Motion Segmentation by Tracking Edge Information over Multiple Frames

Paul Smith, Tom Drummond, and Roberto Cipolla

Department of Engineering
University of Cambridge
Cambridge CB2 1PZ, UK
{pas1001|twd20|cipolla}@eng.cam.ac.uk
http://www-svr.eng.cam.ac.uk/~pas1001/

Abstract. This paper presents a new Bayesian framework for layered motion segmentation, dividing the frames of an image sequence into foreground and background layers by tracking edges. The first frame in the sequence is segmented into regions using image edges, which are tracked to estimate two affine motions. The probability of the edges fitting each motion is calculated using 1st order statistics along the edge. The most likely region labelling is then resolved using these probabilities, together with a Markov Random Field prior. As part of this process one of the motions is also identified as the foreground motion.
Good results are obtained using only two frames for segmentation. However, it is also demonstrated that over multiple frames the probabilities may be accumulated to provide an even more accurate and robust segmentation. The final region labelling can be used, together with the two motion models, to produce a good segmentation of an extended sequence.

1 Introduction

Video segmentation is a first stage in many further areas of video analysis. For example, there is growing interest in video indexing – where image sequences are indexed and retrieved by their content – and semantic analysis of an image sequence requires moving objects to be distinguished from the background. Further, the emerging 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.

A recent trend in motion segmentation is the use of layers [7, 16]. This avoids some of the traditional multiple-motion estimation problems by assuming that motion within a layer is consistent, but layer boundaries mark motion discontinuities. The motions and layers may be estimated using the recursive dominant motion approach [1, 10], or by fitting many layers simultaneously [11, 15, 16, 17].

Motion estimation is poor in regions of low texture, and here the structure of the image has to play a part. Smooth regions are expected to move coherently, and changes in motion are more likely to occur at edges in the image. A common approach is to use the local image intensity as a prior when assigning pixels to layers [10, 15, 17]. The normalized cuts method of Shi and Malik [12] can combine both the motion and intensity information of pixels into a weighted graph, for which the best partition has then to be found.

Alternatively, the image structure may be considered before the motion estimation stage by performing an initial static segmentation of the frame based on pixel colour or intensity. This reduces the problem to one of identifying the correct motion labelling for the each region. Both Bergen and Meyer [2] and Moscheni and Dufaux [9] have had some success in merging regions with similar motion fields.

This paper concentrates on the edges in an image. Edges are very valuable features to consider, both for motion estimation and segmentation. Object tracking is commonly performed using edge information (in the form of snakes), while image segmentation techniques naturally use the structure cues given by edges. If an image from a motion sequence is already segmented into regions of similar colour or intensity along edges, it is clear that a large proportion of the motion information will come from these edges rather than the interior of regions. This paper shows how this edge information alone is sufficient to both estimate and track motions, and label image regions.

Many papers on motion segmentation avoid the question of occlusion or the ordering of layers. Occluded pixels are commonly treated as outliers which the algorithm has to be able to tolerate, although reasoned analysis and modelling of these outliers can be used to retrieve the layer ordering and identify occluded regions [2, 3, 16]. With the edge-based method proposed in this paper, the problem of occluded pixels is greatly reduced since it is only the occluding boundary, and not the region below, which is being tracked. Furthermore, the relationship between edges and regions inherently also depends on the layer ordering, and this is extracted as an integral part of the algorithm.

This paper describes a novel and efficient framework for segmenting frames from a sequence into layers using edge motions. The theory linking the motions of edges and regions is outlined and a Bayesian probabilistic framework developed to enable a solution for the most likely region labelling to be inferred from edge motions. This work extends the approach first proposed in [14], developing more powerful probabilistic models and demonstrating that evidence may be accumulated over a sequence to provide a more accurate and robust segmentation.

The theoretical and probabilistic framework for analysing edge motions is presented in Sect. 2. The current implementation of this theory is outlined in Sect. 3, with experimental results presented in Sect. 4.

2 Theoretical Framework

Edges in the image are important features since the desired segmentation divides the image along occluding edges of the foreground object (or objects) in the image. Edges are also very good features to consider for motion estimation: 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 their motion. However, segmentation ultimately involves regions, since the task is one of labelling image pixels according to the motions. If it is assumed that the image is already segmented into regions along edges, then there is a natural link between the regions and the edges. In this section the relationship between the motion of regions and edges is outlined and a probabilistic framework is developed to enable a region labelling to be estimated from edge data.

