A unified mixture framework for motion segmentation: incorporating spatial coherence and estimating the number of models

Yair Weiss and Edward H. Adelson
Dept. of Brain and Cognitive Sciences
MIT E10-120, Cambridge, MA 02139, USA
{yweiss,adelson}@psyche.mit.edu

Abstract

Describing a video sequence in terms of a small number of coherently moving segments is useful for tasks ranging from video compression to event perception. A promising approach is to view the motion segmentation problem in a mixture estimation framework. However, existing formulations generally use only the motion data and thus fail to make use of static cues when segmenting the sequence. Furthermore, the number of models is either specified in advance or estimated outside the mixture model framework. In this work we address both of these issues. We show how to add spatial constraints to the mixture formulations and present a variant of the EM algorithm that makes use of both the form and the motion constraints. Moreover this algorithm estimates the number of segments given knowledge about the level of model failure expected in the sequence. The algorithm's performance is illustrated on synthetic and real image sequences.[1]

1 Motivation

Significant progress in scene analysis has been achieved by systems that segment primarily based on common motion (e.g. [14, 3]). Yet automatic segmentation of arbitrary sequences remains difficult for computer vision systems. In this paper we argue that this difficulty is partially a result of the exclusive reliance on motion data which in many cases can be segmented equally well by multiple interpretations. Let us consider two such examples.

The first example is the two-bars sequence depicted in figure 1. Two intersecting bars are moving, one to the left and one to the right.

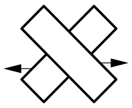


Figure 1: The two-Bars sequence. Two intersecting bars are moving, one to the left and one to the right.

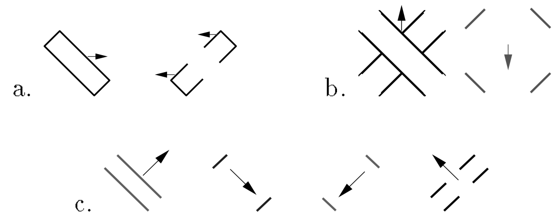


Figure 2: **a.** The desired description of the two-bars sequence. There are two motion groups with support maps as shown. **b.** An incorrect description with two groups which fits the motion data equally well. **c.** An incorrect description with four groups which fits the motion data equally well.

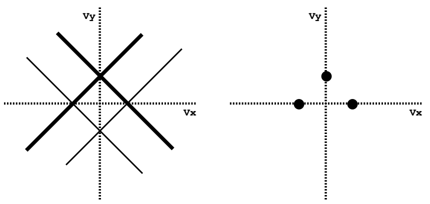
Following [3, 8, 7, 14, 1] we seek to describe the scene in terms of a small number of coherent motion groups. Figure 2a shows the desired description for this sequence: two coherent motion groups, leftward and rightward, with support maps as shown. However, the instantaneous motion data can also be accommodated by the interpretations shown in figure 2b-c. The interpretation depicted in figure 2b describes the scene using two coherent motion groups, upward and downward, while the interpretation shown in figure 2c uses four coherent motion groups in the four principal diagonal directions.

Let us be more precise regarding what it means to explain the motion data. One way to do this is to look at the constraints imposed by the normal flow in velocity space. This is shown in figure 3. There are four constraint lines, and their thickness corresponds to the number of votes. Obviously, all three interpretations satisfy the constraints equally well. In fact the motion that satisfies the most constraints, i.e. the "dominant" motion of this scene, is the incorrect upward one.

A second example of the ambiguity in segmentation based on motion alone is the disc sequence shown in figure 4a. A textured disc is translating in front of a stationary textured background.

In an ideal noiseless world, the optic flow measurements derived for this sequence would be as shown in figure 4b. However, the flow shown in figure 4c is more

[1] a shorter version of this paper appears in CVPR 96

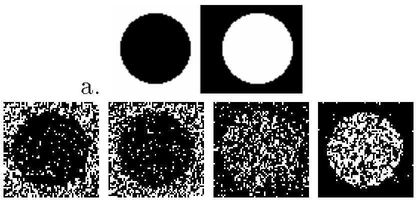


a b

Figure 3: **a.** The normal constraint lines for the two-bars sequence. The thickness of the line corresponds to number of votes. Note that all descriptions shown in figure 2 satisfy the constraints equally well and that the dominant motion of the scene is the spurious upwards one. **b.** the corner constraints for the two-bars sequence. The spurious upward motion is still supported.

a b c

Figure 4: **a.** The disc sequence. A textured disc is translating in front of a stationary background with identical texture. **b.** Noise-free optic flow shown (x component of motion indicated by gray level) **c.** Noisy optic flow (x component only).



a.
b.

