An EM-like Algorithm for Motion Segmentation via Eigendecomposition

Antonio Robles-Kelly* Edwin R. Hancock
Department of Computer Science
The University of York, York, Y01 5DD, UK
email: arobkell,erh@cs.york.ac.uk

Abstract

This paper presents an iterative maximum likelihood framework for motion segmentation via the pairwise checking of pixel blocks. We commence from a characterisation of the motion blocks in terms of a matrix of pairwise similarity weghts for their motion vectors. The eigenvectors of this similarity weight matrix represent the initial pairwise clusters, i.e the independant motions present in the scene. We develop a maximum likelihood framework which allows to update both the link weight matrix and the associated set of pairwise clusters. We experiment with the resulting clustering method on a number of real world motion sequences. Here ground truth data indicates that the method can result in motion classification errors as low as 3%.

1 Introduction

There has recently been considerable interest in the use of probabilistic methods for motion segmentation and analysis. At the segmentation-level several authors have exploited the apparatus of Markov random fields [1]. For instance, Konrad and Dubois have developed a maximum *a posteriori* probability framework for simultaneous motion estimation and moving object segmentation [2]. These low-level approaches rely on the similarity in motion, grey-level and appearance among groups of pixels from the image sequence. At higher level several authors including Maclean and Jepson [3], Black and Anandan [4], and Adelson and Weiss [5] have used the EM algorithm [6] to detect independently moving objects. Focussing on the issue of optmisation, Isard and Blake [7] have developed a maximum likelihood sampling method known as CONDENSATION to track independently moving objects.

There are two difficulties which must be overcome in motion analysis. The first of these is that of defining motion coherence. Most approaches adopt a model based on central clustering. When the EM algorithm is used then each individual object is represented by a separate Gaussian distribution of motion vectors. Each distribution has its own mean motion vector and covariance matrix which need to be estimated. The second problem is that of controlling the number of independent motion components. In the case of the EM algorithm, the set of moving objects is represented using a mixture model and the order of the mixture model is equal to the number of moving objects in the scene. There are two aspects to the problem of setting the order of a mixture model. The first of these is to choose a utility measure which measures the tradeoff between data-closeness and

* Supported by CONACYT, under grant No. 146475/151752.

model complexity [8, 9]. The possibilities here include the minimum description length, the Aikake criterion and likelihood ratios. The second task is how use the utility measure to control the splitting or merging of mixture components [10].

Because of the dual problems of estimating the parameters of the motion mixture model and of controlling its structure, the EM algorithm has proved to be cumbersome to use in practice. In this paper we describe a more easily controlled motion segmentation strategy. The method uses a maximum-likelihood method to detect moving objects by performing pairwise clustering on a set of motion vectors. We commence from an initial characterisation of the motion structure using a matrix of motion block similarity weights. The elements of the matrix are calculated from the scalar products of local motion vectors. These motion vectors are obtained using a block matching algorithm. The number of clusters, i.e. the number of independently moving objects, is controlled using the set of same-sign positive eigenvectors of the motion block similarity weight matrix. This is an idea that has its roots in spectral graph theory [11]. In particular, we draw on the work of Sarkar and Boyer [12] who have used the leading eigenvector of the similarity weight matrix to perform line-grouping. In the so-called normalised cut method, Shi and Malik [13] use the second smallest eigenvalue to threshold the similarity matrix to preform image segmentation. Perona and Freeman [14] extend this work by showing that the second largest eigenvector of the affinity matrix can be used for both point-set clustering and line-grouping. Weiss has shown how each of these methods can be improved by nomalising the elements of the similarity weight, or affinity, matrix [15].

However, our novel contribution here is to cast the spectral approach into a probabilistic setting and to exploit it for independent motion detection. Starting from a model in which the independently moving objects are represented by a Bernoulli distribution, we develop a dual-step approach to moving object detection. This dual-step algorithm is closely akin to the EM algorithm [6]. In the E-step we update the cluster membership probabilities In the M-step we locate revised motion block similarity weights which maximize the expected log-likelihood function. The algorithm iterates until convergence. In this way we circumvent the problem of defining motion coherence to group motion blocks. Our method is based on the pairwise similarity of velocity vectors, rather than their proximity to the centre of a cluster. Moreover, the number of motion components is controlled using the distinct eigenmodes of the pairwise similarity matrix for the motion vectors.

