Motion Estimation by Swendsen-Wang Cuts

Adrian Barbu
University of California, Los Angeles
Computer Science Department
abarbu@ucla.edu

Alan Yuille
University of California, Los Angeles
Statistics Department
yuille@stat.ucla.edu

Abstract

Our paper has two main contributions. Firstly, it presents a model for image sequences motivated by an image encoding perspective. It models accreted regions, *where objects appear, as well as motion and motion boundaries. We formulate the problem as probabilistic inference using prior models of images and the motion field. Secondly, it introduces a new algorithm for motion estimation based on Swendsen-Wang Cuts, which performs inference on the image sequence model using bottom-up proposals to guide the search. The algorithm proceeds by first estimating the motion without the boundaries, and then by clustering in the velocity space to obtain initial estimates of the motion boundaries. The algorithm performs MAP estimation by evolving the motion boundaries by a stochastic boundary diffusion algorithm, while improving the motion estimates. Our approach is illustrated on real images of city scenes and on simulated data and can deal with large motions (even 10 pixels or more per frame). We show a brief comparison of Swendsen-Wang Cuts with Graph Cuts and Belief Propagation on the related stereo matching problem.*

1. Introduction

This paper introduces a new algorithm for motion estimation and segmentation. The segmentation partitions the image into regions which contain similar velocity. We view motion estimation as an image encoding problem and develop inference algorithms for this purpose.

Image encoding gives a way to think of motion estimation which differs from more traditional approaches for calculating optical flow. From a coding theory viewpoint, the purpose of motion estimation is to provide an efficient coding scheme for the entire motion sequence. The first image frame is encoded based on static image properties, while the subsequent frames are modeled by the previous frames and the estimated motion. This involves using the image and motion in the first frame to predict the image in the next frame, which will allow us to relax standard assumptions such as the optical flow constraint [7]. Our model assumes that the images consist of regions (possibly corresponding to objects) which can move with different velocities. But we also need models to deal with *accreted* and/or *deleted* sub-regions which appear or disappear due to occlusion or limitations in the prediction model. For example, in Figure 1, the motion prediction model cannot account for two of the legs of the cheetah, the tail, the part of the cheetah that has been occluded by the grass, and the background that was occluded by the cheetah. These effects become serious in image sequences taken by a moving video camera where all the objects in the original image frame have typically vanished from the visual field after a few seconds. In this paper, we are less concerned with detecting deleted regions because they are in the past and so are not useful for encoding (though our algorithm can also detect them).

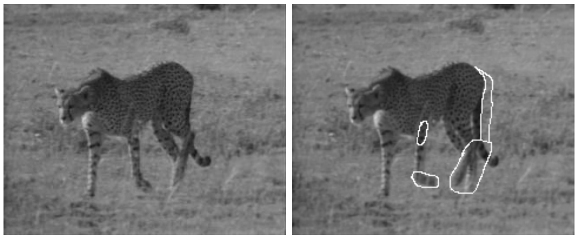

Figure 1. The image sub-regions (circled in white) cannot be explained as pixels moving from the previous frame because either they were not previously visible or the motion prediction model was not accurate enough. These *accreted* sub-regions need to be modeled separately.

We formulate image encoding as Maximum a Posteriori (MAP) estimation. Our algorithm proceeds in three stages. The first two stages perform inference on simplified approximations to the problem and provide initialization for the final stage. More precisely, the first stage uses the novel Swendsen-Wang Cuts algorithm [1] to estimate the motion field and the accreted sub-regions but without attempting to perform the motion segmentation. The second stage uses this result to estimate the motion segmentation using clustering techniques. The third stage attempts to perform full MAP estimation and estimate the motion, the motion segmentation and the accreted sub-regions simultaneously. This is done using Swendsen-Wang Cuts and stochastic boundary diffusion.

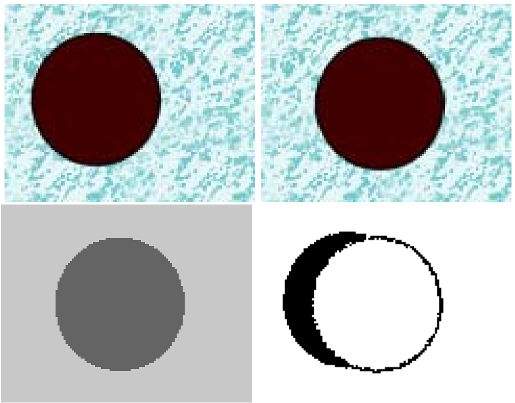

Figure 2. First row: Input sequence of a translating black circle over a textured background. Second row: the estimated horizontal motion (left) using our algorithm and the accreted pixel map (right). In the motion estimation image (left), light grey indicates zero motion, dark grey is motion to the right. In the accreted pixel map (right), black indicates accreted pixels and white is non-accreted.

Our image encoding approach, implemented for two-frame motion, automatically finds accreted sub-regions and estimates the motion where it is well defined. For example, Figure 2 shows a pair of images of size 120x95 (left and middle) where the black circle translates 13 pixels to the right and 2 pixels down, revealing the texture behind it. Our algorithm correctly estimates the displacement of the circle and detects the accreted pixels (shown in black in the right image) and their motion, even though the foreground object has no features or texture.