Figure 5: **a.** The correct segmentation of the disc scene. Two coherent groups are shown with white pixels indicating membership in a group. **b.** An incorrect segmentation with four groups that explains the motion data in figure 4c better than the two group description.

realistic: aliasing, reflectance changes and measurement noise will cause the measured flow to be noisy. The correct description, consisting of two coherently moving motion groups is shown on the top of figure 5, but the description shown on the bottom of figure 5 consisting of four coherently moving groups actually explains the motion data better. In fact a description where all pixels are moving independently would explain the motion data best.

Why should the first description be favored? Part of the answer lies in the fact that it explains the data using fewer models. But obviously there exist scenes where more than two models are needed. A second reason to prefer the first description is that the resulting segmentation is spatially coherent, while in the second one the segmentation is fragmented and we know that such support maps are less likely to occur.

Thus these are the problems we want to address: we want to use spatial coherence to constrain the possible motion models and we want to automatically estimate the number of models. As the sequences discussed above show, these problems are closely related. In this paper, we show that a new formulation of the standard mixture model provides a natural unified framework in which to address both of these problems.

2 Mixture models and the EM algorithm

Mixture estimation refers to the estimation of parameters given data that was generated by multiple processes. As in all MLE techniques, the goal is to estimate the parameters of the models (which we will denote by θ_k) which maximize the likelihood of the observed data (denoted by $\{O(r)\}_r$. The Expectation-Maximization (EM) algorithm [4] treats mixture estimation as a special case of estimation with incomplete data. The underlying model is that the complete data includes not only $O(r)$ (the "visible data"), but also the "hidden data", labels $L(r)$ specifying which process generated the data (L(r) is a binary vector such that $L_k(r) = 1$ iff process k generated the data at r). The assumption is that if $L(r)$ were known, the estimation of θ_k would be simple.

The EM algorithm calls for replacing $L(r)$ at each iteration with its conditional expectation (this is the expectation, or E step) based on the current parameter estimates. We will denote the conditional expectation of $L_k(r)$ by $g_k(r)$. Although $L_k(r)$ is binary valued, $g_k(r)$ takes on continuous values between zero and one and $g_k(r)$ sum to one for fixed r. The maximization, or M step uses current values of $g_k(r)$ to maximize likelihood of the parameters θ_k (since it treats $L(r)$ as known, this step is assumed to be simple), and the algorithm is iterated until convergence. Dempster et al. [4] have shown that each iteration is guaranteed to increase the likelihood of the estimates of θ_k.

In the case of motion analysis θ_k would be a parametric description of the motion predicted by model k and $\{O(r)\}$ would be spatiotemporal measurements made at location r. In this case $g_k(r)$ would be a kind of "soft" segmentation of the image. Existing EM motion algorithms [8, 1] update $g_k(r)$ by calculating at every pixel a deviation measure $D_k(r)$ that measures the residual between the observed measurements $O(r)$ and the predicted measurements assuming the motion of model k. They can be characterized as assigning each pixel to the model which minimizes the residual (E step), and then updating model parameters based on these assignments (M step). The "softness" of the assignment $g_k(r)$ is determined by how peaked the assumed distribution of $D_k(r)$ is: if $D_k(r)$ is assumed to be narrowly concentrated around zero residual (e.g. a Gaussian with small σ) the conditional probability $g_k(r)$ vanishes for all models except the one with smallest residual and each pixel is effectively assigned to exactly one process. If $D_k(r)$ is assumed to be broadly distributed around zero residual (e.g. a Gaussian with large σ) a pixel can be assigned to multiple processes. In this case the assignments, $g_k(r)$ will be a set of weights summing to one for every pixel.

2.1 Spatial constraints in mixture models

As noted above, existing EM algorithms [8, 1] for motion segmentation assign pixels to models based on local residual. In effect this assumes a type of independence in the hidden variables $L(r)$, namely that knowing the membership of a particular location yields no information on the membership of all other locations in the image. In image formation, this is rarely the case: e.g. neighboring points with the same intensity are likely to be from the same object. We have developed alternate E steps which assume spatial dependence of $L(r)$ and are useful for motion segmentation.

