Edge tracking for motion segmentation and depth ordering

P. Smith, T. Drummond and R. Cipolla

Department of Engineering
University of Cambridge
Cambridge CB2 1PZ,UK
{pas1001|twd20|cipolla}@eng.cam.ac.uk

Abstract

This paper presents a new theoretical framework for motion segmentation based on the motion of tracked region edges. By considering the visible edges of an object, constraints may be placed on the motion labelling of edges. This enables the most likely region labelling and layer ordering to be established, thus producing a segmentation.

An implementation is outlined and demonstrated on test sequences containing two motions. The image is divided into regions using a colour edge-based segmentation scheme and the normal motion of these edges is tracked. The EM algorithm is used to partition the edges and fit the best two motions accordingly. Hypothesising each motion in turn to be the foreground motion, the labelling constraints can be applied and the frame segmented. The hypothesis which best fits the observed edge motions indicates the layer ordering and leads to a very accurate segmentation.

1 Introduction

The segmentation of a video sequence into moving objects is a first stage in many further areas of video analysis. For example, in the fields of tracking and video indexing it is desirable to divide the elements in a sequence into foreground and background objects and perform independent analysis on the different elements. Further, the MPEG-4 standard represents sequences as objects on a series of layers, and so these objects and layers must be identified to encode a video sequence.

Early motion segmentation techniques estimated the global motion field (optic flow) across the frame and then grouped together pixels with a similar motion. These require smoothing to obtain a reliable flow field and this smoothing cuts across motion boundaries, which means that the edges of objects cannot be accurately determined. Some success has been achieved by assuming that the motion consists of a number of *layers*, each with a different, smooth, flow field [1, 2, 7, 15]. The number of layers and the motion models may be simultaneously estimated within a mixture model framework [1, 15]. Another common framework for estimating the otions of the layers is the *dominant motion* approach. This technique robustly estimates the motion which fits the most pixels (the dominant motion), segments out these pixels, and then repeats [8, 11].

The pixel-based optic flow techniques suffer from the problem that they do not consider the wider scope of the frame, and do not encode the knowledge that object motion is spatially coherent. A Markov Random Field (MRF) approach (e.g. [10, 15]) can enforce spatial coherency, but its emphasis on clustering pixels together can once more lead to inaccurate motion boundaries.

Clues to the location of motion boundaries can be found by considering the *structure* of the image. Smooth regions are expected to move coherently, and changes in motion are more likely to occur at edges in the image. One approach to make use of this, the normalized cuts method of Shi and Malik [12], combines both the motion and intensity information of pixels into a weighted graph; the problem is then reduced to finding the best partitioning of this graph.

Rather than computing the image motion, an alternative starting point is an intensity- or colour-based segmentation of the image, extracting information about the image structure prior to any motion analysis. The assumption made by this approach is that the boundaries of the segmented regions are a superset of the motion boundaries; the task is then to identify the correct motion labelling for the different regions. Work by Thompson [14] followed this route and, more recently, Bergen and Meyer [3] and Moscheni and Dufaux [9] have had some success in iteratively merging neighbouring regions if their motion fields are similar. The current paper shows that comparable results can be obtained by only considering the motion of the edges of regions.

The *ordering* of the layers in a sequence is usually glossed over in the literature and the dominant motion is often considered to be that of the background. The correct identification of the layer ordering is essential dealing with occlusion and producing an accurate segmentation. In one of the few papers to consider the ordering of motion layers, Bergen and Meyer [3] note that errors in the motion estimation of pixels generally occur when the pixels have been occluded and so regions containing errors near a region boundary can be labelled as being occluded by that neighbouring region. The current paper shows that the constraints on the edge labelling provide enough information to determine the ordering.

This paper follows the region merging paradigm, and demonstrates that only the motion of the edges of regions need be considered. Edges are tracked between frames and partitioned according to their real-world motions. Given knowledge of the layer ordering, the regions can be labelled to be consistent with the majority of the tracked edges. By hypothesising and testing all possible layer orders, the ordering which fits the most edges can be found and the associated segmentation extracted.

The theoretical framework for analysing edge motions is presented in Section 2. Section 3 explains the current implementation of this theory, with experimental results presented in Section 4.