The outline of this paper is as follows. In Section 2 we discuss the use of block-matching for motion vector estimation. We introduce the use of segmentation by matrix factorization in Section 3. In Section 4 we describe our maximum likelihood framework for pairwise clustering. Section 5 details a hierarchical refinement which can be used to improve the motion field segmentations returned by the pairwise clustering method. Experimental results on real-world motion sequences with ground truthh are provided in Section 6. Finally, in Section 7 we conclude the paper and offer some suggestions for future work.

2 Computing Motion Vectors

The motion vectors used in our analysis have been computed using a single resolution block matching algorithm [16]. The method measures the similarity of motion blocks using spatial correlation and uses predictive search to efficiently compute block-correspondences in different frames. The block matching algorithm assumes that the translational motion from frame to frame is constant. The current frame is divided into blocks that will be compared with the next frame in order to find the displaced coordinates of the corresponding block within the search area of the reference frame. Since the computational complexity is much lower than the optical flow equation and the pel-recursive methods, block matching has been widely adopted as a standard for video coding and hence it provides a good starting point.

However, the drawback of the single resolution block-matching scheme is that while the high resolution field of motion vectors obtained with small block sizes captures fine detail, it is susceptible to noise. At low resolution, i.e. for large block sizes, the field of motion vectors is less noisy but the fine structure is lost. To strike a compromise between low-resolution noise suppression and high resolution recovery of fine detail, there have been several attempts to develop multi-resolution block matching algorithms. These methods have provided good predictive performance and also improvements in speed. However, one of the major problems with the multi-resolution block matching method is that random motions can have a significant degradational effect on the estimated motion field. For these reasons, we have used a single high-resolution block matching algorithm to estimate the raw motion field. This potentially noisy information is refined in the motion segmentation step, where we exploit hierarchical information.

3 Motion Modes by Matrix Factorization

We pose the problem of grouping motion blocks into coherent moving objects as that of finding pairwise clusters. This process of pairwise clustering is somewhat different to the more familiar one of central clustering. Whereas central clustering aims to characterise cluster-membership using the cluster mean and variance, in pairwise clustering it is the relational similarity of pairs of objects which are used to establish cluster membership. Although less well studied than central clustering, there has recently been renewed interest in pairwise clustering aimed at placing the method on a more principled footing using techniques such as mean-field annealing [17].

To commence, we require some formalism. The 2D velocity vectors for the extracted motion blocks are characterised using a matrix of pairwise similarity weights. Suppose that $\hat{\mathbf{n}}_i$ and $\hat{\mathbf{n}}_j$ are the unit motion vectors for the blocks indexed i and j. The elements of this weight matrix are given by

$$W^{(0)}_{i,j} = \begin{cases} \frac{1}{2}(1 + \hat{\mathbf{n}}_i.\hat{\mathbf{n}}_j) & \text{if } i \neq j \\ 0 & otherwise \end{cases} \tag{1}$$

The aim in pairwise clustering is to locate the updated set of similarity weights which partition the image into regions of uniform motion. To be more formal, let V denote the index-set of the detected motion blocks in the image and suppose that Ω is the set of pairwise-clusters, i.e. distinct moving objects, to which these blocks are to be assigned. The initial set of clusters are defined by the eigenmodes of the link-weight matrix $W^{(0)}$. Here we follow Sarkar and Boyer [12] who have shown how the positive eigenvectors of the matrix of link-weights can be used to assign objects to perceptual clusters. Using the Rayleigh-Ritz theorem, they observe that the scalar quantity $\underline{x}^t W^{(0)} \underline{x}$ is maximised when

$\mathbf{x}$ is the leading eigenvector of $W^{(0)}$. Moreover, each of the subdominant eigenvectors corresponds to a disjoint pairwise cluster. They confine their attention to the same-sign positive eigenvectors (i.e. those whose corresponding eigenvalues are real and positive, and whose components are either all positive or are all negative in sign). If a component of a positive same-sign eigenvector is non-zero, then the corresponding object belongs to the associated cluster of motion blocks. The eigenvalues $\lambda_1, \lambda_2 ....$ of $W^{(0)}$ are the solutions of the equation $|W^{(0)} - \lambda I| = 0$ where I is the $|V| \times |V|$ identity matrix. The corresponding eigenvectors $\mathbf{x}_{\lambda_1}, \mathbf{x}_{\lambda_2}, ...$ are found by solving the equation $W^{(0)}\mathbf{x}_{\lambda_i} = \lambda_i \mathbf{x}_{\lambda_i}$. With this notation, the set of positive same-sign eigenvectors is represented by $\Omega = \{\omega | \lambda_\omega > 0 \wedge [(\mathbf{x}^*_\omega(i) > 0 \forall i) \vee \mathbf{x}^*_\omega(i) < 0 \forall i])\}$.