Segmented images. Suppose we can segment the image based on static intensity cues into multiple fragments of similar intensities. (cf. [2]) We still don't know the correct motion segmentation of the image (i.e. many of the fragments may be moving together) but we can assume that if one pixel was generated by a certain process, so were all other pixels in the same fragment. It is easy to show that under this assumption the E step reduces to assigning pixels to models based on the summed deviation from model prediction for all pixels in the same fragment. Again the assignment can be soft or hard depending on the assumed probability distribution of the deviations.

MRF prior on L. A weaker form of prior knowledge on L is to assume a Markov Random Field (MRF) distribution, i.e. that nearby pixels are likely to belong to the same model. As is well known from applying MRF approaches to vision problems, an exact calculation of the probabilities of one variable in a MRF given the others is computationally intensive (see e.g. [6]). Since in the EM algorithm, this calculation must be carried out at every iteration one can understand the reluctance of some researchers to use this prior in a segmentation algorithm. We have found that an incremental algorithm based on the mean field approximation (e.g. [16]) calculates g_k results in reasonable computation time and can be shown to converge to a local maximum of an approximate likelihood function.

Our algorithm is based on some recent results which relate the EM algorithm to statistical physics [9, 15]. The essence of this connection is that solving the mean field equations for a MRF is equivalent to minimizing with respect to g the following "energy-entropy" tradeoff:

$$\begin{aligned} J(g) &= \sum_{k,r} g_k(r) D_k(r)/\sigma^2 - \sum_{k,r,s} w_{rs} g_k(r) g_k(s) \\ &\quad + \sum_{k,r} g_k(r) \log g_k(r) \end{aligned} \tag{1}$$

Where w_{rs} are the expected strength of links between location r and location s in the MRF formulation. In theory, one could solve the MF equations by minimizing J using local gradient updates but in practice this is computationally prohibitive. Neal and Hinton [9] have shown that to guarantee convergence of the EM algorithm it is not necessary to minimize this energy but rather it is sufficient to choose a new estimate of g such that the energy is decreased at every iteration. Our E step calls for trying a reasonable guess for a new estimate of g (obtained by blurring the residuals) and accepting it only if the free energy is decreased. If that estimate is rejected, we apply local gradient updates to the old estimate of g. These local updates are guaranteed to decrease the free energy and hence our E step always decreases the free energy.

2.2 Estimating the number of models

One of the most difficult problems in grouping is to estimate the number of groups. Indeed it might seem that this problem can not be addressed in a simple maximum likelihood framework for mixture estimation, due to the fact that one can always make the data more likely by adding more models. However this intuition breaks down if the distribution of the residuals $D_k(r)$ is assumed to be known. It has been shown [11, 13] that if $D_k(r)$ is assumed to be a Gaussian with variance σ^2 then the local maximum of a K component mixture likelihood is attained with number of *distinct* models that may be smaller than K, and that this number depends on σ.

This result makes intuitive sense. Note that σ expresses how well we expect our models to fit the data. If we expect our description to fit the data perfectly then we will need many models to explain the data. However, if we expect the models to fit imperfectly then fewer models are sufficient. To gain more intuition, let us consider a simple example.

Suppose there are 10 pixels and we try to estimate the parameters of 10 motion models. A trivial solution is to have each model explain one data point exactly. However, for σ significantly greater than zero this is *not* a local maximum of the likelihood: while every model makes its own datapoint infinitely likely, the other points are unlikely. Recall that the parameter σ controls the "softness" of the assignment. Thus if we initialize the standard EM algorithm with this trivial configuration, the behavior will depend on σ: for infinitely large σ each data point will be equally assigned among models and hence the algorithm will converge to ten identical models or one unique model. For σ close to zero, each data point will be assigned to exactly one model, and the algorithm will converge to ten distinct models. In general the number of distinct models increases as one decreases σ [11, 13]. Note that finding the number of distinct models does not require an additional clustering stage, as the algorithm converges to a solution in which redundant models have identical parameter estimates θ_k.