2 Theoretical framework

Segmentation attempts to identify the edges of the object (or objects) in the image; it is therefore the edges in the image which are important. Edges are also very good features to consider: they can be found more reliably than corners and their long extent means that a number of measurements may be taken along their length, leading to a more accurate estimation of the motion. In this section we outline a framework by which we can resolve the region labelling from these edge motions.

2.1 The image motion of edges

Edges in an image are due to the texture of objects, or their boundaries in the scene. Edges can also be due to shadows and specular reflections, but these are not considered at this stage (however, see Figure 4 for an example). It is assumed that as an object moves all of the edges associated with the object move. Hence edges in one frame may be compared with those in the next and partitioned according to different real-world motions.

The motion in the sequence is assumed to be *layered* i.e. one motion takes place completely in front of another. In the case of two motions these are called foreground and background. It is also assumed that any occluding boundary (the edge of a foreground object) is visible in the image. From this it follows that each edge segment obeys the same real-world motion along its length. Given these assumptions, we can state

Axiom 1 *If an edge undergoes a motion then at least one of the two regions it bounds must also undergo that motion.*

Axiom 2 *The occluding boundary between two objects moves according to the foreground motion.*

These are sufficient to identify the foreground motion and label the region via two consequences:

Consequence 1 *A region belongs to the foreground layer only if all of its bounding edges are foreground.*
Proof (by contradiction) Suppose one of the edges of a region belonging to the foreground layer obeys the background motion. Axiom 1 then implies that the region on the other side of this edge must be a background region, and thus this edge is part of an occluding boundary. Axiom 2 states that, in this case, the edge must have the foreground motion, which violates the supposition.

Consequence 2 *No junction may have a single foreground edge. At edge junctions where two different layers meet, two of the edges must belong to the foreground motion.*
Proof If one edge at a junction obeys the foreground motion then, by Axiom 1, one of the regions that it bounds must have the foreground motion. One of the other edges at this junction must also bound this region, and according to Consequence 1 this edge must also have the foreground motion.

2.2 The labelling of motions and regions

Given a set of edges labelled according to their motions this framework enables the ordering of the motions to be determined and the regions to be labelled. In order to label regions from the edges, the ordering of the motion layers must be known. This may, in theory, be found by observing the junctions where edges of different motions meet since, according to Consequence 2, in every case the same motion should be represented by two edges. However, in practice, some edges are mislabelled and this can cause the analysis of edge junctions to yield the incorrect result.

An alternative approach is to hypothesise possible orderings and label regions accordingly. The axioms and consequences of Section 2.1 provide strong constraints on the

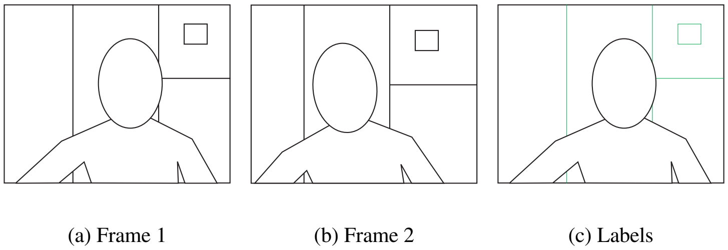

(a) Frame 1 (b) Frame 2 (c) Labels

Figure 1: *Line drawing.* (a) and (b): Two frames as a line drawing. (c) Edges labelled according to rigid motions (black is motion A). The edge labels are sufficient to identify motion A as foreground and segment the image.

edge labelling given a region labelling. The correct ordering is provided by the hypothesis under which the constrained edges which best fit the observed edge labelling.

The regions may be labelled by noting that every region entirely surrounded by foreground edges is foreground; every other case is background (Consequence 1). The edge labelling can then be tested by checking that every edge of a foreground region is foreground and every other edge is background.

2.2.1 A synthetic example

Figure 1 illustrates two frames from a synthetic sequence. In Figure 1(c) the edges from frame 1 have been partitioned according to the two rigid motions in the scene. If motion A (black) were foreground, the head and torso, being entirely surrounded by motion A edges, would be foreground (by Consequence 1). All other regions would be background. This exactly satisfies the edge labelling.

If motion B (grey) were foreground, only the rectangle in the top right corner is entirely surrounded by foreground edges; every other region would be background. However, many of these regions also have motion B edges, which violates Axiom 1. The more likely ordering is therefore that motion A is foreground, with the head and torso regions foreground and everything else background.