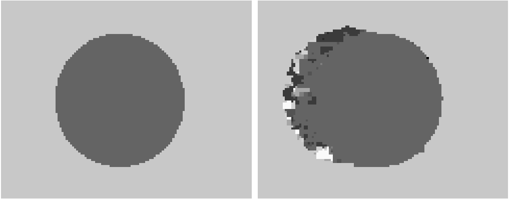

Figure 3. The estimated horizontal motion using our accretion model (left) and without our accretion model (right). As in previous figure, light grey indicates zero motion, dark grey is motion to the right, white is motion to the left. Observe that the motion of the new pixels is incorrectly estimated if the accretion model is not used (right).

For comparison, Figure (3) shows the motion estimation with and without modeling the accreted sub-regions. If the accreted sub-regions are not modeled, then the pixels that were behind the circle have their motion incorrectly estimated.

The structure of this paper is as follows. First we briefly describe previous work in Section (2). Next we introduce our image sequence model, motivated by image encoding, in Section (3). Section (4) describes the first stage of our algorithm, which uses Swendsen-Wang Cuts to estimate the motion, and shows results. Section (5) describes the second two stages of the algorithm and gives results of the complete algorithm. Section (6) compares the performance of two Swendsen-Wang Cuts variants with Graph Cuts and Belief Propagation on the classical stereo problem[10].

This work was sponsored by NSF SGER grant IIS-0240148 and NIH grant RO1-EY 012691-04.

2 Previous Work

There has been considerable work on motion estimation. The current literature mainly focuses on two approaches. The first approach, for example [2, 3], calculates the spatiotemporal gradient $\nabla I = (I_x, I_y, I_t)$ and extracts local motion estimates using the optical flow constraint [7]. Prior probabilities, typically expressed by Markov Random Fields, are used to enforce the computed motion flow to be smooth [7, 18], or piecewise smooth. But the optical flow constraint becomes problematic if the motion is more than a few pixels per time frame, because the discretization of the spatiotemporal gradient becomes inaccurate, or if the intensity of objects change over time (our model does not require the intensity to be constant in time). Moreover, it is hard to deal with accreted sub-regions although the concept of motion layers [16] gives partial ability to deal with this. Some of these limitations can be overcome by ignoring the optical flow constraint and measuring the motion by window correlation, for example see [8]. The second approach to motion estimation [13] is to track feature points and then interpolate the motion of the remaining pixels. This approach can be very effective for some applications but it is limited because it throws away a lot of information in the image. Overall, none of these approaches are suitable for our goal of formulating motion estimation as image encoding.

In binocular stereo, occluded pixels have also been modeled by energy function (Bayesian) methods by [5]. However, the occluded pixels are treated in a simpler manner and only the geometric prior is used to assign disparity values to the occluded pixels. In our approach, the pixel will prefer to be assigned to the motion region containing nearby pixels of similar intensity. In section 6 we will compare our Swendsen-Wang Cuts with the Graph Cuts and Belief Propagation algorithms.

3 The image sequence model

This section introduces our image sequence model. In this paper, we only consider two image frames. Our model assumes that the two images consist of regions which are moving with different velocities. We formulate the problem as Bayesian inference which enables us to predict based on variables which must be inferred. This involves: (i) a model for how each region in I_1 predicts the position and intensity of the corresponding region in I_2, (ii) a model for the intensity of accreted regions in I_2, (iii) prior probabilities for the velocities in each region, and (iv) prior probabilities for the motion segmentation.

The variables used are as follows. The *velocity field* $\vec{V} = \{\vec{v}(\vec{x})\}$ defined for all points $\vec{x}$ in the second image. The *accretion map* $A = \{A(\vec{x})\}$ defined for all points $\vec{x}$ in the second image. $A(\vec{x}) = 0$ if $\vec{x}$ is accreted (i.e. has no corresponding point in the first image) and $A(\vec{x}) = 1$ otherwise. The *segmentation* R which partitions the second image into regions r. We use the notation $R(\vec{x}) = r$ to mean that $\vec{x} \in r$. There are also variables $\{T_r\}$ used for defining prior probabilities on the velocities, which are specified by $T^r = (\vec{v}_r, \sigma_r^2)$ associated to a velocity $\vec{v}_r$, and a covariance σ_r^2 for each region r. In addition, there are variables $\{H_r\}$ defining probability models on the intensities of the regions, and $\{H_r^p\}$ for probabilities of how the intensity of regions change over time.

The motion sequence model consists of a prediction term $P(I_2|I_1, \vec{V}, R, \{H_r\}, \{H_r^p\}, A)$ which predicts the intensity I_2 of the second image conditioned on the intensity I_1 of the first, the accretion field A, the velocity field $\vec{V}$, the motion segmentation R, and the image models $\{H_r\}, \{H_r^p\}$ for all $r \in R$ is:

$$P(I_2 \quad |I_1, \vec{V}, R, \{H_r\}, \{H_r^p\}, A) = \prod_{r \in R} \prod_{\vec{x} \in r; A(\vec{x}) \neq 0} P_{H_r^p}(I_2(\vec{x}) - I_1(\vec{x} - \vec{v}(\vec{x}))) \cdot \prod_{r \in R} \prod_{\vec{x}; A(\vec{x})=0, R(\vec{x})=r} p_{H_r}(I_2(\vec{x})) \tag{1}$$

The model $P_{H_r^p}(I_2(\vec{x}) - I_1(\vec{x} - \vec{v}(\vec{x})))$ is the model for intensity prediction. Observe that it does not assume that corresponding points in the two images have the same intensity values, unlike the optical flow constraint. In this paper, we set $P_{H_r^p}$ to a histogram distribution defined by all the points $\vec{x} \in r$. This distribution is specified by the counts in each bin of the histogram, represented by H_r^p. Alternative distributions could be used if we have more knowledge about the region r – for example, if the region is identified to be water or fire then we could use distributions defined for synthesizing such motion patterns [15, 11].