To summarize, two assumptions are needed to characterize the statistical model of the scene. First, the conditional probability of assigning a pixel to a model given the assignment of its neighbors (this determines w_{rs} in equation 1). Second, the probability of observing a residual $D_k(r)$ given that pixel r is indeed moving with motion of model k (this determines σ in equation 1). Given these two assumptions, the EM algorithm described here will automatically determine the number of models and will segment the scene based on the motion data and the prior static constraints.

3 Experimental results

An important choice in applying EM to motion segmentation is to choose the deviation measure $D_k(r)$, i.e. a measure of how well a predicted velocity at a pixel matches the image data. We chose to approximate this by a quadratic function, i.e. to assume the distribution of the image data given a predicted velocity is Gaussian in velocity space [12]. The true distribution is of course nonparametric and complicated but we chose the Gaussian for two main reasons: it is complicated enough to capture aperture effects and it makes the M step closed form.

For small motions we used a modification of the algorithm described in [12] to derive the mean μ and covariance matrix Σ^{-1} of the deviation function in velocity space. In this algorithm the covariance matrix depends nonlinearly on the local image gradients. Our modification adds a further dependence on the local residual after alignment, so that regions of accretion and deletion near occlusion boundaries receive very high uncertainty. To generalize this approach to large motions, we first used a nonlinear multiscale optical flow algorithm to align the two images (the optical flow algorithm used is described in [14]). We took the output of the optical flow as μ and calculated Σ^{-1} by using the method of [12] on the aligned images.

In all these simulations, the parameters θ corresponded to the six parameters of affine motion and hence the M step involved solving a $6x6$ system of equations. To increase the stability of the M step, we found it necessary to introduce a prior on θ that favors lower order velocity fields.

3.1 The two-bars sequence

For the two-bars sequence we used the EM algorithm described for segmented images. The segmentation used as input is shown in figure 6a. It was derived by linking together contiguous segments which straddle the same two regions. Note that the segments on the border between the long bar and the occluded bar are not linked to those between the long bar and the background. To link those would require some knowledge about occlusion relationships, and we preferred to let the motion data determine this. The results are shown in figure 6b-c. In figure 6b we show the segmentation and in figure 6b we plot at each pixel the motion of the model to which it is most likely to belong.

a b c

Figure 6: **a.** Static segmentation for the two-bars sequence used as input to the algorithm. The bars are fragmented into six segments **b.** The motion segmentation computed by our algorithm. Two groups are found. **c.** The motions. The algorithm correctly identifies the number of models and their motions

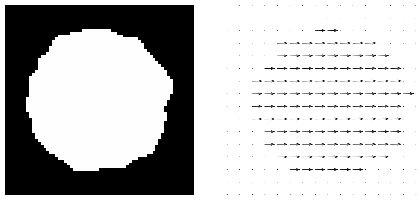


Figure 7: The results of our algorithm on the disc sequence when MRF priors are used on the segmentation. Both the number of models and their motions are estimated correctly. Left: Segmentation, Right: estimated flow.

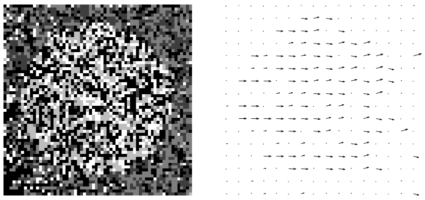
The algorithm correctly identifies both the number of models and the correct segmentation and motions. These results were stable over multiple values of σ and also when independent Gaussian noise was added to both images.

We also tried to segment this scene based solely on the motion data using EM. The results strongly depended on initial conditions and converged to some description (with varying number of models) that satisfied the constraints. The chances of getting the correct description without spatial constraints was very low since the dominant motion (i.e. the spurious upward one in figure 2b) was almost always chosen as one of the motions.

3.2 The disc sequence

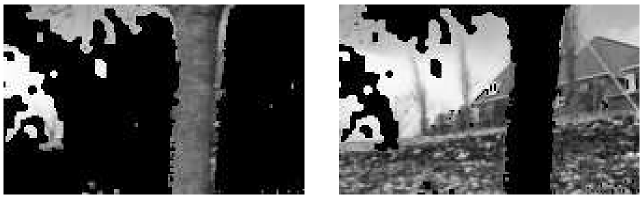
The optical flow estimated for the disc sequence is shown in figure 4a. We used the MRF prior on L_k with a neighborhood consisting of a pixel and its four nearest neighbors.

