On the relationship between image and motion segmentation

Adrian Barbu[1] and Song Chun Zhu[2]

[1] UCLA, Computer Science Department,
Los Angeles, CA 90095
abarbu@ucla.edu
http://www.cs.ucla.edu/∼abarbu/Research/
[2] UCLA, Statistics Department,
Los angeles, CA 90095
sczhu@stat.ucla.edu
http://www.stat.ucla.edu/∼sczhu/

Abstract. In this paper we present a generative model for image sequences, which can be applied to motion segmentation and tracking, and to image sequence compression. The model consists of regions of relatively constant color that have a motion model explaining their motion in time. At each frame, the model can allow accretion and deletion of pixels. We also present an algorithm for maximizing the posterior probability of the image sequence model, based on the recently introduced Swendsen-Wang Cuts algorithm. We show how one can use multiple cues and model switching in a reversible manner to make better bottom-up proposals. The algorithm works on the 3d spatiotemporal pixel volume to reassign entire trajectories of constant color in very few steps, while maintaining detailed balance.

1 Introduction

Motion segmentation and tracking can be performed together in a unified way as it was clearly showed in [3]. However, there is still a lot to be done in this direction. Better motion and shape priors should be studied, the 3d reconstruction of the moving objects and their motion should be used wherever possible as better motion models. Flexible motion models (2d or 3d) should also be studied.

Another important and very interesting open question is the relationship between the image segmentation and motion segmentation. When we see a moving scene, we perceive it as some "segmentation". But this is neither image segmentation based on image intensity, neither motion segmentation based on motion. Somehow, our brain is capable of combining all the existing information to give a better segmentation than either the intensity based or the motion based segmentation. How does the brain combine the two different segmentations cues into a single segmentation? In this paper, we try to give an answer to this question using generative models in a probabilistic Bayesian framework.

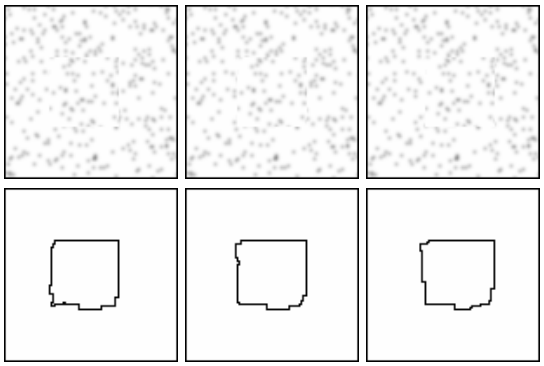

Fig. 1. A rectangle of uniform texture moving in a similar background is properly modeled and segmented by our method.

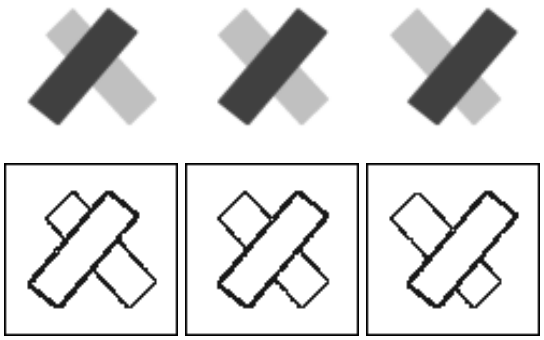

Fig. 2. Two slanted rectangles of constant intensity moving horizontally create false motion cues at their intersection points. Also, there is no motion information inside the rectangles. But they are properly modeled and segmented by our method.

There are some classical examples that should be explained by a system combining image and motion segmentation. For example, if one has an image

of uniform texture with an object of the exactly same texture moving in the center of the image, the system should be able to correctly segment the object out based on motion. Here the motion segmentation provides the clues for the desired "segmentation". On the other hand, in an image sequence with objects of constant intensity, the motion cues are very sparse, namely only at the boundary of the objects and in direction perpendicular to the boundary. Some corners could even have motion not compatible with any of the moving objects of the scene (see Fig. 2). The desired "segmentation" has to integrate these sparse and sometimes misleading motion cues and combine them with intensity information to obtain a good segmentation, based on the prior information that motion boundaries often occur at intensity boundaries. Here the image segmentation provides most of the cues for the desired "segmentation".