2.1 The Image Motion of Region 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. It is assumed that as an object moves all of the edges associated with the object move, and hence edges in one frame may be compared with those in the next and partitioned according to different real-world motions.

The work in this paper assumes that the motion in the sequence is layered i.e. one motion takes place completely in front of another. Typically the layer farthest from the camera is referred to as the background, with foreground layers in front of this. It is also assumed that any occluding boundary (the edge of a foreground object) is visible in the image. With regions in the image defined by the edges, this implies that each region obeys only one motion, and an edge which is an occluding boundary will have the motion of the occluding region. This enables a general rule to be stated for labelling edges from regions:

Labelling Rule: **The layer to which an edge belongs is that of the nearer of the two regions which it bounds.**

2.2 Probabilistic Formulation

There are a large number of parameters which must be solved to give a complete motion segmentation. In this section a Bayesian framework is developed to enable the most likely value of these parameters to be estimated.

The complete model of the segmentation, $\boldsymbol{M}$, consists of the elements $\boldsymbol{M} = \{\boldsymbol{\Theta}, \boldsymbol{F}, \boldsymbol{R}\}$ where

$\boldsymbol{\Theta}$ is the parameters of the layer motion models,
$\boldsymbol{F}$ is the depth ordering of the motion layers,
$\boldsymbol{R}$ is the motion label (layer) for each region.

The region edge labels are not part of the model, but are completely defined by $\boldsymbol{R}$ and $\boldsymbol{F}$ from the Labelling Rule of Sect. 2.1.

Given the image data $\boldsymbol{D}$ (and any other prior information assumed about the world), the task is to find the model $\boldsymbol{M}$ with the maximum probability given this data and priors:[1]

$$\max_{\boldsymbol{M}} \mathrm{P}\left(\boldsymbol{M}|\boldsymbol{D}\right) = \max_{\boldsymbol{RF\Theta}} \mathrm{P}\left(\boldsymbol{RF\Theta}|\boldsymbol{D}\right) . \tag{1}$$

[1] Throughout this paper, max is used to also represent argmax, as frequently both the maximum value and the parameters giving this are required.

This can be further decomposed into a motion estimation component and region labelling:

$$\max_{\boldsymbol{RF\Theta}} \mathrm{P}\left(\boldsymbol{RF\Theta}|\boldsymbol{D}\right) = \max_{\boldsymbol{RF\Theta}} \mathrm{P}\left(\boldsymbol{\Theta}|\boldsymbol{D}\right) \mathrm{P}\left(\boldsymbol{RF}|\boldsymbol{\Theta D}\right) . \tag{2}$$

At this stage a simplification is made: it is assumed that the maximum *value* (not the model parameters which give this) of (2) is independent of the motion, and thus the motion parameters $\boldsymbol{\Theta}$ can be maximised independently of the others. The expression to be maximised is thus

$$\underbrace{\max_{\boldsymbol{\Theta}} \mathrm{P}\left(\boldsymbol{\Theta}|\boldsymbol{D}\right)}_{\mathrm{a}} \underbrace{\max_{\boldsymbol{RF}} \mathrm{P}\left(\boldsymbol{RF}|\boldsymbol{\Theta D}\right)}_{\mathrm{b}} , \tag{3}$$

where the value of $\boldsymbol{\Theta}$ used in term (b) is that which maximises term (a). The two components of (3) can be evaluated in turn: first (a) and then (b).

(a) Estimating the Motions $\boldsymbol{\Theta}$. The first term in (3) estimates the motions between frames, which this may be estimated by tracking features. As outlined in Sect. 2.1, edges are robust features to track and they also provide a natural link to the regions which are to be labelled.

In order to estimate the motion models from the edges it is necessary to know which edges belong to which motion, which is not something that is known a priori. In order to resolve this, another random variable is introduced, $\boldsymbol{e}$, which is the labelling of an edge: which motion the edge obeys. The motion estimation can then be expressed in terms of an Expectation-Maximisation problem [5]:

$$\begin{cases} \mathrm{P}\left(\boldsymbol{e}|\boldsymbol{\Theta_n D}\right) & \text{E-stage} \\ \max_{\boldsymbol{\Theta}_{n+1}} \mathrm{P}\left(\boldsymbol{\Theta}_{n+1}|\boldsymbol{eD}\right) \mathrm{P}\left(\boldsymbol{e}|\boldsymbol{\Theta}_n \boldsymbol{D}\right) & \text{M-stage} . \end{cases} \tag{4}$$

Starting with an initial guess of the motions, the expected edge labelling is estimated. This edge labelling can then be used to maximise the estimate of the motions, and the process iterates until convergence.