4 Maximum Likelihood Framework

In this paper, we are interested in exploiting the factorisation property of Sarkar and Boyer [12] to develop a maximum likelihood method for updating the motion block similarity weight matrix W with the aim of developing an easily controlled motion segmentation method. We commence by factorising the likelihood over the set of modal clusters of the motion block similarity weight matrix. Since the set of modal clusters are disjoint we can write,

$$P(W) = \prod_{\omega \in \Omega} P(\Phi_\omega) \tag{2}$$

where $P(\Phi_\omega)$ is the probability distribution for the set of motion block similarity weights belonging to the modal-cluster indexed ω. Next, we assume that there are putative links between each pair of motion blocks (i, j) belonging to the cluster. The set of putative links is $\Phi_\omega = V_\omega \times V_\omega - \{(i,i) | i \in V\}$. If the motion block similarity weights for the individual clusters are independent of one another, we can write

$$P(\Phi_\omega) = \prod_{(i,j) \in \Phi_\omega} P(W_{i,j}) \tag{3}$$

To proceed, we require a model of probability distribution for the motion vector similarity weights. To develop this model, we introduce a cluster membership indicator $s_{i\omega}$ which represents the degree of affinity of the object indexed i to the cluster with modal index ω. Our model is developed under the assumption that the motion blocks associate to form objects or clusters under a Bernoulli distribution. The parameter of this distribution is the similarity weight $W_{i,j}$. The probability that the block association is correct is $W_{i,j}$ while the probability that it is in error is $1 - W_{i,j}$. To gauge the correctness of the block association, we check whether the blocks i and j belong to the same pairwise cluster. To test for cluster-consistency we make use of the quantity $s_{i\omega} s_{j\omega}$. This is unity if both block belong to the same object or cluster and is zero otherwise. Using this switching property, the Bernoulli distribution becomes

$$p(W_{i,j}) = W_{i,j}^{s_{i\omega} s_{j\omega}} (1 - W_{i,j})^{1 - s_{i\omega} s_{j\omega}} \tag{4}$$

This distribution takes on its largest values when either the motion vector similarity weight W_{ij} is unity and $s_{i\omega} = s_{j\omega} = 1$, or if $W_{i,j} = 0$ and $s_{i\omega} = s_{j\omega} = 0$.

With these ingredients the log-likelihood function for the observed observed set of motion vector similarity weights is

$$\mathcal{L} = \sum_{\omega \in \Omega} \sum_{(i,j) \in \Phi_\omega} \left\{ s_{i\omega} s_{j\omega} \ln W_{ij} + (1 - s_{i\omega} s_{j\omega}) \ln(1 - W_{i,j}) \right\} \tag{5}$$

After some algebra to collect terms, the log-likelihood function simplifies to

$$\mathcal{L} = \sum_{\omega \in \Omega} \sum_{(i,j) \in \Phi_\omega} \left\{ s_{i\omega} s_{j\omega} \ln \frac{W_{ij}}{1 - W_{ij}} + \ln(1 - W_{i,j}) \right\} \tag{6}$$

Posed in this way the structure of the log-likelihood function has several features that are reminiscent of that underpinning the expectation-maximisation algorithm. First, the modes of the motion vector similarity weight matrix play the role of mixing components. Second, the product of cluster-membership variables $s_{i\omega} s_{j\omega}$ plays the role of an *a posteriori* measurement probability. Third, the similarity weights are the parameters which must be estimated. However, there are important differences. The most important of these is that the modal clusters are disjoint. As a result there is no mixing between them.

Based on this observation, we will exploit an EM-like process to update the motion block similarity weights and the cluster-membership variables. In the "M" step we will locate maximum likelihood block similarity-weights. In the "E" step we will use the revised similarity-weight matrix to update the modal clusters. To this end we index the similarity weights and cluster memberships with iteration number and aim to optimise the quantity

$$Q(W^{(n+1)} | W^{(n)}) = \sum_{\omega \in \Omega} \sum_{(i,j) \in \Phi_\omega} \left\{ s^{(n)}_{i\omega} s^{(n)}_{j\omega} \ln \frac{W^{(n+1)}_{ij}}{1 - W^{(n+1)}_{ij}} + \ln(1 - W^{(n+1)}_{i,j}) \right\} \tag{7}$$

The revised motion block similarity weights are indexed at iteration $n + 1$ while the cluster-memberships are indexed at iteration n.