Our generative model can easily handle these examples, as one can see in Figures 1 and 2. Moreover, it can also handle extreme cases with large motion as shown in Figure 3. In a square of size 100x100 filled with random pixels, a 10x10 square of random pixels is moving, on three frames, with the speed of 30 pixels per frame. By having the right motion hypotheses, our framework can segment and track the little square.

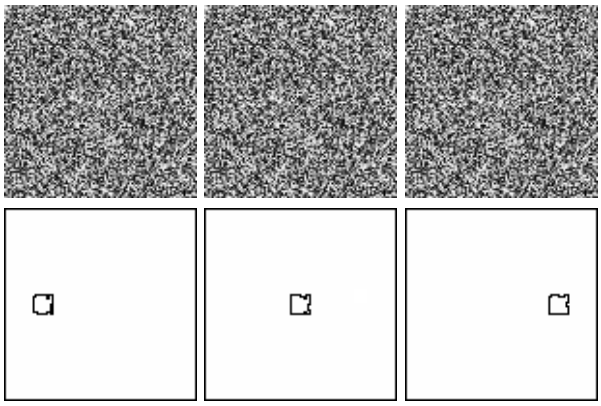

Fig. 3. A 10x10 square of random pixels moving in a similar background at a speed of 30 pixels per frame is properly modeled and segmented by our method.

It is our belief that image segmentation and motion segmentation should be performed together, in a hierarchical fashion. A reason for this is the fact that intensity regions can already have their motion inferred, so the motion segmentation of an object with many intensity regions can be thought as just combining the intensity regions based on their common motions.

In other words, the first level is a special kind of image segmentation into regions of relatively constant intensity moving by some motion model, then these regions are combined together, on a second level of representation and computation, based solely on their motion models to give the motion segmentation. See Figures 11 and 12 for motion segmentation results based on simple clustering of the motion models of the intensity regions.

However, there is more to the second level (of motion segmentation) than just clustering the motions of the intensity regions. The higher level model can be a

3d model (rigid 3d motion for example) while the motions of the different image regions would be usually much simpler (planar motion models). Together, the intensity regions can provide the necessary information to allow the computation of a global 3d model, even though each of them individually doesn't have that information. Also, the higher level will have priors for motion segmentation on top of the image segmentation priors of the lower level.

The purpose of this paper is to present the model and a computational framework for the first level, that of segmenting and tracking of regions of relatively constant intensity and motion. In a subsequent paper we will show how to integrate this level with the second level of motion segmentation tracked in time.

Our framework is based on a probabilistic Bayesian generative model that explains all the pixels in the motion sequence. Usually (see [2, 3, 7, 9]), probabilistic models for motion are put on the spatiotemporal gradient of the image, without explaining the original image sequence. This is based on the assumption that the scene is lambertian without any abrupt changes in intensity or motion. This highly restricts the applicability of the gradient based methods.

In our framework we use the gradient based motion models as one kind of bottom-up proposals. Since they do not always give the correct motion model, our algorithm will take them, and accept or reject them based on the Metropolis-Hastings algorithm and our posterior probability.

An important aspect of our framework is that we don't try to track pixels, we try to track small regions of relatively constant color. This allows the usage of motion models beyond the classical translational models.

This work was sponsored by the NSF SGER grant IIS-0240148.

2 The image sequence model

Let $I = (I_0, ..., I_k)$ be the observed image sequence. We want to find a partition R of I into an unknown number n of subsets R_i of relatively constant color or texture, which we call region trajectories, or just regions. Each R_i represents the trajectory of a patch of constant color or uniform texture tracked in time by a rigid motion model (translation, affine, projective or homography). We denote by R_i^t the region R_i at frame t.