The intensities in the accreted sub-regions are defined to be generated by probability models P_{H_r}. These are also represented by histogram distributions, represented by H_r. But other models, such as those described in [14], will be used in future work (the histogram distribution is one of the four basic distributions used to model images in [14]).

We must also specify prior probabilities on the variables $\vec{V}, R$. The prior probability on the velocities is specified by $P(\vec{V}|R, \{T_r\})$ In this paper $T_r = (\vec{v}_r, \sigma_r^2)$ where $\vec{v}_r$ is the mean velocity for the region r with covariance σ_r^2 (both quantities will be estimated). But other choices such as affine motion are also suitable. We also include a term to allow for local fluctuations. This gives an overall prior:

$$P(\vec{V}|R, \{T_r\}) \quad \propto \prod_r \exp\Big(- \alpha \sum_{\vec{x} \in r} \Big[(1/2\sigma_r^2)|\vec{v}(\vec{x}) - \vec{v}_r|^2 + \beta \sum_{\vec{x}' \in \partial \vec{x}} (|\vec{v}(\vec{x}') - \vec{v}(\vec{x})|\Big]\Big)$$

Here α, β are constants and ∂x is the neighborhood region of $\vec{x}$ (see later, for specification).

The prior $P(R)$ is solely based on the length of boundary between different regions. This is a common assumption in segmentation models, see [19], but other more sophisticated priors could be used.

$$P(R) \propto \exp(-\gamma \mathrm{length}(\partial R)) \tag{2}$$

We assume uniform priors on the accretion map A and the image probability distribution parameters $\{H_r\}, \{H_r^p\}$.

By Bayes rule, the posterior probability is:

$$P(\vec{V}, A, R, \{T_r\}, \{H_r\}, \{H_r^p\}|I_1, I_2) \propto P(I_2|I_1, \vec{V}, R, \{H_r\}, \{H_r^p\}, A) \cdot P(\vec{V}|R, \{T_r\}) P(R) \prod_r P(H_r)$$

The rest of the paper concentrates on performing inference on $P(\vec{V}, R, \{T_r\}, \{H_r\}, \{H_r^p\}, A|I_1, I_2)$. Our strategy is described in the next section.

The goal of this paper is not to encode images accurately, the models we are using are not yet adequate for that. But observe that the effectiveness of image encoding of motion sequences occurs because the intensity prediction distributions $\{P_{H_r^p}\}$ have usually far less variance than the intensity distributions $\{P_{H_r}\}$ of the static images (e.g. variances of 10, compared to variances of 1000). Also the space of image motions is typically low-dimensional. Essentially, image motions are have regularities and are largely predictable, which is why encoding schemes like MPEG work so well.

3.1 The 3 Stage Strategy to Estimate the Motion

The task is to estimate $(\vec{V}, A, R, \{H_r\}, \{H_r^p\}, \{T_r\})$ from $P(\vec{V}, A, R, \{H_r\}, \{H_r^p\}, \{T_r\}|I_2, I_1)$. This is a hard estimation problem and we perform it in three stages. The first two stages give initial conditions for the final stage that attempts the full estimation.

The first stage gives an estimate $\vec{V}^*$ of the motion field, A^* of the accretion map, and parameters for global intensity models P_H, P_{H^p}. This stage does not estimate the segmentation R and so the intensity models $\{P_{H_r}, P_{H_r^p}\}$ must be replaced by global models. We use Swendsen-Wang Cuts to estimate $\vec{V}^*, A^*, P_h, P_H^p$ from a distribution $Q_1(\vec{V}, A, H, H^p|I_2, I_1)$ which is an approximation to the marginal of the full posterior distribution $P(\vec{V}, A, H, H_p|I_2, I_1) = \sum_R \sum_{T_r} P(\vec{V}, R, \{H_r\}, \{H_r^p\}, \{T_r\}|I_2, I_1)$.

The second stage estimates R^* and $\{T_r\}$ from a distribution $Q_2(R, \{T_r\}|\vec{V}^*)$. This corresponds to clustering and is similar to K means. $Q_2(R, \{T_r\}|\vec{V}^*)$ is an approximation to $P(R, \{T_r\}|\vec{V}^*)$ obtained from the full posterior $P(\vec{V}, A, R, \{H_r\}, \{H_r^p\}, \{T_r\}|I_2, I_1)$ by marginalization.

Finally, the third stage performs MAP estimate on the full posterior distribution $P(\vec{V}, A, R, \{H_r\}, \{H_r^p\}, \{T_r\}|I_2, I_1)$ using the results of the first two stages as initial conditions. In this third stage, all the variables are updated by an algorithm which combines Swendsen-Wang Cuts with stochastic diffusion.