(b) Estimating the Labellings $\boldsymbol{R}$ and $\boldsymbol{F}$. Having obtained the most likely motions, the remaining parameters of the model $\boldsymbol{M}$ can be maximised. Once again, the edge labels are used as an intermediate step. The motion estimation allows the edge probabilities to be estimated, and from Sect. 2.1 the relationship between edges and regions is known. Term (3b) can be augmented by the edge labelling $\boldsymbol{e}$, which must then be marginalised, giving

$$\max_{\boldsymbol{RF}} \mathrm{P}\left(\boldsymbol{RF}|\boldsymbol{\Theta D}\right) = \max_{\boldsymbol{RF}} \sum_{\boldsymbol{e}} \mathrm{P}\left(\boldsymbol{RF}|\mathrm{e}\right) \mathrm{P}\left(\mathrm{e}|\boldsymbol{\Theta D}\right) , \tag{5}$$

since $\boldsymbol{R}$ and $\boldsymbol{F}$ are conditionally independent of $\boldsymbol{D}$ given $\boldsymbol{e}$ (which is entirely defined by $\boldsymbol{R}$ and $\boldsymbol{F}$).

The second term, the edge probabilities, can extracted directly from the motion estimation stage. The first term is more difficult to estimate, and it is easier to recast this using Bayes' Rule, giving

$$\mathrm{P}\left(\boldsymbol{RF}|\boldsymbol{e}\right) = \frac{\mathrm{P}\left(\boldsymbol{e}|\boldsymbol{RF}\right)\mathrm{P}\left(\boldsymbol{RF}\right)}{\mathrm{P}\left(\boldsymbol{e}\right)} . \tag{6}$$

The maximisation is over $\boldsymbol{R}$ and $\boldsymbol{F}$, so $\mathrm{P}\left(\boldsymbol{e}\right)$ is constant. It can also be assumed that the priors of $\boldsymbol{R}$ and $\boldsymbol{F}$ are independent, and any foreground motion is equally likely, so $\mathrm{P}\left(\boldsymbol{F}\right)$ is constant. The last term, the prior probability of a particular region labelling $\mathrm{P}\left(\boldsymbol{R}\right)$, is not constant, which leaves the following expression to be evaluated:

$$\max_{\boldsymbol{RF}} \sum_{\boldsymbol{e}} \mathrm{P}\left(\boldsymbol{e}|\boldsymbol{RF}\right)\mathrm{P}\left(\boldsymbol{R}\right)\mathrm{P}\left(\boldsymbol{e}|\boldsymbol{\Theta D}\right) . \tag{7}$$

The $\mathrm{P}\left(\boldsymbol{e}|\boldsymbol{RF}\right)$ term is very useful. $\boldsymbol{e}$ is only an intermediate variable, and is entirely defined by the region labelling $\boldsymbol{R}$ and the foreground motion $\boldsymbol{F}$. This probability therefore takes on a binary value – it is 1 if that edge labelling is implied and 0 if it is not. The sum in (7) can thus be removed, and the $\boldsymbol{e}$ in the final term replaced by a function of $\boldsymbol{R}$ and $\boldsymbol{F}$ which gives the correct edge labels:

$$\max_{\boldsymbol{RF}} \underbrace{\mathrm{P}\left(\mathrm{e}\left(\boldsymbol{R},\boldsymbol{F}\right)|\boldsymbol{\Theta D}\right)}_{\mathrm{a}} \underbrace{\mathrm{P}\left(\boldsymbol{R}\right)}_{\mathrm{b}} . \tag{8}$$

The variable $\boldsymbol{F}$ takes only a discrete set of values (in the case of two layers, only two: either one motion is foreground, or the other). Equation (8) can therefore be maximised in two stages: $\boldsymbol{F}$ can be fixed at one value and the expression maximised over $\boldsymbol{R}$, and the process then repeated with other values of $\boldsymbol{F}$ and the global maximum taken.

The maximisation over $\boldsymbol{R}$ can be performed by hypothesising a complete region labelling and then testing the *evidence* (8a) – calculating the probability of the edge labelling given the regions and the motions – and the *prior* (8b), calculating the likelihood of that particular labelling configuration. An exhaustive search is impractical, and in the implementation presented here region labellings are hypothesised using simulated annealing.