The model of R_i consists of an image model and a motion model. The image model models all pixels that are accreted (their projection through the motion model in the previous frame falls outside the region R_i). In our framework, we chose the image model to be a Gaussian model $\mathcal{N}(\mu_i, \sigma_i)$. The motion model T_i^t is a transformation that describes how the region changes from frame t to frame $t+1$. At each frame it is a translation, affine, projective or homography transformation. Later on we could consider even flexible models. Since the reconstruction from the previous frame through the motion model is not perfect, we have a noise model η_i modeled by a Gaussian $\mathcal{N}(\nu_i, \tau_i)$. Thus

$$I_{t+1}(T_i^t(x)) - I_t(x) \sim \mathcal{N}(\nu_i, \tau_i),\ \forall x \in R_i^t\ s.t.\ T_i^t(x) \in R_i^{t+1} \tag{1}$$

Let $T_i = \{T_i^t\}$ be all the transformations for region R_i and $T = \{T_1, ..., T_n\}$ be the transformations of all the regions. Similarly let $\theta = \{\theta_1, ..., \theta_n\}$ be the image

models and $\eta = \{\eta_1, ..., \eta_n\}$ be the noise models for all the regions. For now, we assume the regions are independent. In a subsequent paper, we will group them into moving objects, based on their motion similarity.

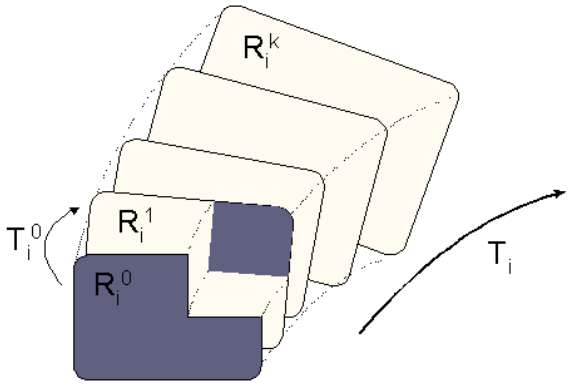

Fig. 4. A region trajectory R_i has an image model that explains the *accreted* pixels (shown in dark), whose projection in the previous frame is not inside R_i, and a motion model for the rest of the pixels (shown in light color).

Thus the hidden variables are:

$$W = \{n, (R_i, \theta_i, T_i, \eta_i), i \in \{1, .., n\}\} \tag{2}$$

We work in a Bayesian framework with prior and likelihood:

$$P(W|I) \propto P(I|W)P(W) = P(I|R, \theta, T, \eta)p(R, \theta, T, \eta)$$

We assume a Markov dependence of each frame on the previous one:

$$P(I|R, \theta, T, \eta) = \prod_t P(I_{t+1}|I_t, R, \theta, T, \eta)$$

The pixels $I^{t+1}(R_i)$ of region R_i at frame $t+1$ are reconstructed from I_t, R, θ, T, η as follows: All pixels $x \in R_i^{t+1}$ whose back-projection by T_i^t falls inside R_i^t are modeled by the noise model η_i. Let M_i^{t+1} be these pixels, namely:

$$M_i^{t+1} = \{x \in R_i^{t+1}, x = T_i^t(y), y \in R_i^t\} \tag{3}$$

The pixels of $R_i^{t+1} - M_i^{t+1}$ are modeled by the image model θ_i. Thus:

$$\begin{aligned} P(I_{t+1}|I_t, R, \theta, T, \eta) &= \textstyle\prod_i \prod_{x=T_i^t(y) \in M_i^{t+1}} P_{\eta_i}(I(x) - I(y)) \prod_i \prod_{x \in R_i^{t+1} - M_i^{t+1}} P_{\theta_i}(I(x)) \\ &\propto \textstyle\prod_i e^{-\frac{1}{2\tau_i^2} \sum_{x=T_i^t(y) \in M_i^{t+1}} (I(x) - I(y) - \nu_i)^2} \prod_i e^{-\frac{1}{2\sigma_i^2} \sum_{x \in R_i^{t+1} - M_i^{t+1}} (I(x) - \mu_i)^2} \end{aligned} \tag{4}$$

The prior is simplified as $P(R, \theta, T, \eta) = P(R|T)P(T)P(\theta)P(\eta)$. For now we take $P(T) = \prod_i \prod_t \delta(T_i^t - T_i^{t-1})$. We also assume $P(\theta)$ and $P(\eta)$ to be uniform.