This three stage algorithm is very effective on the datasets that we tried. Although it is not guaranteed to converge to the MAP estimate it gives good solutions on our dataset. But the algorithm does have limitations. It is certainly possible to construct artificial stimuli for which the estimates from Q_1 and Q_2 will not provide sufficiently good initial conditions to enable the third stage algorithm to converge to the MAP estimate of the full posterior (fortunately, such stimuli do not appear in the natural images we tested our algorithm on). For example, the first stage will give poor initial conditions if the foreground and background both have constant intensity, or if the intensity of one image region is changing over time very differently from the other regions (for example, due to unusual illumination conditions). The second stage will give poor results if two neighbouring regions are moving with similar velocities (so that the boundary between them cannot be detected). The estimate can also be sensitive to the threshold used by the clustering. One way to avoid these potential problems is to use additional information, such as static image segmentation cues to help motion segmentation. Another way, is to use Q_1 and Q_2 as proposal probabilities for a DDMCMC algorithm [14]. Our current work is investigating both possibilities.

4 Stage 1: The motion estimation algorithm

This section describes the first stage of the algorithm. We describe the Swendsen-Wang Cuts algorithm [1], apply it to motion estimation, and give results.

4.1 Swendsen-Wang Cuts

We first review the basic steps of the Swendsen-Wang Cuts algorithm. The algorithm partitions a given graph G into subgraphs. For example, the graph can be the image lattice with 4-nearest neighbor connections. The algorithm uses bottom-up cues to drive a stochastic search to maximize an *a posteriori* probability distribution $P(\pi)$ on the space of graph partitions.

The bottom-up cues give proposal probabilities [14, 1]. The choice of these proposals affects the speed of convergence of the algorithm *but not* the end result.

The bottom-up cues are encoded by weights of the edges of the graph. Intuitively, the bottom-up cues should be an empirical measure of the likelihood that neighboring nodes (e.g. pixels) belong to the same subgraph. The more informative these cues, the faster the algorithm will converge. The bottom-up cues for motion estimation will be given in the next subsection.

The Swendsen-Wang Cuts algorithm acts in the space of graph partitions. At each time step, with current partition state S_1, it does the following:

1. Turns the graph edges on or off as follows:
   - edges between different subgraphs are turned ”off”.
   - the rest of the edges are turned ”on” with probability equal to their weight.
2. Then it randomly picks a connected component C of the graph of ”on” edges.
3. Picks a new label l' for C by sampling from a label reassignment probability $q(l'|C, S_1, G)$. The proposed move is to change the label of C from l to l', obtaining a new state S_2. $q(l'|C, S_1, G)$ is given in the next subsection.
4. Accepts the label change move with probability $\alpha(S_1 \to S_2)$ given below.

The move from S_1 to S_2 is accepted with probability [1]

$$\alpha(S_1 \to S_2) = \min(1, \frac{\prod_{e \in \mathcal{C}(C, V_{l'} - C)} (1 - q_e)}{\prod_{e \in \mathcal{C}(C, V_l - C)} (1 - q_e)} \frac{q(l|C, S_2, G)}{q(l'|C, S_1, G)} \frac{p(S_2|I)}{p(S_1|I)})$$

where the current graph partition is $\pi = V_0 \cup ... \cup V_n$. (3)

4.2 Motion Estimation

In this stage 1 algorithm, we want to estimate the motion by directly finding the motion segmentation. We approximate the marginal of the full distribution by an approximation $Q_1(\vec{V}, A, H, H^p|I_2, I_1)$. This include replacing the smoothness prior on the velocity field by a “robust prior” which is insensitive to motion boundaries. This is similar to the analysis performed Geiger *et al* [4] for the Geman and Geman model [6] and replacing the prior $P(\vec{V}|R, \{T^r\},)$ by $Q_p(\vec{V}) \propto \exp(-\sum_{\vec{x}} \sum_{\vec{x}' \in \partial \vec{x}} \Psi(u(\vec{x}) - u(\vec{x}')) + \Psi(v(\vec{x}) - v(\vec{x}')))$, where (u, v) are the components of $\vec{v}$ and $\Psi(.)$ is a robust potential. This is set to be of form

$$\Psi(\Delta) = |\Delta|, \text{ for } |\Delta < T|, \text{ and } \Psi(\Delta) = T \text{ otherwise.}$$

More precisely,

$$Q_1(\vec{V}, A, H, H^p | I_2, I_1) \propto$$
$$\prod_{\vec{x}; A(\vec{x})=1} P_{H^p}(I_2(\vec{x}) - I_1(\vec{x} - \vec{v}(\vec{x}))) \prod_{\vec{x}; A(\vec{x})=0} p_H(I_2(\vec{x}))$$
$$\cdot \exp\Big(-\alpha \sum_{\vec{x}} \sum_{\vec{x}' \in \partial\vec{x}} \{\Psi(|u(\vec{x}') - u(\vec{x})| + \Psi(|v(\vec{x}') - v(\vec{x})|\}\Big)$$

We now apply Swendsen-Wang Cuts to estimate $(\vec{V}^*, A^*, H, H^p) = \arg\max_{(\vec{V})} Q_1(\vec{V}, A, H, H^p | I_2, I_1)$.