3 Implementation

This section outlines the implementation of the framework presented in Sect. 2 for two layers (foreground and background), with the motions of each modelled by an affine motion. The basic implementation is divided into three sections (see Fig. 1):

1. Find edges and regions in the first frame
2. Estimate the motions and edge probabilities
3. Label the regions and foreground motion

The second two stages can then be continued over subsequent frames and the edge probabilities accumulated.

(a) Initial static segmentation

(b) Edges labelled as motion 1 or motion 2

(c) Foreground regions

Fig. 1. 'Foreman' segmentation from two frames. The foreman moves his head very slightly to the left between frames, but this is enough to accurately estimate the motions and calculate edge probabilities. The foreground motion can then be identified and the regions labelled to produce a good segmentation of the head.

3.1 Finding Edges and Regions

To implement the framework outlined in Sect. 2, regions and edges must first be located in the image. The implementation presented here uses a scheme developed by Sinclair [13] but other edge-based schemes, such the morphological segmentation used in [2], are also suitable.

Under Sinclair's scheme, colour edges are found in the image and 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 (see Fig. 1(a)). The edges referred to in this paper are the region boundaries: each boundary between two distinct regions is an edge.

3.2 Estimating the Motions Θ

As described in Sect. 2.2, the problem of labelling the segmented regions can be divided into two stages: first estimating the motions and then the motion and region labelling. In order to estimate the motions, features are tracked from one frame to the next; the obvious features to use are the region edges. The motion is parameterised by a 2D affine transformation, which gives a good approximation to the small inter-frame motions.

Multiple-motion estimation is a circular problem. If it were known which edges belonged to which motion, these could be used to directly estimate the motions. However, edge motion labelling cannot be performed without knowing the motions. In order to resolve this, Expectation-Maximisation (EM) is used [5], implementing the formulation outlined in (4) as described below.

Edge Tracking. Both stages of the EM process make use of group-constrained snake technology [4, 6]. For each edge, tracking nodes are assigned at regular

a) b)

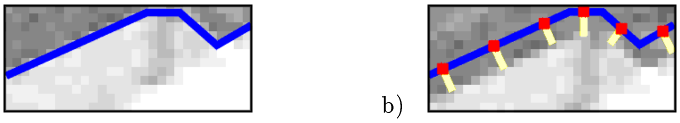


Fig. 2. Edge tracking example. (a) Edge in initial frame. (b) In the next frame the image edge has moved. Tracking nodes are initialised along the model edge and then search normal to the edge to find the new location. The best-fit motion is the one that minimises the squared distance error between the tracking nodes and the edge.

intervals along the edge (see Fig. 2). The motion of these nodes are considered to be representative of the edge motion (there are around $1,400$ tracking nodes in a typical frame). The tracking nodes from the first frame are mapped into the next according to the current best guess of the motion. A 1-dimensional search is then made along the edge normal (for 5 pixels either direction) to find a matching edge pixel based on colour image gradients. The image distance d, between the original node location and its match in the next image, is measured (see Fig. 2(b)).

At each tracking node the expected image motion due to the 2D affine motion $\boldsymbol{\theta}$ can be calculated. The best fit solution is the one which minimises the residual:

$$\min_{\boldsymbol{\theta}} \sum_{e} \sum_{t \in e} \left(d_t - n(\boldsymbol{\theta}, t)\right)^2 \tag{9}$$

over all edges e and tracking nodes t, where d_t is the measurement and $n(\boldsymbol{\theta}, t)$ the component of the image motion normal to the edge at that image location. This expression may be minimised using least squares, although in practice an M-estimator (see, for example, [11]) is used to provide robustness to outliers.

Maximisation: Estimating the Motions. Given the previous estimate of the motions $\boldsymbol{\Theta}$, all tracking nodes are mapped into the next frame according to both motions. From each of the two possible locations a normal search is performed as described above and the best match found (or 'no match' is reported if none is above a threshold). These distances are combined into the estimate of the affine motion parameters (9) in proportion to the current edge probabilities.

Expectation: Calculating Edge Probabilities. For simplicity, it is assumed that the tracking nodes along each edge are independent and that tracker errors can be modelled by a normal distribution. Experiments have shown Gaussian statistics to be a good fit, and although the independence assumption is less valid (see Sec. 3.3), it still performs satisfactorily for the EM stage.