The factor $P(R|T)$ represents that the regions should be big, should have a smooth boundary, and be consistent with the transformations T. We take:

$$P(R|T) \propto \prod_i \exp[-\alpha \text{Vol}(R_i)^{0.9} - \beta \text{Area}(\partial R_i) - \gamma(n_i^{in} + n_i^{out})] \tag{5}$$

where n_i^{in} represents the number of pixels of R_i whose back-projection to the previous frame is not in R_i, and n_i^{out} represents the number of pixels of R_i whose projection to the next frame is not in R_i.

3 Space-time segmentation by Swendsen-Wang Cuts

3.1 The multi-cue Swendsen-Wang Cuts

Sometimes there are many cues which provide bottom-up information for the graph partitioning. How can we combine these cues while maintaining detailed balance?

Great help for answering this question comes from the following

Theorem 1 *Let $q_1, ..., q_n$ be Markov moves with transition kernels $K_1, ..., K_n$, such that all q_i observe detailed balance with respect to the same probability p. Let $\alpha_1, ..., \alpha_n \geq 0$ be such that $\alpha_1 + ... + \alpha_n = 1$. Then the Markov move q that at each step randomly selects an $i \in \{1, ..., n\}$ with probability α_i and executes q_i has transition kernel:*

$$K = \sum_{i=1}^{n} \alpha_i K_i \tag{6}$$

and also satisfies the detailed balance equation for p.

From this theorem, the answer to our question comes easily:

Corollary 2 *Let $SW_1, ..., SW_n$ be a number of Swendsen-Wang Cuts algorithms working on the same nodes V and same posterior probability P, with adjacency graphs $G_1, ..., G_n$. Let $\alpha_1, ..., \alpha_n \geq 0$ be fixed numbers such that $\alpha_1 + ... + \alpha_n = 1$. Then the move consisting of randomly choosing an i with probability α_i and executing SW_i is reversible and ergodic.*

We can think of each SW_i as a hypothesis that is being tested in a reversible manner.

The only restriction in using the above results is that the α_i be fixed. We can still use, if possible, bottom-up information to select good values for α_i, resulting in an efficient visiting schedule of the different SW_i as long as the schedule is fixed a priori. This way we can have some hypotheses more likely than other, so they are tested more often. For each hypothesis, the algorithm will be efficient at the places where that hypothesis is valid. By combining a good set of hypotheses, the algorithm will be efficient everywhere.

3.2 Multi-cue SW Cuts for motion

The SW Cut algorithm is most efficient when the sampled connected components closely resemble the segments in the desired segmentation. Thus, in order to have good bottom-up information, the proposals should be long region trajectories. For that, the edges of the SW graph must be very informative. It is not enough

just to have them on a 3d lattice with weights based on pixel similarity, we have to bring in the motion and intensity hypotheses.

A motion hypothesis is a k-tuple $\boldsymbol{m} = (T^1, ..., T^k)$ for the hypothesized affine transformation at each frame. Usually, we use only constant velocity hypotheses $(v, v, ..., v)$. In the future, we will use more complex hypotheses which are based on more complex motion (rotation, zoom, etc).

An intensity hypothesis is an intensity model G (we use gaussian, but we can also use histograms of filter responses for texture, or a histogram of intensity values for the clutter model).

Together they compose a hypothesis $h = (G, \boldsymbol{m}) = (G, T^1, ..., T^k)$.