3 Implementation

3.1 Finding edges

To implement this framework, regions and edges must first be located in the image. The implementation presented here uses a scheme developed by Sinclair [13]; other edge-based schemes, such the morphological segmentation used in [3] are also suitable.

Colour edges are found in the image and Voronoi seed points for region growing are then found at the locations furthest from these edges. Regions are grown, by pixel colour, with image edges acting as hard barriers. The result is a series of connected, closed, region edges generated from the original fragmented edges (e.g. Figure 3(a)). The edges of these regions are divided into edge segments for labelling. (An edge segment is a length of edge between two edge junctions.)

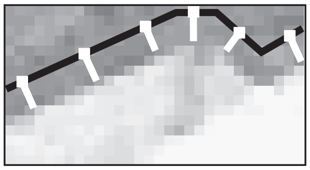


Figure 2: *Edge Tracking.* Initialise tracking nodes along the edge (in black), then search normal to the edge to find the new location (white lines) The best-fit motion is the one that minimises the squared distance error between the tracking nodes and the edge.

3.2 Labelling edges

The second stage of the implementation is the partitioning of edges into sets corresponding to different real-world motions. In this paper the case of two motions (background and foreground) is considered; the estimation of more than two motions within the same framework is a subject of ongoing research. It is assumed that the image motions due to the real-world motion may be modelled (to within a few pixels) by an affine transformation. The tracking method used in this paper is robust to small errors and so the affine assumption works well while having the benefit of simplicity.

Labelling the edges is a circular problem:

1. If an edge partitioning is known, the two motions can be calculated as those which best fit the two sets of edges
2. If two motions are known, the edges can be partitioned according to the motion that they fit best

To resolve this problem an iterative approach, the Expectation-Maximisation (EM) algorithm [5], is used. Given an initial partitioning, Stage 1 calculates the motions. These motions may then be used to update the partitioning (Stage 2) and the process repeated until convergence. Alternatively, the process may be initialised with two motions.

In this implementation, both stages of the EM process make novel use of technology adapted from contour and model tracking [4, 6]. For each edge in the first frame, tracking nodes are assigned at regular intervals along the edge (see Figure 2). The motion of these nodes are considered to be representative of the edge motion. This approach provides a vast reduction in the data to be processed; there are around $1,400$ tracking nodes in a typical frame.

3.2.1 Estimating motion from edges

The tracking nodes from the first frame are mapped into the next according to the current estimated motion. A 1-dimensional search is then made in the direction of the edge normal in order to find a matching edge, based on both image gradient and colour. This distance, d_t, is measured (see Figure 2).

At each tracking node the expected image motion due to motion m (with parameters $\boldsymbol{\alpha}_m$) can be calculated. The best fit solution is the one which minimises

$$\min_{\boldsymbol{\alpha}_m} \sum_e \sum_{t \in e} r_{m,t}^2 \qquad (1)$$

where e indexes the edges and t the tracking nodes. The residual, $r_{m,t}$, is the difference between the measured distance, d_t, and the component of the image motion normal to the edge. This expression may be minimised using least squares.

3.2.2 Partitioning edges by their motion

The probability that edge e_j belongs to motion m may be calculated from the residual, assuming a Gaussian distribution:

$$E_m^j = \exp\left(-\sum_{t \in e_j} r_{m,t}^2 / 2\sigma^2\right) \tag{2}$$

σ is set at 2 pixels, which is a typical RMS error. When performing EM the edge probabilities are fed back and used in Stage 2, where each edge contributes to each motion in proportion to the probabilities.

3.2.3 Solution by Expectation-Maximisation

The process is initialised by assigning one motion to be the mean (the best fit to all edges) and the other to be the null motion. This is a reasonable guess since in typical video sequences either the camera is stationary, or is tracking the foreground object. The other motion will be closer to the mean motion than to the null motion, thus aiding convergence.

The EM iteration process is repeated until convergence (i.e. until there is no significant change in probabilities), or for 20 iterations. Figures 3(b) and 3(c) illustrate the final estimated motions for one of the test sequences.