The distribution P_H is calculated from the entire image I_2. The predicted image distribution P_{H^p} is calculated from I_1, I_2 in terms of the current estimates of $\vec{V}$ and A.

Because the accretion map A has no prior term, it can be calculated independently by an EM-type algorithm. Given the current estimate of p_{H^p}, at each pixel $\vec{x}$ one can compute the likelihood ratio $P(A(\vec{x}) = 0)/P(A(\vec{x}) = 1)$. If this ratio is larger than 1, the value of $A(x)$ is set to 0 otherwise it is set to 1. The estimates of p_{H^p} are then updated. This algorithm converges in about 5 iterations and it is used to initialize the accretion map. During the motion estimation computation by Swendsen-Wang Cuts, we only update the accretion map by the above mentioned procedure at the pixels involved in the current repartition step. This is because we assume that at each step the parameters H do not change much.

We now give the details of the Swendsen-Wang Cuts. In this stage 1 algorithm, the subgraphs of Swendsen-Wang Cuts correspond to pixels with common velocity (u, v). Each subgraph is assigned a label $l = (u, v)$. To reduce the search space, we require that u and v take integer values within the intervals $[-\max_u, \max_u]$, $[-\max_v, \max_v]$. This gives $(2\max_u + 1)(2\max_v + 1)$ possible labels. It is straightforward to extend this representation to allow for non-integer (e.g. sub-pixel) velocities at the cost of enlarging the search space and increasing the computation time.

We now define the weights of the edges of the graph (recall that their choice will affect the speed at which we reach the solution). For any two neighboring pixels i and j, the probability that they have the same motion is given by integrating over all motions (u', v') at these locations:

$$q(i,j) = \int_{(u',v')} p(u(i) = u(j) = u', v(i) = v(j) = v')du'dv' \tag{4}$$

Where we set:

$$p(u(i) = u, v(i) = v, u(j) = u', v(j) = v') \propto$$
$$e^{-(|I_2(x_i,y_i) - I_2(x_i - u, y_i - v)| + |I_2(x_j,y_j) - I_2(x_j - u', y_j - v')|)/10}.$$

In practice, the integrand of $q(i, j)$ is typically highly peaked and well approximated by the motion that best fits i and j at the same time. This is given by $(u^*, v^*) = \arg\min s(i, j, u, v)$ where

$$\begin{aligned} s(i,j,u,v) &= |I_2(x_i, y_i) - I_2(x_i - u, y_i - v)| \\ &+ |I_2(x_j, y_j) - I_2(x_j - u, y_j - v)| \end{aligned} \tag{5}$$

We define the edge weights to be:

$$q(i,j) = 0.1 + 0.8\exp(-s(i,j,u^*,v^*)/10) \tag{6}$$

where we have added a uniform term for robustness.

To get an idea of the information conveyed by the weights, Figure 4 shows the horizontal edge weights (left) (The vertical edge weights look similar). To show the effectiveness of the proposals, we also show in Figure 4(right) some connected components (shown in different grey values) of the sampled graph for the circle sequence of Figure 2. It is clear that the Swendsen-Wang proposes, within a few steps, to assign the correct motion for the entire circle, even though the local information is not sufficient to specify exactly what the motion of the circle is (this is the well known aperture problem). In future work, we will extend the label space to affine, 3d motion, and other motion models.

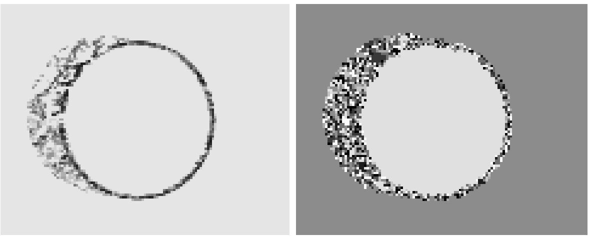

Figure 4. Horizontal edge weights (left) for the circle sequence in Figure 2 (whiter means larger edge weight). The two moving objects are clearly separated. Connected components obtained using these edge weights (right) give proposals for label switching in the Swendsen-Wang Cuts algorithm.

We choose the label reassignment probability to be:

$$q(l'|C, S_1, G) = \begin{cases} a \text{ if } G_{l'} \text{ is adjacent to C,} \\ b \text{ if } l' = (u \pm 1, v \pm 1) \text{ for } \mathrm{l}'' = (\mathrm{u}, \mathrm{v}) \\ \qquad \text{with } \mathrm{G}_{\mathrm{l}''} \text{ adjacent to C} \\ c \text{ else} \end{cases} \tag{7}$$

where a, b, c are parameters (whose values were tuned to give best overall performance).

This means that we propose to reassign C with a large probability to similar motions, with a smaller probability to motions close to the neighboring motions, and with even smaller probability for the remaining motions.

The Swendsen-Wang algorithm has to compute connected components for the whole graph at every step. This can be slow in a graph with tens of thousands of nodes. To speed up, we restrict the algorithm to a small region of interest (usually 15x15 pixels) randomly chosen in the image.

A new and more effective way is to use the Wolff variant [17] which grows a single connected component, from a seed which depends on a ”cry” map of unhappy pixels (with big error) or from the boundary, alternatively.