Fig. 5. The 2d space of x-motion (x-axis) and intensity (y-axis) of feature points from Figures 1,2,9,10 and 12 respectively. Darker means more instances of features.

To obtain motion hypotheses, we extract some feature points (corners) and we perform clustering in the joint space of intensity-motion hypotheses of the corners. We obtain a few (less than 20) clusters. For each such cluster we will have a SW graph. For illustration, we present in Figure 5 the x-motion (x-axis) and intensity (y-axis) obtained from the feature points of Figures 1,2,9,10 and 12 respectively.

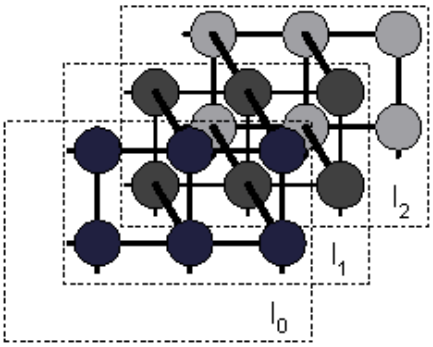

Fig. 6. For each hypothesis $(G, T^1, ..., T^k)$, the SW graph is a lattice at each frame (solid thin lines), and between frames $t-1$ and t there are edges only in the direction $\boldsymbol{x} \to T^t(\boldsymbol{x})$ (thick lines). Shown is the SW graph for hypothesis $(G, v, ..., v), v = (-1, 0)$.

For a hypothesis $(G, T^1, ..., T^k)$, the SW graph is a lattice at each frame, and from frame $t-1$ to t consists only of edges $\boldsymbol{x} \to T_t(\boldsymbol{x})$. This way the edges for each graph are in a reasonable number, while being able to cover the most likely motions.

For hypothesis $h = (G, T^1, ..., T^k)$ with $G = (\mu, \sigma^2)$, and x at frame t, define

$$g_h(x) = \frac{(I_t(x) - \mu)^2}{2\sigma^2}, d_h(x) = |I_t(x) - I_{t+1}(T^{t+1}(x))|, d^h(x) = |I_t(x) - I_{t-1}((T^t)^{-1}(x))| \tag{7}$$

$$q_h(x) = \begin{cases} d_h(x) & \text{if } x \in I_0 \\ d^h(x) & \text{if } x \in I_k \\ (d_h(x) + d^h(x))/2 & \text{else} \end{cases} \tag{8}$$

Then the edge weights for hypothesis $h = (G, T^1, ..., T^k)$ are:

$$q(x, y) = e^{-0.1(g_h(x)+g_h(y)+q_h(x)+q_h(y))} \tag{9}$$

To get connected components, we use the Wolff variant, which grows a connected component from a seed. First we sample a hypothesis $h = (G, \boldsymbol{m})$ proportional to the sum of the saliencies $S_h(x)$ of all pixels x. Then the seed s is sampled from a "cry map" C_h of how well each pixel fits its assigned image model and motion model. Small value $C_h(x)$ means x fits well its assigned model or is not salient to this h. We take $C_h(x) = \alpha S_h(x)(g_m + q_m(x))$, with g_m, q_m from (7),(8), m being the current model of x and α is chosen so that $\sum_x C_h(x) = 1$. Then, by sampling the seed s from the pmf C_h, we usually obtain a pixel that is unhappy with its current model and salient to the current h. Then we use the SW graph G_h corresponding to h to grow the seed s to a component C, and flip its label. It is easy to check that

Theorem 3 *(SW Cuts with "cry map"). Consider a candidate component C selected by* SWC *using a "cry map" C_h^A (for state A), as described above. Let $q_c(C|A) = \sum_{x \in C} C_h^A(x)$. If the proposed move to reassign C from G_l to $G_{l'}$ is accepted with probability*

$$\alpha(A \to B) = \min(1, \frac{q_c(C|B)}{q_c(C|A)} \frac{\prod_{e \in C(C, V_{l'} - C)} (1 - q_e)}{\prod_{e \in C(C, V_l - C)} (1 - q_e)} \frac{q(l|C, B, G_h)}{q(l'|C, A, G_h)} \frac{p(B|I)}{p(A|I)}) \tag{10}$$

then the Markov chain is reversible and ergodic.