3.3 Region labelling

The final stage is region labelling. This presents another circular problem—the regions may not be labelled until the ordering of the layers is known, but the ordering of the layers is most reliably determined from testing a region labelling (see Section 2.2). This may be resolved by hypothesising and testing both possible motion orderings. A segmentation is produced under each case and the one which best fits the edge labelling is used.

3.3.1 Labelling regions in the presence of outliers

The region labelling rule, Consequence 1, states that if a region is entirely surrounded by foreground edges then it is foreground, else it is background. This is a powerful rule, and as a result it is very sensitive to labelling errors. A single erroneous background edge will cause the regions on either side to be mislabelled. Since shorter edges have less tracking nodes and so a higher chance of an error, the region labelling scheme is adjusted to allow a small number of mislabelled edge pixels around a region.

A vote of foreground versus background pixels is totalled around the region. Each edge bounding the region contributes its number of pixels to the vote, voting as either foreground or background depending upon which is more likely for the edge. Uncertain edges, those with a probability of less than r_1 (60%), do not vote. If the foreground vote is greater than r_2 of the total ($r_2 = 85\%$) then the region is foreground. This yields an initial segmentation (e.g. Figure 3(e)), but one in which the labelling constraints have not been enforced

3.3.2 Enforcing labelling constraints

A simple region labelling based on the original edge labels does not automatically lead to one entirely consistent with the edge labels. A single foreground edge between a pair of a background regions will have no effect on the region labelling, but violates Axiom 1. In making the region labelling robust to outliers, some background edges will also be ignored.

To ensure that all labels are consistent, the edges are relabelled from the regions. Each region labels its edges with the same motion as itself. Where another region has already labelled an edge as foreground, this may not be over-written by a background edge (at an occluding boundary it is the foreground edge which is visible). Given this new, definite, edge labelling the regions may be relabelled under the basic rule of Consequence 1.

Some regions may not have been labelled under Section 3.3.1 if all of their edges were uncertain. These are also labelled when the constraints are enforced, taking on the labelling of their neighbours. If there are clusters of unlabelled regions, this constraint step may need to be repeated until no more changes are observed in the region labelling.

3.3.3 Identifying the correct layer order

For the two-motion case considered in the paper, the region labelling and constraint propagation is performed twice: with motion A as foreground and then motion B, to give two possible segmentations. The correct segmentation is the one where the edge labels from the final segmentation best fit the observed edges. (i.e. the one which leads to the least residual error across all of the tracking nodes,

$$\sum_{e \in A} \sum_{t \in e} r^2_{A,t} + \sum_{e \in B} \sum_{t \in e} r^2_{B,t} \tag{3}$$

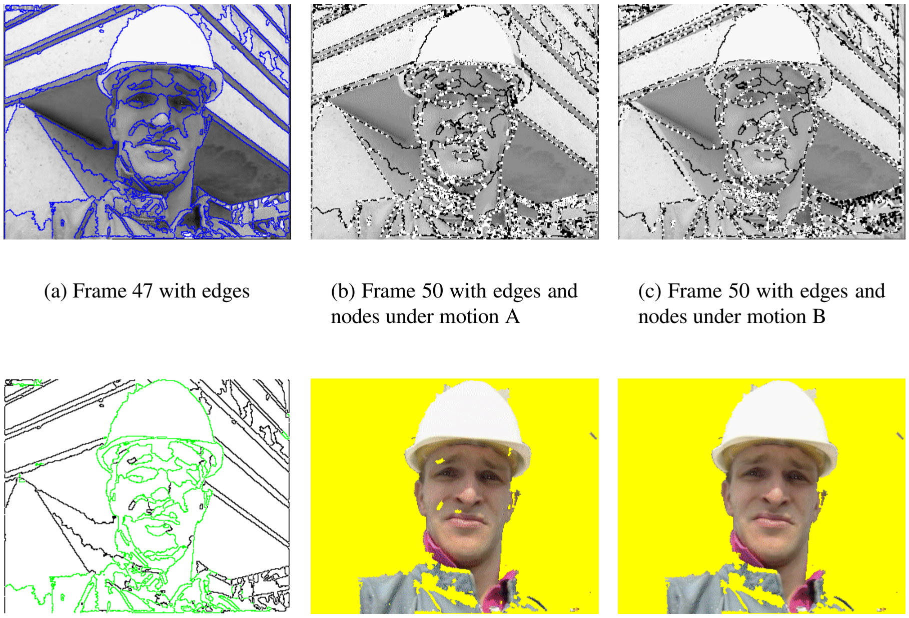
where A is the set of motion A edges and B the set of motion B edges.)