By assuming independence, the edge probability under each motion is the product of the tracking node probabilities. Each tracking node tries to find a match under each of the two motions, yielding either an error distance d_i or finding no match above a threshold (denoted by $d_i = \otimes$). There are three distinct cases when matching under the two motions: a match is found under both motions, under neither motion, or under only one motion. The probability distributions for each case have been modelled from data by considering an ideal solution.

Match Found Under Both Motions. The errors under both motions are modelled by normal distributions and, a priori, both are equally likely. The probability of a tracker belonging to motion 1 is given by the normalised probability

$$\mathrm{P}\left(\text{Motion } 1 | d_1 d_2\right) = \frac{1}{1 + \exp\left(-\frac{1}{2\sigma_1^2}\left(d_2^2 - d_1^2\right)\right)} \tag{10}$$

where, from data, $1/2\sigma_1^2 = 0.3$. The probability of it belonging to motion 2 is, of course, $(1 - \mathrm{P}\left(\text{Motion } 1 | d_1 d_2\right))$.

Match Found Under Only One Motion. A Gaussian was found to be a good fit to experimental data:

$$\mathrm{P}\left(\text{Motion } 1 | d_1, d_2 = \otimes\right) = \alpha e^{-\beta d_1^2}\,, \tag{11}$$

with $\alpha = 0.97$ and $\beta = 0.0265$. The same equation holds, but with d_2, if the single match were under motion 2 instead.

No Match Found Under Either Motion. In this case, no information is available and a uniform prior is used:

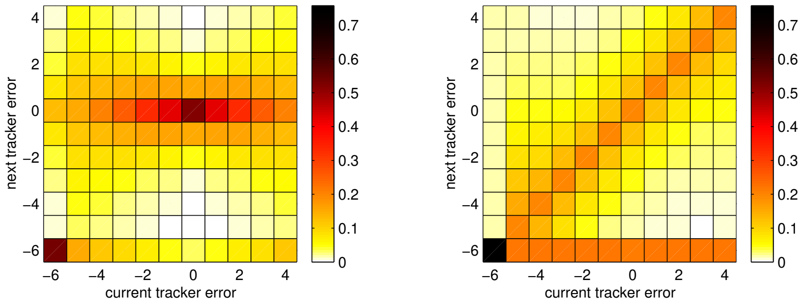
$$\mathrm{P}\left(\text{Motion } 1 | d_1 = d_2 = \otimes\right) = 0.5\,. \tag{12}$$

Initialisation and Convergence. The EM is initialised with a guess of the two motions $\boldsymbol{\Theta}$. For the first frame, the initial guesses are zero motion (the camera is likely to be stationary or tracking the foreground object) and the mean motion, estimated from the initial errors of all the edges. For subsequent frames, a velocity estimate is used (see Sec. 3.4). For the first iteration of EM, the tracker search path is set at 20 pixels to compensate for a possible poor initialisation.

Convergence is gauged by considering the Maximum A Posteriori labelling of each edge (either motion 1 or motion 2 depending on which is most likely). If no edge changes labelling between two iterations then convergence is assumed. The maximum number of iterations is set at 40, which takes around 3 seconds on a 300MHz Pentium II.

3.3 Labelling Regions $\boldsymbol{R}$ and finding the Layer Order $\boldsymbol{F}$

Having estimated the most likely motions $\boldsymbol{\Theta}$, the second term of (3) can be maximised. This finds the most likely region labelling and identifies the motion most likely to be foreground. Using (8), this can be performed by hypothesising possible region and foreground motion labellings and calculating their probabilities, selecting the most probable.



(a) Correct Motion (b) Incorrect Motion

Fig. 3. Markov chain transition probabilities. These are used to calculate the probability of observing a particular sequence of tracking node errors along an edge. A residual of -6 corresponds to no match being found at that tracker under that motion.

Region Probabilities from Edge Data. Given a hypothesised region labelling and layer ordering, the edges can all be labelled as motion 1 or motion 2 by following the Labelling Rule from Sect. 2.1. The probability of this region labelling given the data (term (8a)) is given by the probability of the edges having these labels.

The edge probabilities used in the EM of Sec. 3.2 made the assumption that tracking node errors were independent. While this is acceptable for the EM, under this assumption the edge probabilities are too confident and can result in an incorrect region labelling solution. As a result, a more suitable edge probability model was developed. Correlations between tracking nodes along an edge can be decoupled using Markov chains, which encode 1st order probabilistic relationships. (Used, for example, by MacCormick and Blake [8] to make their contour matching more robust to occlusion.) These higher-order statistics cannot be used for the EM since they are only valid at (or near) convergence. However, to ensure that the EM solution maximises the Markov chain edge probabilities, the EM switches to the Markov chain model near convergence.