Thus basically we have an SW move for each hypothesis h, which is reversible. Then the overall move is also reversible by Corollary 2.

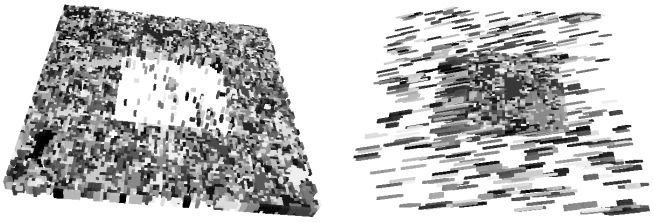

Fig. 7. For an image sequence like Fig. 1 but with more texture, we present samples from the graph in the direction of motion hypotheses $(0, 0)$ and $(4, 0)$ respectively. We see that for the right hypothesis, the SW reassigns long fibers.

The two extreme cases presented in the introduction, can explain very well what happens. In the case of the rectangle of same texture as the background, the chosen component C_{seed} are relatively small, but the motion information is very peaked, so the correct motion is sampled most of the time. Thus the components that are flipped are long "fibers", relatively thin, as shown in Figure 7.

In the case of the two slanted rectangles of uniform color, the lattice edges at each time frame have big weight, so the chosen component C_{seed} is big, and even though the motion information might not be precise, the sampled component C will be big in both space and time, as shown in Figure 8. In this case, the problem is more of a "blob" tracking problem.

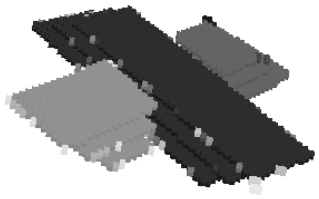

Fig. 8. For the image in Fig. 2, we present samples from the graph in the direction of hypothesis of motion $(0, 0)$. For clarity, the component of the background has been removed. Even though the motion hypothesis is not correct for the moving rectangles, the samples are big and meaningful in both space and time.

For real world problems, the components sampled will be between the extreme cases, making use of the available intensity or motion information.

3.3 Model switching in Swendsen-Wang Cuts

It is clear that in our approach, the motion models are very important. If, at each step, the proposed image models are inappropriate, the move will be rejected and the algorithm will slow down drastically. This is why we have to combine the model switching with the Swendsen-Wang Cuts reassignment in a single step:

Theorem 4 *(SW Cuts with model switching). In the notations of [1], consider a candidate component V_0 selected by* SWC*. Let $q_m(\mu_i|V_i)$ be a proposal probability from which the model μ_i of a subgraph V_i of the partition is chosen by sampling. If the proposed move to reassign V_0 from G_l to $G_{l'}$, and then change the model of G_l from μ_l^A to μ_l^B and the model of $G_{l'}$ from $\mu_{l'}^A$ to $\mu_{l'}^B$ is accepted with probability*

$$\alpha(A \to B) = \min(1, \frac{q_m(\mu_l^A|V_l \cup V_0)q_m(\mu_{l'}^A|V_{l'} - V_0)}{q_m(\mu_l^B|V_l - V_0)q_m(\mu_{l'}^B|V_{l'} \cup V_0)} \frac{\prod_{e \in C(V_0, V_{l'} - V_0)}(1 - q_e)}{\prod_{e \in C(V_0, V_l - V_0)}(1 - q_e)} \frac{q(l|V_0, B, G_o)p(B|I)}{q(l'|V_0, A, G_o)p(A|I)}) \tag{11}$$

then the Markov chain is reversible and ergodic.