Figure 7 shows the description derived by our algorithm. Again the figure shows the at each location the motion of the model to which it is most likely to belong. Both the number of models and the correct motions of the two models were estimated. For comparison, figure 8 shows the description derived when no priors on L_k are used. In this case the algorithm "overfits" and finds four models even though it uses the same assumption about the level of noise expected



a b

Figure 8: The results of the EM algorithm without MRF priors on the disc sequence. The algorithm now overfits and finds four models. Left: Composite segmentation (each gray level corresponds to a different model), right:estimated flow.



a b

Figure 9: Segmentation achieved when a relatively large amount of model failure is expected. The algorithm finds two segments corresponding to the tree and the rest of the scene

in the description σ. Thus both priors on expected model failure and on spatial coherence influence the number of models found with the EM algorithm.

As illustrated above, the algorithm with spatial constraints is significantly more stable with respect to various choices of σ than the EM algorithm assuming independent L_k. However, for very large values of σ it will find just one motion model, while for very small values it will also overfit. Indeed if we expect a very large error in the model fitting, the "one model" description is a reasonable one.

3.3 The MPEG garden sequence

The MPEG flower garden sequence was segmented using affine motion in [14] and in [1]. The camera is translating and different areas of the scene move with different motions due to parallax. We used the same optic flow used by Wang and Adelson [14] but rather than fitting affine models to it, we use it as a center of a Gaussian distribution in velocity space with an estimated covariance.

Figures 9–11 show the calculated segmentations as σ is varied. These pictures were obtained by taking affine parameters of the different models estimated by the EM algorithm, and calculating the most likely assignment of each pixel based on the alignment error and the prior on the assignments (i.e. by iteratively minimizing J in equation 1 until convergence.)

For large values of σ the algorithm finds just one affine model. Figure 9 shows what happens as σ is lowered: two models are found, one corresponding to the tree and another to the rest of the scene. The next

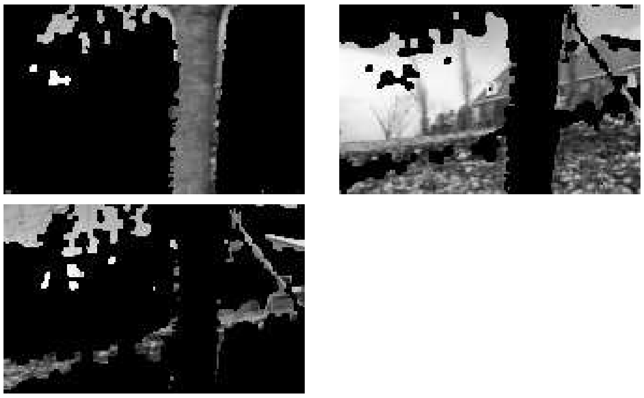


Figure 10: Segmentation achieved when σ is lowered. The algorithm finds three segments - branches which are closer to the camera than the rest of the tree are segmented from it.

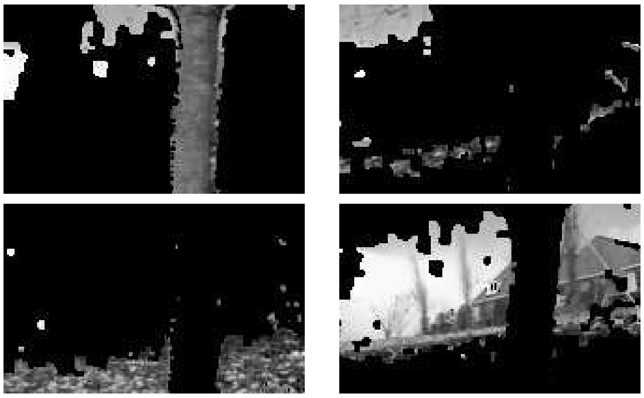


Figure 11: Segmentation achieved when σ is lowered even more. The algorithm finds four segments – the flower bed and the house are segregated.

segmentation is shown in figure 10 the branches (which are farther away from the camera) are segmented from the tree and three models are chosen. As σ is lowered even more we obtain four models: the flower bed and the house are segregated. In the next segmentation (not shown) the flower bed is split into two parts (near and far).