The Markov chain models the relationship between one tracking node and the next along an edge, giving the probability of a tracking node having a certain residual d_i given the residual at the previous tracking node. These transition probabilities were estimated from data for the cases where an edge is matched under the correct motion and under the incorrect motion, and the modelled probabilities can be seen in Fig. 3. It is found that under the correct motion, a low residual distance is likely, and the residuals are largely independent (unless no match is found, in which case it is highly likely that the next tracking node will also find no match). Under the incorrect motion, the residual distances are highly correlated, and there is always a high probability that no match will be found.

There are two models of edge tracking node sequence formation: either motion 1 is correct and motion 2 incorrect, or vice versa. Both are considered equally likely a priori. The chain probability is calculated from the product of the transition probabilities, and it is assumed that the probabilities under the correct and incorrect motions are independent and so the two can be multiplied to give the hypothesis probability. Finally, the two hypothesis probabilities must be normalised by their sum to give the posterior edge probability.

The region probability given the data is the probability that all its edges obey the correct motions. It is assumed that the edges are independent, so (8a) can be evaluated by multiplying together all region edge probabilities under the edge labelling implied by $\boldsymbol{R}$ and $\boldsymbol{F}$.

Region Prior Term (8b) encodes the a priori region labelling. This is implemented using a Markov Random Field (MRF), where the prior probability of a region's labelling depends on its immediate neighbours. Neighbours are considered in term of the fractional boundary length such that the more of a region's boundary adjoins foreground regions, the more likely the region is to be foreground.

The prior model was estimated from examples of correct region segmentations. An asymmetric sigmoid is a good fit to the data, where it is more likely to have a promontory of foreground in a sea of background than an inlet in the foreground (f is the percentage of foreground boundary around the region):

$$\mathrm{P}\left(\boldsymbol{R}\right) = \frac{1}{1 + \exp\left(-10\left(f - 0.4\right)\right)} \tag{13}$$

Solution by Simulated Annealing. In order to minimise over all possible region labellings, simulated annealing (SA) is used. This begins with an initial guess and then repeatedly tries flipping individual region labels one by one to see how the change affects the overall probability. (This is a simple process since a single region label change only causes local changes.)

The annealing process is initialised with a guess based on the edge probabilities. According to Sec. 2.1, foreground regions are entirely surrounded by foreground edges. This can be used as a region-labelling rule, although it is found that it works better if slightly diluted to allow for outliers. The initial region labelling labels as foreground any region with more than 85% of its edges having a high foreground probability.

Taking each region in turn,[2] they are considered both as foreground and as background and the probability of each hypothesis is calculated. In each case, the prior $\mathrm{P}\left(\boldsymbol{R}\right)$ can be calculated by reference to the current labels of its neighbours and the evidence calculated from the edge probabilities (using the edge motions implied by the neighbouring region labels and the layer ordering).

[2] Each pass of the data labels each region, but the order is shuffled each time to avoid systematic errors.

In the first pass, the region is then assigned by a Monte Carlo approach, i.e. randomly according to the two probabilities. However, the cycle is repeated and as the iterations progress, these probabilities are forced to saturate such that after around 30 iterations, all regions will be being assigned to their most likely motion. The annealing process continues until no changes are observed in a complete pass of the data, which takes about 40 iterations.

The random element in SA enables some local minima to be avoided. However, it was found that local minima were still a problem under any reasonable cooling timetable and, under some situations, the optimal solution was only found around a third of the time. This is solved by repeating the annealing process a number of times: 10 maximisations are performed, which gives a 99% probability of finding the optimal solution. The entire maximisation of (8) takes around 2 seconds on a 300MHz Pentium II.

Determining Depth Ordering $\boldsymbol{F}$ and Optimal Segmentation $\boldsymbol{R}$ Moving between region and edge labels, as in the annealing process, requires the layer ordering $\boldsymbol{F}$ to be known. This identifies the occluding edges of regions, and a different layer ordering can result in a very different segmentation. The most likely ordering, and segmentation, is the one which is most consistent with the edge probabilities i.e. the $\boldsymbol{R}$, given $\boldsymbol{F}$, with the highest probability.