4 Experimental Results

4.1 “Foreman”

The algorithm described in this paper has been successfully tested on a number of different motion sequences, of which two are presented here. The first example considers two frames from the “foreman” sequence, commonly used to test motion segmentation schemes. In this part of the sequence, the foreman’s head moves slightly to his right. Figure 3(a) shows the frame to be segmented, overlaid with the detected region edges. The EM motion estimation and partitioning converges quickly, and it can be seen from Figures 3(b) and 3(c) that motion A provides a good match to the background and motion B a good match to the foreground.

The edge labels are shown in Figure 3(d). Almost every edge has been labelled correctly—the few outliers are short edges with very few tracking nodes (and thus more susceptible to noise). Hypothesising motion A as foreground leads to a significantly



(a) Frame 47 with edges

(b) Frame 50 with edges and nodes under motion A

(c) Frame 50 with edges and nodes under motion B

(d) Labelled edges. Grey is motion B

(e) Initial foreground segmentation

(f) Foreground segmentation after enforcing constraints

Figure 3: *"Foreman" example.* The foreman moves his head slightly to the left. Despite the small motion, the pixels and edges are matched well. Motion B is identified as foreground and the head is segmented well.

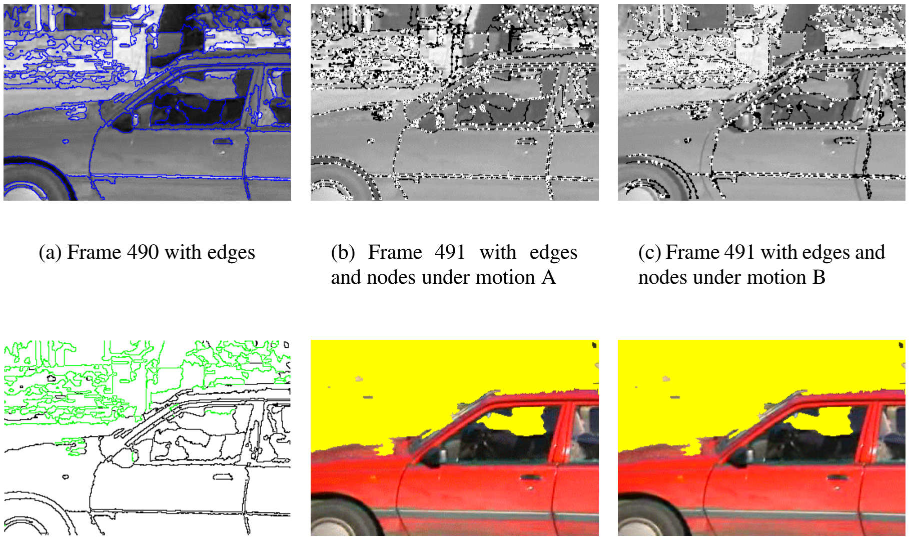
higher squared residual error, assuming independent Gaussian distributions for each node. Consequently, motion B is foreground.

The initial region segmentation (with motion B as foreground) is very good. The relaxation stage successfully enforces the constraints and corrects some of the remaining errors. The system now labels 97.0% of the pixels correctly, compared with a hand-selected segmentation of the regions. The majority of the errors, on his shoulders, could perhaps also be considered correct since his shoulders do not move significantly as his head turns. This is a significantly better result than in [9] and comparable to [3], except that the segmentation used in this paper provides a more accurate occlusion boundary.

4.2 "Car"

The "car" sequence was recorded with a hand-held MPEG video camera. The camera tracks the car as it moves across to the left. This is an unusual sequence as the dominant motion is that of the foreground object, an object which also contains a window through which the background can be seen.