Note that in the three model case, a portion of the flower bed is also segmented with the branches, since they lie on the plane passing through the branches and move with a consistent affine motion. This highlights the shortcoming of the MRF priors we were using: although highly fragmented segmentations are deemed unlikely, a segmentation where a coherent chunk of the bed moves with the branches is not sufficiently penalized. We have also found that for this sequence the motion models estimated by EM with MRF priors and with $L_k(r)$ independent are not significantly different (although the final segmentations are). In current work we are extending the MRF priors in two directions: (1) nonisotropic links e.g. nearby pixels of similar colors have a stronger prior probability of being labeled identically and (2) hierarchical MRF priors which do a better job of incorporating dependencies of far away pixels.

4 Discussion

The use of static intensity constraints for motion computations was also discussed by Black and Jepson (1994) and by Etoh and Shirai (1994). In Black and Jepson's work, the image was first segmented into multiple fragments of similar intensity by a non-isotropic diffusion algorithm. Affine flow was then estimated separately for each fragment. Likewise in Etoh and Shirai's work, the image was segmented into region fragments by a procedure akin to clustering: each fragment was associated with a spatial position, a 2D translation and an intensity. The main difference between these approaches and ours is that we estimate *global* motion models, and use the intensity segmentation to constrain the possible motion models. Thus in the bars sequence, multiple fragments determined by static cues are grouped together based on their consistency with a common global motion.

Even approaches that segment primarily based on motion, often use some sort of static coherence assumptions to postprocess the segmentations [14, 10]. Although such an approach is probably sufficient for the flower garden sequence, it seems hard to believe that postprocessing any of the spurious segmentations for the two-bars sequence (figure 2) or the disc sequence(figure 5) would yield the correct segmentation. Thus in our approach the static coherence constraints are used throughout the segmentation and model selection processes.

A popular approach to estimating the number of models is to use a "minimum description length" criterion (e.g. [3, 1]). The main difference between our approach and the MDL one, is that ours requires setting of the expected level of model failure σ rather than defining the "coding length" or "stochastic complexity" of the description. We find the σ parameter to be a more natural one to set. More importantly, by estimating the number of models within the EM framework, only one set of statistical assumptions about the scene need to be specified. The same assumptions are used in the E step and in the determination of number of models. Compare this with the approach of [1] where in the E step the hidden variables are assumed to be independent but in the model selection step an optimization procedure is used to find non-fragmented support maps.

Our approach is similar to that of Wang and Adelson [14] in that we fit models to optic flow rather than directly to the image measurements, but differs in that we also use covariances in the fit. More importantly, our algorithm only computes a segmentation of two frames, whereas the Wang and Adelson algorithm derived a much more complicated description: a layered representation of the whole scene.

5 Conclusion

As others have argued, motion segmentation can be profitably considered in a mixture estimation framework. In this work we have addressed two shortcomings of previous mixture formulations. We have shown how spatial constraints can be incorporated into the framework by assuming a prior distribution on the hidden variables that models the dependence of one location's assignment on that of its neighbors, and how the number of models can be estimated automatically given assumptions about the expected level of model failure.

Our work is motivated by the belief that automated segmentation of arbitrary image sequences will only be possible if static form constraints are used. Although we have used rather rudimentary static form constraints here, as equation 1 shows, arbitrary static analysis results can be used. The unified statistical framework we have introduced here provides a basis for future investigations into the analysis needed to insure automatic and robust scene segmentation.

Acknowledgements

We thank J. Tenenbaum, J.Y.A. Wang and M. Black for comments on previous versions of this manuscript. YW is supported by a training grant from NIGMS. EA is supported by NEI RO1 EY1105.

References

[1] Serge Ayer and Harpreet S. Sawhney. Layered representation of motion video using robust maximum likelihood estimation of mixture models and mdl encoding. In *Proc. Int'l Conf. Comput. Vision*, pages 777–784, 1995.

[2] M. J. Black and A. Jepson. Estimating multiple independent motions in segmented images using parametric models with local deformations. In *IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, November 1994. (in press).

[3] T. Darrell and A. Pentland. Robust estimation of a multi-layered motion representation. In *Proc. IEEE Workshop on Visual Motion*, pages 173–178, Princeton, New Jersey, October 1991.