The annealing process is thus performed twice, once for each possible value of $\boldsymbol{F}$, first with motion 1 as foreground and then motion 2 as foreground. The segmentation with the greater posterior probability identifies the most likely foreground motion and the segmentation.

3.4 Multiple Frames

The maximisation outlined in Sects 3.2 and 3.3 can be performed over only two frames with good results (see, for an example, Fig. 1(c)). However, over multiple frames more evidence can be accumulated to give a more robust estimate. It is always the segmentation of frame 1 that is being maximised, so after comparing frame 1 to frame 2, frames 1 and 3 are compared, and then 1 and 4 and so on.

Initialisation The estimated motions and edge probabilities between frames 1 and 2, can be used to initialise the EM stage for the next frame. The motion estimate is that for the previous frame incremented by the velocity between the previous two frames. The edge labelling is initialised to be that implied by the region labelling of the previous frame, and the EM begins at the M-stage.

Combining statistics The probability that an edge obeys motion 1 over n frames is the probability that it obeyed motion 1 in each of the n frames. This can be calculated from the product of the probabilities for that edge over all n frames, if it is assumed that the edge probabilities are independent between frames.

To perform the region and foreground labelling based on the cumulative statistics, the method described in Sec. 3.3 is followed but using the cumulative edge statistics rather than those from just one frame.

Occlusion Over only two frames the problem of occlusion has been ignored as it has little effect on the outcome. When tracking over multiple frames, this becomes a significant problem. The foreground/background labelling for edges, however, allows this problem to be overcome. For each edge labelled as background according to the previous frame's region labelling, the tracking nodes' locations in the current image (under the background motion) are projected back into frame 1 under the foreground motion. If they fall into regions currently labelled as foreground, they are marked as occluded and they do not contribute to the tracking for that edge. All trackers are also tested to see if they project to outside the frame under the current motion and, if so, they are also ignored.

Segmenting a Sequence The segmentation of an entire sequence may be approximated by projecting the foreground regions into the other frames of the sequence according to the foreground motion at each frame. These regions may then be used as a 'template' to cut out the object in each of the subsequent frames (see Figs 5 and 6).

4 Results

Figure 1 shows the segmentation from the standard 'foreman' sequence based on two neighbouring frames. Between frames the head moves a few pixels to the left. The first frame is statically segmented (Fig. 1(a)) and then EM run between this frame and the next to extract the motion estimates. Figure 1(b) shows the edge labels based on how well they fit each motion after convergence. It can be seen that the EM process picks out most of the edges correctly, even though the motion is small. The edges on his shoulders are poorly labelled, but this is due to the shoulders' motion being even smaller than that of the head. The correct motion is selected as foreground with very high confidence (a posterior probability of about 99%) and the final segmentation, Fig. 1(c), is very good despite some poor edge labels. In this case the MRF region prior is a great help in producing a plausible segmentation. On a 300MHz Pentium II, it takes around 7 seconds to produce the motion segmentation from an initial static region segmentation.

The effect of using multiple frames can be seen in Fig. 4. Accumulating the edge probabilities over several frames allows random errors to be removed and edge probabilities to be reinforced. The larger motions between more widely separated frames also removes ambiguity. It can be seen that over time the consensus among many edges on the shoulders is towards the foreground motion. The accumulated edge probabilities have a positive effect on the region segmentation, which settles down after a few frames to a very accurate solution. If the

Fig. 4. Evolution of the 'foreman' segmentation, showing the edge probabilities and segmentations of frames 47–52 as the evidence is accumulated. The edge probabilities become more certain and small errors are removed, resulting in an improved region segmentation.

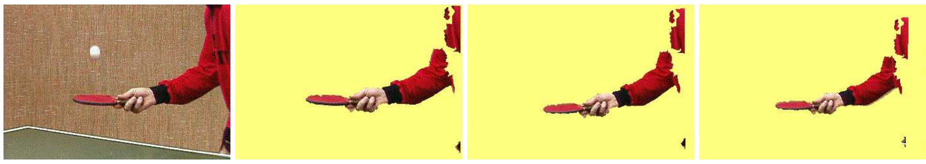


Fig. 5. Multiple-frame segmentation of the 'tennis' sequence. The camera zooms out while the arm slowly descends. Shown is the original frame 29 and then the foreground segmentation of part of the sequence, showing every 5th frame. The final region labelling is used to segment all frames in a second pass of the data.

segmentation were continued over a large number of frames then the errors from assuming affine motion become would become significant (particularly as the foreman tilts his head back and opens his mouth), and the segmentation would break down. Dealing with non-affine motions is a significant element planned for further work.