4.3 Motion estimation results

We applied our motion estimation algorithm to image sequences. Typical results are shown in figures (5,6). For simplicity, we present only the x component of velocity because the motion is mostly in the x direction. For comparison, we also present the motion estimation without our accretion model, see figure (5), and observe that in the presence of the accretion map A, the motion estimation is more accurate. Clearly our model detects the accreted subregions and labels them accordingly.

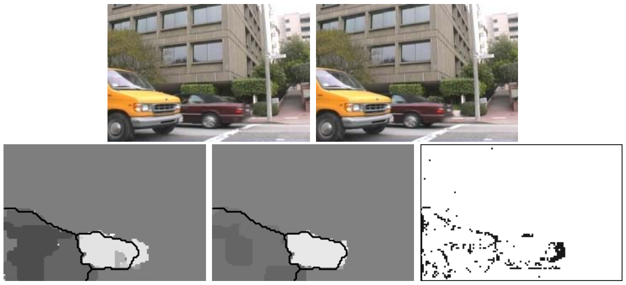

Figure 5. Motion sequence with a static camera and two moving cars (top left and right panels). Motion estimation without the accretion model (bottom left), with the accretion model (bottom middle) and the accretion map A (right). We overimposed a segmentation to show how accurate the motion estimation is. The intensities in the left and middle panel indicate estimated horizontal (x direction) velocity. Light grey indicates zero motion, white is motion to the left, dark grey is motion to the right. Accreted regions are shown in black.

For the woman sequence, see figure (6), we show the motion estimation without our accretion model (left), with our image sequence model but no segmentation (middle) and with our image sequence model and segmentation (right). Some parts of the image have been labeled as accreted subregions because they have constant intensity and modeling them with an image model achieves higher probability than with the motion model.

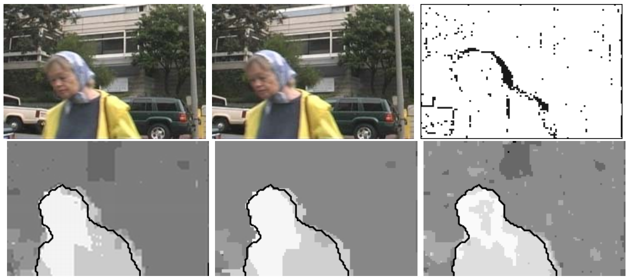

Figure 6. Motion estimation without our accretion model (left), with our image model but without segmentation (middle) and with our image sequence model and segmentation, see next section (right). Both the camera and the person are moving. Same conventions for representing horizontal velocity.

This stage 1 model is robust to motion boundaries but does not explicitly detect them. Hence there is a possibility of "leaking" between different regions which are, in reality, moving at different velocities. Our next section shows how to reduce this effect by using the full probability model.

5 The full motion estimation and segmentation algorithm

In this section we combine the motion estimation algorithm with a simple motion segmentation algorithm using motion clustering and boundary diffusion. We now use the full posterior distribution $P(\vec{V}, A, R, \{T_r\}, \{H_r\}, \{H_r^p\}|I_1, I_2)$. This improves the quality of the results because the image and prior models fill in information at places where the motion estimation cannot be reliably computed. The joint motion estimation and segmentation proceeds as follows:

We begin with the first stage estimation of motion as described in section 4. This gives an initial estimate of $\vec{V}$ and A. This is usually sufficient to detect the main moving regions and their motions.

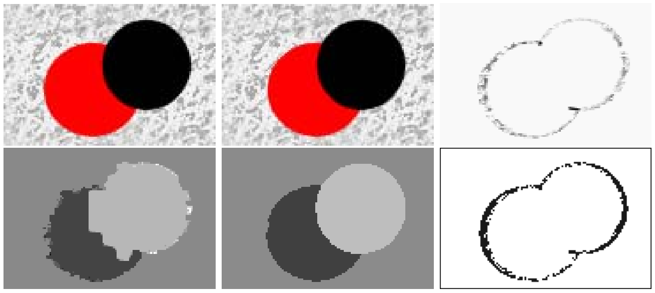

Figure 7. Motion estimation and segmentation of a sequence with two moving circles of constant color. First row, left to right: I_1, I_2, horizontal edge weights of the graph. Second row: initial motion estimation without segmentation, final motion estimation after boundary diffusion, accretion map.

Then we perform the second stage algorithm. This involves performing motion clustering on the velocities. It gives initial coarse motion segmentation, as shown in Figure 7, lower left. This gives an estimate of $R, \{T_r\}$. From now on we will work with the full probability distribution $P(\vec{V}, R, \{T_r\}, \{H_r\}, \{H_r^p\}, A|I_1, I_2)$.

The boundaries of the segmentation are refined by a diffusion process, see next section, that automatically handles topology changes (like the level set curves of [9]) and also allows multiple objects (like the region competition of [19]). At the same time, the motion estimation is updated using the current motion boundaries to "break" the motion smoothness. After convergence, we obtain motion estimation, motion segmentation, and an accretion map.