The EM again converges well, with the motions correctly extracted and most edges labelled correctly. When the two possible hypotheses are considered, the errors due to motion A are smaller, again by a significant amount. The foreground motion is therefore



(a) Frame 490 with edges

(b) Frame 491 with edges and nodes under motion A

(c) Frame 491 with edges and nodes under motion B

(d) Labelled edges. Grey is motion B

(e) Initial foreground segmentation

(f) Foreground segmentation after relaxation

Figure 4: *"Car" example*. In the "car" sequence a car is tracked by the camera. Since both the car and the background motion are horizontal this provides a tricky segmentation problem. Motion A is identified as the foreground motion and the car is successfully segmented, with even the window correctly assigned to the background.

correctly identified as motion A. The final segmentation is good, with 97.3% of pixels correct (again, compared with a manual segmentation). The very top of the car, and part of the front, are segmented into the background; this is because these parts of the car have visible reflections of the background, which move as the background. The window has been correctly segmented as background.

5 Conclusions and discussion

The most important elements in segmenting a video sequence are the the edges of the objects. If we assume that the object edges can be seen in the frame, it is then simply a case of identifying these edges. In this paper a framework has been developed which shows how analysis of labelled edges in a frame can be used to identify the ordering of motion layers and extract the correct labelling of regions in the frame.

Implementing this framework has shown it to produce very good segmentations. A good edge labelling is provided by performing region segmentation and then tracking edges between frames. Techniques have then been developed which correctly identify the layer order and label the regions even in the presence of some edge mislabelling. The result is a very accurate motion segmentation from an analysis of only two frames. The implementation is also efficient since only a small fraction of the pixels in the frame are considered.

Ongoing work looks at the segmentation of multiple motion layers and the extension of the tracking and segmentation techniques to multiple frames. In this larger system, the system presented here provides a useful bootstrap for an accurate segmentation of a whole motion sequence.

Acknowledgements

This research was funded by a United Kingdom EPSRC studentship, with a CASE award from the AT&T Laboratories, Cambridge, UK. Thanks go to David Sinclair for the use of his image segmentation code and to Ken Wood for some useful discussions.

References

[1] S. Ayer and H. S. Sawhney. Layered representation of motion video using robust maximum-likelihood estimation of mixture models and MDL encoding. In *Proc. 5th International Conference on Computer Vision*, pages 777–784, Cambridge, MA, USA, June 1995.

[2] S. Baker, R. Szeliski, and P. Anandan. A layered approach to stereo reconstruction. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 434–441, Santa Barbara, CA, June 1998.

[3] L. Bergen and F. Meyer. Motion segmentation and depth ordering based on morphological segmentation. In *Computer Vision—ECCV '98 (Proc. 5th European Conference on Computer Vision)*, pages 531–547, Freiburg, Germany, June 1998.

[4] A. Blake and M. Isard. *Active Contours*. Springer-Verlag, 1998.

[5] A. P. Dempster, H. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistical Society Series B*, 39:1–38, 1977.

[6] T. Drummond and R. Cipolla. Visual tracking and control using lie algebras. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 652–657, Fort Collins, CO, June 1999.

[7] S. Hsu, P. Anandan, and S. Peleg. Accurate computation of optical flow by using layered motion representations. In *International Conference on Pattern Recognition*, pages 743–746, Jerusalem, Israel, October 1994.

[8] M. Irani, B. Rousso, and S. Peleg. Computing occluding and transparent motions. *International Journal of Computer Vision*, 12(1):5–16, January 1994.

[9] F. Moscheni and F. Dufaux. Region merging based on robust statistical testing. In *SPIE Proc. Visual Communications and Image Processing*, Orlando, Florida, USA, March 1996.

[10] J. M. Odobez and P. Bouthemy. Separation of moving regions from background in an image sequence acquired with a mobile camera. In *Video Data Compression for Multimedia Computing*, pages 283–311. Kluwer Academic Publisher, 1997.

[11] H. S. Sawhney and S. Ayer. Compact representations of videos through dominant and multiple motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):814–830, August 1996.

[12] J. Shi and J. Malik. Motion segmentation and tracking using normalized cuts. In *Proc. 6th International Conference on Computer Vision*, pages 1154–1160, Bombay, India, January 1998.

[13] D. Sinclair. Voronoi seeded colour image segmentation. AT&T Laboratories Cambridge, submitted for publication, 1999.

[14] W. Thompson. Combining motion and contrast for segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(6):543–549, November 1980.

[15] Y. Weiss and E. H. Adelson. A unified mixture framework for motion segmentation: Incorporating spatial coherence and estimating the number of models. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 321–326, San Francisco, CA, June 1996.