Figures 5 and 6 show some frames from extended sequences segmented using this method. In the 'tennis' sequence (Fig. 5) the arm again does not obey the affine motion particularly well (and the upper arm and torso hardly obey it at all), but is still tracked and segmented well over a short sequence of frames. The 'car' sequence, Fig. 6, is atypical – it has a large background motion (around 10 pixels per frame), a hole in the foreground object, and the dominant motion is the foreground. However, it is still segmented very cleanly (including the window) and the correct motion is identified as foreground. In this case the layer ordering is rather unsure over 2 frames (70%/30%), but over many frames the edge labellings are reinforced and the final decision is clearly in favour of the correct labelling. The affine motion fits the side of the car well over a large number of frames.

Fig. 6. Multiple-frame segmentation of the 'car' sequence. The camera pans to the left to track the car. Shown is the original frame 490 and then the foreground segmentation of part of the sequence, showing every 5th frame. The final region labelling is used to segment all frames in a second pass of the data.

5 Conclusions and Future Work

This paper develops and demonstrates a novel Bayesian framework for segmenting a video sequence into foreground and background regions based on tracking the edges of an initial region segmentation between frames. It is demonstrated that edges can be reliably tracked and labelled between frames of a sequence and are sufficient to label regions and the motion ordering.

The EM algorithm is used to simultaneously estimate the two motions and the edge probabilities (which can be robustly estimated using a Markov chain along the edge). The correct foreground motion and region labelling can be identified by hypothesising and testing to maximise the probability of the model given the edge data and a MRF prior. The algorithm runs quickly and the results are very good over two frames. Over multiple frames the edge probabilities can be accumulated resulting in a very accurate and robust region segmentation.

The current implementation considers only two layers under affine motions. Future work will concentrate on extended multi-frame sequences, since over a longer sequence the edge motions cannot be well modelled by an affine motion model. Disoccluded edges also appear, and should be incorporated into the model. Both problems may be solved by using the tracked edges to assist in the resegmentation of future frames in the sequence, which then behave as new 'key frames' for the segmentation process described in this paper. This would allow the system to adapt to non-rigid, non-affine motions over the longer term.

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 many useful discussions.

References

[1] S. Ayer, P. Schroeter, and J. Bigün. Segmentation of moving objects by robust motion parameter estimation over multiple frames. In *Proc. 3rd European Con-*

ference on Computer Vision, volume II, pages 317–327, Stockholm, Sweden, May 1994.

[2] L. Bergen and F. Meyer. Motion segmentation and depth ordering based on morphological segmentation. In *Proc. 5th European Conference on Computer Vision*, volume II, pages 531–547, Freiburg, Germany, June 1998.

[3] M.J. Black and D.J. Fleet. Probabilistic detection and tracking of motion discontinuities. In *Proc. 7th International Conference on Computer Vision*, volume I, pages 551–558, Kerkyra, Greece, September 1999.

[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 '99*, 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 *Proc. 12th International Conference on Pattern Recognition*, pages 743–746, Jerusalem, Israel, October 1994.

[8] J. MacCormick and A. Blake. Spatial dependence in the observation of visual contours. In *Proc. 5th European Conference on Computer Vision*, volume II, pages 765–781, Freiburg, Germany, June 1998.

[9] F. Moscheni and F. Dufaux. Region merging based on robust statistical testing. In *Proc. SPIE Visual Communications and Image Processing '96*, 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. Technical Report 1999.3, AT&T Laboratories Cambridge, 1999.

[14] P. Smith, T. Drummond, and R. Cipolla. Edge tracking for motion segmentation and depth ordering. In *Proc. 10th British Machine Vision Conference*, volume 2, pages 369–378, Nottingham, September 1999.

[15] P. H. S. Torr, R. Szeliski, and P. Anandan. An integrated Bayesian approach to layer extraction from image sequences. In *Proc. 7th International Conference on Computer Vision*, volume II, pages 983–990, Kerkyra, Greece, September 1999.

[16] J.Y.A Wang. and E.H. Adelson. Layered representation for motion analysis. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition '93*, pages 361–366, New York, NY, June 1993.

[17] 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 '96*, pages 321–326, San Francisco, CA, June 1996.