5.1 The motion estimation

In this step, the motion boundaries are fixed and the motion estimation is updated inside each region. Since the motion boundaries are fixed, the algorithm works on:

$$P(\vec{V}, a, \{H_r^p\}, \{T_r\} \qquad |I_2, I_1, \{H_r\}, R) = \prod_{r \in R} \prod_{\vec{x} \in r; A(\vec{x}) \neq 0} P_{H_r^p}(I_2(\vec{x}) - I_1(\vec{x} - \vec{v}(\vec{x}))) \cdot \prod_{r \in R} \prod_{\vec{x}; A(\vec{x})=0, R(\vec{x})=r} p_{H_r}(I_2(\vec{x})) \cdot \prod_r \exp\Big(-\alpha \textstyle\sum_{\vec{x} \in r} \Big[(1/2\sigma_r^2)|\vec{v}(\vec{x}) - \vec{v}_r|^2 + \beta \displaystyle\sum_{\vec{x}' \in \partial \vec{x}} (|u(\vec{x}') - u(\vec{x})| + |v(\vec{x}') - v(\vec{x})|) \Big] \Big) \tag{8}$$

The algorithm is identical with robust motion estimation described in subsection 4.2. Like there, the accretion map A is updated at each step by computing the likelihood ratio $P(A(\vec{x}) = 0)/P(A(\vec{x}) = 1)$ and deterministically choose $A(\vec{x}) = 0$ if and only if the ratio is greater than 1. The difference is that because we are given the current segmentation, we remove the edges of the Swendsen-Wang graph between different motion regions. This way, at each step, sets of pixels are moved from one velocity label to another, but each time the set being moved is inside a single region. At each move, the estimates of H_r^p and $\{T_r\}$ are updated.

5.2 The boundary diffusion process

We derive boundary diffusion process using the Metropolis-Hastings algorithm applied to the full probability distribution. At the current state S_1 (boundary position, velocity estimates, accretion map plus image and motion models), let B_1 be the set of all pixels which are neighbors of the region boundaries (i.e. the set of pixels with at least one neighbor of a different region label).

In the spirit of the motion estimation algorithm explained in 4.2, for any pixel, we encode the velocity $\vec{v}(\vec{x})$ and the region label r by a label $l = (\vec{v}(\vec{x}), r)$. This way, the motion estimation and region competition becomes just a label changing problem.

Then we randomly pick one pixel $\vec{x}$ of the set B_1 and propose to change its label to l' as follows:

$$q(l'|\vec{x}, S_1, B_1) \propto \begin{cases} 1 & \text{if } l' \text{ is the label of } \vec{x} \text{ or a} \\ & \text{ neighbor of } \vec{x} \\ 0.01 & \text{for all other existing labels} \\ 0.01 & \text{for one non existing label} \end{cases}$$

Let S_2 be the state after the label change. The probability to go from state S_1 to state S_2 is $q(S_1 \to S_2) = \frac{1}{|B_1|} q(l'|\vec{x}, S_1, B_1)$ because the probability to pick $\vec{x}$ is $1/|B_1|$ and the probability to pick l' is $q(l'|\vec{x}, S_1, B_1)$. Similarly, the probability to go back from state S_2 to state S_1 is $q(S_2 \to S_1) = \frac{1}{|B_2|} q(l|\vec{x}, S_2, B_2)$.

Then the acceptance probability for the label change move, based on the Metropolis-Hastings method, is:

$$\alpha(S_1 \to S_2) = \min(1, \frac{q(S_2 \to S_1) p(S_2)}{q(S_1 \to S_2) p(S_1)} \tag{9}$$

The posterior probability has been specified in Eq. 8.

5.3 Results

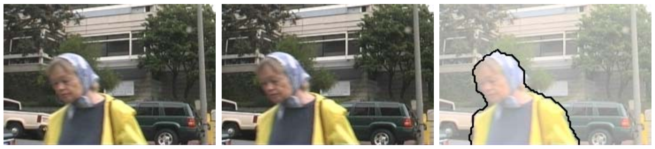

Figure 8. Motion segmentation where both camera and person are moving.I_1(left), I_2(middle), segmentation (right).

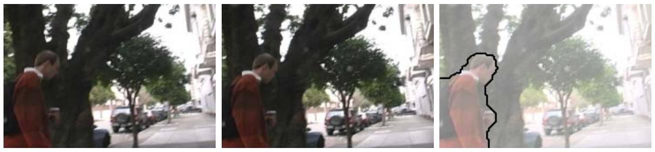

Figure 9. Motion segmentation where both camera and person are moving.I_1(left), I_2(middle), segmentation (right).

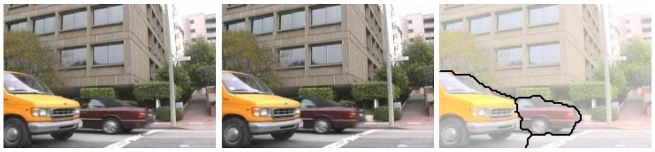

Figure 10. Motion segmentation with static camera and two moving cars.I_1(left), I_2(middle), segmentation (right).

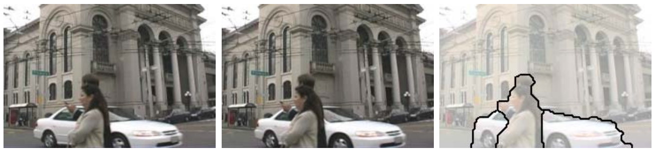

Figure 11. Motion segmentation with moving camera, car and people.I_1(left), I_2(middle), segmentation (right).