For the model proposal probabilities, we will have three kind of proposals $q_1(T_i|I), q_2(T_i|I), q_3(T_i|I)$ used with frequency $0.98, 0.01, 0.01$ respectively. Each proposal is one type of reversible SW move and since we are using them with constant frequency, the overall move is still reversible. First proposal, is $q_1(T_i|I) = \delta(T_i)$ enforces no change in model. Second proposal is $q_2(T_i|I) = c$ is uniform in a discrete window of possible motions. This ensures that even if all other proposals fail, one could still obtain the right model after enough time. The third proposal is the constant velocity affine model from [3]. The usefulness of the Cremers model (working on the spatiotemporal gradient $\nabla_3 I = (I_x, I_y, I_t)$)

for our purpose is that it can be computed incrementally, so we always have it available for all regions, without much overhead. It is,

$$q_3(T_i|I) =\propto \prod_{(x,y)\in R_i} \exp(-\frac{p_i^T M(x,y)^T \nabla_3 I}{|p_i|^2 |M(x,y)^T \nabla_3 I|^2}) \tag{12}$$

where

$T_i^t = \begin{pmatrix} a_i & b_i & c_i \\ d_i & e_i & f_i \\ 0 & 0 & 1 \end{pmatrix}, p_i = (a_i, b_i, c_i, d_i, e_i, f_i, 1)^T, M(x,y) = \begin{pmatrix} x & y & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & y & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$

and $\nabla_3 I = (I_x, I_y, I_t)$ is the spatiotemporal gradient of the image sequence I.

This corresponds to a velocity field at frame t: $v_i(x,t) = T_i^t \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = M(x,y)p_i$.

Since we discretized our model parameter space, we can easily compute the proposal probability at each bin and then sample from the discretized proposal probability $q(p_i)$, without having to compute the eigenvalues and eigenvectors.

4 Experiments

Fig. 9. A moving person in front of a moving background.

We have already presented some experiments on synthetic sequences in Figures 1,2,3. Our experiments are performed on 3-5 frames of grayscale image sequences.

Other results are presented in Figures 9,10, 11 and 12. We see that the framework is capable of producing new image regions when new objects are visible in the image, like the small car in Figure 10. In figures 11 and 12, we also show some motion segmentation results obtained by simple clustering on the motion velocities of the regions. The motion segmentations can be improved by having priors on the motion regions, and not assuming the regions to be independent. Animated demos for all examples can be found on the web at http://www.cs.ucla.edu/∼abarbu/Research/SWMotion3d/

Fig. 10. A car appears from behind another car in a static background.

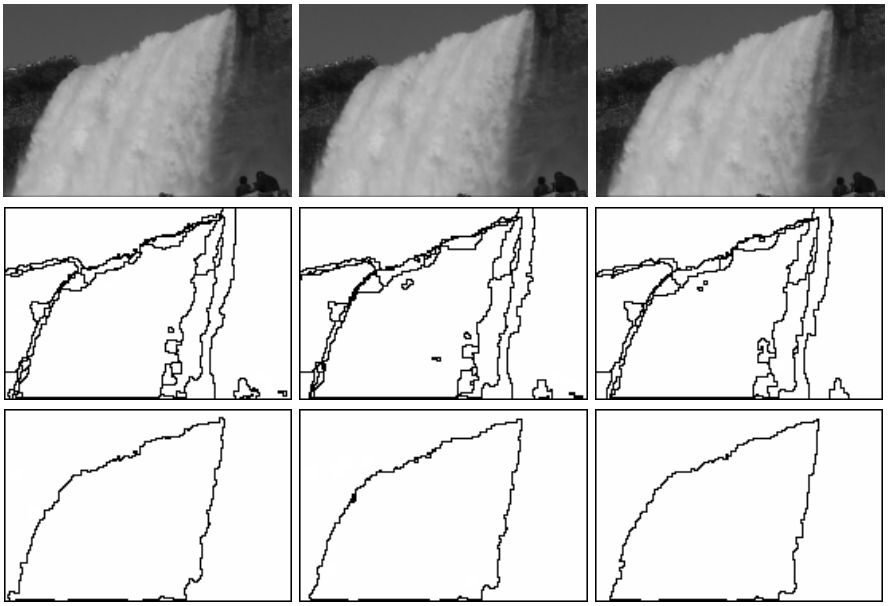

Fig. 11. Waterfall with a moving background. Input sequence (first row),image segmentation (second row) and motion segmentation obtained by clustering of the velocities of the regions (third row).

