal predictions. In *CVPR*, pages 358–365, 2001.

[29] S. A. Niyogi. Detecting kinetic occlusion. In *ICCV*, 1995.

[30] A. S. Ogale, C. Fermüller, and Y. Aloimonos. Motion segmentation using occlusions. *PAMI*, 27(6):988–992, June 2005.

[31] X. Ren, C. C. Fowlkes, and J. Malik. Cue integration for figure/ground labeling. In *NIPS*, 2005.

[32] X. Ren, C. C. Fowlkes, and J. Malik. Figure/ground assignment in natural images. In *ECCV*, 2006.

[33] X. Ren and J. Malik. Learning a classification model for segmentation. In *ICCV*, volume 1, pages 10–17, 2003.

[34] M. G. Ross and L. P. Kaelbling. Learning static object segmentation from motion segmentation. In *AAAI*, 2005.

[35] M. Ruzon and C. Tomasi. Color edge detection with the compass operator. In *CVPR*, pages 160–166, June 1999.

[36] J. Sato and R. Cipolla. Affine reconstruction of curved surfaces from uncalibrated views of apparent contours. *PAMI*, 21(11):1188–1197, 1999.

[37] A. Sethi, D. Renaudie, D. Kriegman, and J. Ponce. Curve and surface duals and the recognition of curved 3d objects from their silhouettes. *IJCV*, 58(1):73–86, 2004.

[38] P. Smith, T. Drummond, and R. Cipolla. Layered motion segmentation and depth ordering by tracking edges. *PAMI*, 26(4):479–494, April 2004.

[39] A. Stein and M. Hebert. Combining local appearance and motion cues for occlusion boundary detection. In *BMVC*, 2007.

[40] A. N. Stein and M. Hebert. Local detection of occlusion boundaries in video. In *BMVC*, 2006.

[41] A. N. Stein and M. Hebert. Using spatio-temporal patches for simultaneous estimation of edge strength, orientation, and motion. In *Beyond Patches Workshop at CVPR*, 2006.

[42] W. B. Thompson, K. M. Mutch, and V. A. Berzins. Dynamic occlusion analysis in optical flow fields. *PAMI*, 7:374–383, 1985.

[43] R. Vaillant and O. D. Faugeras. Using extremal boundaries for 3-D object modeling. *PAMI*, 14(2):157–173, 1992.

[44] Y. Weiss. Interpreting images by propagating bayesian beliefs. In *Advances in Neural Information Processing Systems*, volume 9, page 908, 1997.

[45] J. Xiao and M. Shah. Accurate motion layer segmentation and matting. In *CVPR*, 2005.

[46] P. Yin, A. Criminisi, J. Winn, and I. Essa. Tree-based classifiers for bilayer video segmentation. In *CVPR*, 2007.

[47] S. Yu and J. Shi. Segmentation with pairwise attraction and repulsion. In *ICCV*, July 2001.

[48] A. L. Yuille. CCCP algorithms to minimize the Bethe and Kikuchi free energies: convergent alternatives to belief propagation. *Neural Computation*, 14(7):1691–1722, July 2002.