[4] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Statist. Soc. B*, 39:1–38, 1977.

[5] M Etoh and Y. Shirai. Segmentation and 2d motion estimation by region fragments. In *Proc. Int'l Conf. Comput. Vision*, pages 192–199, 1994.

[6] D. Geiger and F. Girosi. Parallel and deterministic algorithms from MRFs: surface reconstruction. *IEEE Trans. PAMI*, 13(5):401–412, 1991.

[7] M. Irani and S. Peleg. Image sequence enhancement using multiple motions analysis. In *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, pages 216–221, Champaign, Illinois, June 1992.

[8] A. Jepson and M. J. Black. Mixture models for optical flow computation. In *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, pages 760–761, New York, June 1993.

[9] R.M. Neal and G.E. Hinton. A new view of the EM algorithm that justifies incremental and other variants. *Biometrika*, 1993. submitted.

[10] J.M. Odobez and P. Bouthemy. Detection of multiple moving objects using multiscale mrf with camera motion compensation. In *Proc. ICIP*, pages II 257–261, 1994.

[11] K. Rose, F. Gurewitz, and G. Fox. Statistical mechanics and phase transitions in clustering. *Physical Review Letters*, 65:945–948, 1990.

[12] E.P. Simoncelli, E.H. Adelson, and D.J. Heeger. Probability distributions of optical flow. In *Proc. IEEE Conf. Comput. Vision Pattern Recog.*, pages 310–315, 1991.

[13] J. B. Tenenbaum and E. V. Todorov. Factorial learning by clustering features. In G. Tesauro, D.S. Touretzky, and K. Leen, editors, *Advances in Neural Information Processing Systems 7*, 1995.

[14] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *IEEE Transactions on Image Processing Special Issue: Image Sequence Compression*, 3(5):625–638, September 1994.

[15] A. Yuille, P. Stolorz, and J. Ultans. Statistical physics, mixtures of distributions and the EM algorithm. *Neural Computation*, 6:334–340, 1994.

[16] J. Zhang, W. Modestino, and D.A. Langan. Maximum-likelihood parameter estimation for unsupervised model-based image segmentation. *IEEE Transactions on Image Processing*, 3(4):404–420, July 1994.

A Affine segmentation given optic flow and covariance

We derive here the M step when we the deviation meaure $D_k(r)$ is given by:

$$D_k(r) = (v_k - \mu)^t \Sigma^{-1} (v_k - \mu) \tag{2}$$

and the velocity fields are assumed to be a sum of N basis functions (e.g. six in the case of affine motion). Define $\Psi(r)$ a 2 by N matrix which give the two components of the basis functions at location r (i.e. $v_k(r) = \Psi(r)\theta_k$), then taking the derivative of the log likelihood with respect to θ_k gives:

$$\left(\sum_r g_k(r)\Psi^t(r)\Sigma^{-1}(r)\Psi(r)\right)\theta = \left(\sum_r g_k(r)\Psi^t(r)\Sigma^{-1}\mu(r)\right) \tag{3}$$

Which gives an N by N system of equations.

Adding hyper-priors on θ

The N by N system of equations obtained above may be ill-conditioned. We add hyper-priors on θ by assuming a prior distribution on the flow fields of each model:

$$\log P(v) = -\lambda_1 \sum_r \|v(r)\| - \lambda_2 \sum_r \|\partial_x v(r)\| + \|\partial_y v(r)\| \tag{4}$$

Now define a matrix $\Psi_x(r)$ which gives the partial derivative with respect to x of the basis functions at location r and similarly $\Psi_y(r)$. Then this gives a prior distribution on the parameters of each model:

$$\log P(\theta) = -\theta^t M_\lambda \theta \tag{5}$$

With:

$$M_\lambda = \left(\sum_r \lambda_1 \Psi^t(r)\Psi(r) + \lambda_2 \Psi_x^t(r)\Psi_x(r) + \lambda_2 \Psi_y^t(r)\Psi_y(r)\right) \tag{6}$$

Taking the derivative of the prior with respect to θ gives:

$$M_\lambda \theta = 0 \tag{7}$$

Which should be added to equation 3 to give an N by N system of equations for θ in the M step. Note that M_λ can be precomputed in advance. In particular for the case of affine basis functions, it is easy to show that M_λ is a diagonal matrix.