5 Limitations and future work

Currently, the computation time is quite high, between 2 and 10 minutes per frame for a 100x100 image on a PC. One reason is that the graph edges are still not informative enough. For example, if one has two neighboring regions of very similar intensity and no texture, moving differently, the edges will be strong and the algorithm will not be able to separate the regions quickly. We will examine how to use belief propagation and top-down information to change the edges of

the graph to be more informative. Also, better shape priors which can be quickly computed are needed to be studied.

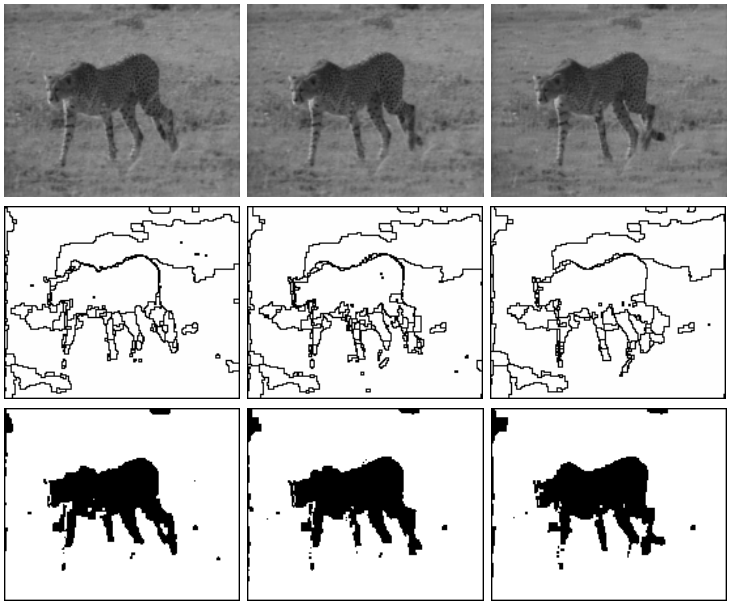

Fig. 12. A walking cheetah on a moving background. Input sequence (first row),image segmentation (second row) and motion segmentation obtained by clustering of the velocities of the regions (third row).

References

1. Barbu A., Zhu S.C.: Graph Partition by Swendsen-Wang Cuts IEEE International Conference in Computer Vision (2003)
2. Black M. J., Jepson A. D.: Estimating optical flow in segmented images using variable order parametric models with local deformations. IEEE Trans PAMI **18** no. 10 (1996) 972–986
3. Cremers D., Soatto S.: Variational Space-Time Motion Segmentation, ICCV, (2003)
4. Horn, B.K.P.: Robot Vision. MIT Press. Cambridge, Ma. (1986)
5. Gaucher, L. and Medioni, G. Accurate Motion Flow with Discontinuities. ICCV, pp 695-702. 1999.
6. Torr P.H.S.: Geometric motion segmentation and model selection. Philosophical Trans of the Royal Society A, (1998) 1321–1340
7. Weiss Y.: Smoothness in Layers: Motion segmentation using nonparametric mixture estimation. Proc. of IEEE CVPR (1997) 520-527
8. Wolff U., "Collective Monte Carlo updating for spin systems", *Phys. Rev. Lett.*, vol. 62, no. 4, pp. 361-364, 1989.
9. Yuille A.L., Grzywacz N.M.: A mathematical analysis of the motion coherence theory. Int. J. Computer Vision **3** (1989) 155–175