We present motion segmentation results from real city scenes with one or two foreground objects, and camera moving or static, see figures (8,9, 10,11). Because of our model's ability to detect accreted regions and use of accretion model in those places, our segmentation can follow the moving objects contour closely. Without our accretion model, the regions without motion information will have unreliable motion estimation and segmentation. We can see the difference between the robust motion estimation without our accretion model, with our accretion model but without segmentation, and with segmentation and accretion model in Figure 6. As one can see, there is still a small error in the motion estimation at places around the boundary of the person where the background has no texture (the car behind the woman). This is because of using our histogram image model for whole object, instead of a true image segmentation into intensity regions.

6 Comparison with Graph Cuts and Belief Propagation

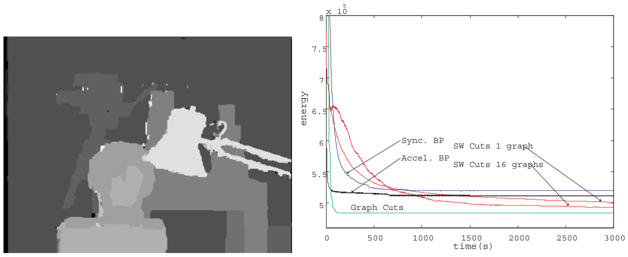

Figure 12. Stereo disparity estimation result using SW cuts and convergence plot, $K = 20$.

We cannot directly compare our algorithm with Belief Propagation or Graph Cuts because they cannot handle our energy function. Instead, we compared on the classical stereo problem[12, 10]. For the Tsukuba sequence, we compared the algorithm used in this paper with a similar SW cuts algorithm working on 16 graphs (one for each possible disparity). We found that our 16-graph algorithm can get to within less than 1% of the final energy of the Graph Cuts Algorithm. It reaches the energy level of the Belief Propagation in 15 minutes. For the comparison, we used Scharstein's[10] package and Tappen's[12] extension to BP.

7 Conclusion

This paper introduces a new approach to motion estimation based on the Swendsen-Wang Cuts algorithm. We also use an image sequence model motivated by image encoding, which allows us to deal with accreted regions which are visible only in the second frame. We use a three stage algorithm to maximize the posterior distribution.

We consider this paper to provide proof of concept for our approach and, in particular, the use of the Swendsen-Wang Cuts algorithm for this problem. Our implementation made a number of simplifying assumptions about the image models and the velocities. In our current work, we make these assumptions more realistic and combine them with image segmentation.

References

[1] A. Barbu and S.C. Zhu, "Graph partition by Swendsen-Wang Cuts", *ICCV*, Nice, France, 2003.

[2] M. J. Black, A. D. Jepson,"Estimating optical flow in segmented images using variable order parametric models with local deformations",*IEEE Trans PAMI*, Vol. 18, no. 10, pp. 972–986, 1996.

[3] D. Cremers, "A Variational Framework for Image Segmentation Combining Motion Estimation and Shape Regularization", *IEEE CVPR*, 2003.

[4] D. Geiger, F. Girosi, "Parallel and Deterministic Algorithms from Mrfs: Surface Reconstruction". *IEEE Trans. PAMI*, Vol. 13. 1991

[5] D. Geiger, B. Ladendorf, A.L. Yuille, "Occlusions and Binocular Stereo". *ECCV*, 1992

[6] S. Geman, D. Geman. "Stochastic relaxation, Gibbs distribution and the Bayesian restoration of images". *IEEE Trans PAMI*,Vol. 6 No.6,pp 721–741, 1984

[7] B.K.P. Horn, *Robot Vision*. MIT Press, 1986.

[8] L. Gaucher and G. Medioni "Accurate Motion Flow with Discontinuities". *ICCV*, pp 695-702, 1999.

[9] S. Osher, J.A. Sethian,"Fronts propagating with curvature dependent speed". *J. Computational Physics*, Vol. 79, pp. 12-49, 1988.

[10] D. Scharstein and R. Szeliski. "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms", *IJCV*, 2002

[11] S. Soatto, G. Doretto, Y. N. Wu, "Dynamic Texture", *ICCV*, 2001

[12] M. F. Tappen, W. T. Freeman, "Comparison of Graph Cuts with Belief Propagation for Stereo, using Identical MRF Parameters" *ICCV* 2003

[13] P.H.S. Torr,"Geometric motion segmentation and model selection", *Philosophical Trans of the Royal Society A*,pp. 1321–1340, 1998.

[14] Z.W. Tu, S. C. Zhu, "Image segmentation by data-driven Markov chain Monte Carlo", *IEEE Trans. on PAMI*, Vol 24, no. 5, 2002.

[15] Y. Z. Wang, S.C. Zhu, "Modeling Textured Motion : Particle, Wave and Sketch", *ICCV* 2003.

[16] Y. Weiss,"Smoothness in Layers: Motion segmentation using nonparametric mixture estimation", *Proc. of IEEE CVPR* pp. 520-527, 1997.

[17] U. Wolff, "Collective Monte Carlo updating for spin systems", *Phys. Rev. Lett.*, vol. 62, no. 4, pp. 361-364, 1989.

[18] A.L. Yuille, N.M. Grzywacz,"A mathematical analysis of the motion coherence theory" *Int. J. Computer Vision* **3**, pp. 155–175, 1989.

[19] S.C. Zhu, A.L Yuille,"Region Competition",*IEEE Trans. PAMI*, Vol. 18, no.9 pp. 884-900, 1996.