4.1 Expectation

To update the cluster-membership variables we have used a gradient-based method. We have computed the derivatives of the expected log-likelihood function with respect to the cluster-membership variable

$$\frac{\partial Q(W^{(n+1)} | W^{(n)})}{\partial s^{(n+1)}_{i\omega}} = \sum_{j \in V_\omega} s^{(n)}_{j\omega} \ln \frac{W^{(n+1)}_{ij}}{1 - W^{(n+1)}_{ij}} \tag{8}$$

Since the associated saddle-point equations are not tractable in closed form, we use the soft-assign ansatz of Bridle [18] to update the cluster membership assignment variables. This involves exponentiating the partial derivatives of the expected log-likelihood function in the following manner

$$s^{(n+1)}_{i\omega} = \frac{\exp\left[\frac{\partial Q(W^{(n+1)} | W^{(n)})}{\partial s^{(n)}_{i\omega}}\right]}{\sum_{i \in V_\omega} \exp\left[\frac{\partial Q(W^{(n+1)} | W^{(n)})}{\partial s^{(n)}_{i\omega}}\right]} \tag{9}$$

As a result the update equation for the cluster membership indicator variables is

$$s_{i\omega}^{(n+1)} = \frac{\exp\left[\sum_{j\in V_\omega} s_{j\omega}^{(n)} \ln \frac{W_{i,j}^{(n+1)}}{1-W_{ij}^{(n+1)}}\right]}{\sum_{i\in V_\omega} \exp\left[\sum_{j\in V_\omega} s_{j\omega}^{(n)} \ln \frac{W_{i,j}^{(n+1)}}{1-W_{ij}^{(n+1)}}\right]} = \frac{\prod_{j\in V_\omega}\left\{\frac{W_{i,j}^{(n+1)}}{1-W_{ij}^{(n+1)}}\right\}^{s_{j\omega}^{(n)}}}{\sum_{i\in V_\omega}\prod_{j\in V_\omega}\left\{\frac{W_{ij}^{(n+1)}}{1-W_{ij}^{(n+1)}}\right\}^{s_{j\omega}^{(n)}}}$$

We initialise the cluster memberships using the components of the same-sign positive eigenvectors. We set

$$s_{iw}^{(0)} = \frac{|\mathrm{x}_\omega^*(i)|}{\sum_{i\in V_\omega} |\mathrm{x}_\omega^*(i)|} \tag{10}$$

Using these variables, we develop a model of probability distribution for the similarity or link weights associated with the individual clusters.

4.2 Maximisation

Once the revised cluster membership variables are to hand then we can apply the maximisation step of the algorithm to update the motion block similarity weight matrix. The updated entries of the weight matrix are found by computing the derivatives of the expected log-likelihood function

$$\frac{\partial Q(W^{(n+1)}|W^{(n)})}{\partial W_{ij}^{(n+1)}} = \sum_{\omega\in\Omega} \zeta_{i,j,\omega}^{(n)} \left\{ s_{i\omega}^{(n)} s_{j\omega}^{(n)} \frac{1}{W_{ij}^{(n+1)}(1-W_{ij}^{(n+1)})} - \frac{1}{1-W_{i}j^{(n+1)}} \right\} \tag{11}$$

and solving the saddle-point equations

$$\frac{\partial Q(W^{(n+1)}|W^{(n)})}{\partial W_{ij}^{(n+1)}} = 0 \tag{12}$$

As a result the updated elements of the weight matrix are given by

$$W_{ij}^{(n+1)} = \sum_{\omega\in\Omega} s_{i\omega}^{(n)} s_{j\omega}^{(n)} \tag{13}$$

In other words, the similarity weight for the pair of blocks (i, j) is simply the average of the product of individual cluster memberships over the different clusters. Since each mode is associated with a unique cluster, this means that the updated affinity matrix is composed of non-overlapping blocks. Moreover, the motion block similarity weights are are guaranteed to be in the interval $[0, 1]$.

5 Hierarchical Motion Segmentation

As mentioned earlier, we use a single-level high-resolution block-matching method to estimate the motion field. The resulting field of motion vectors is therefore likely to be noisy. To control the effects of motion-vector noise, we have developed a multi-resolution extension to the clustering approach described above.

The adopted approach is as follows.

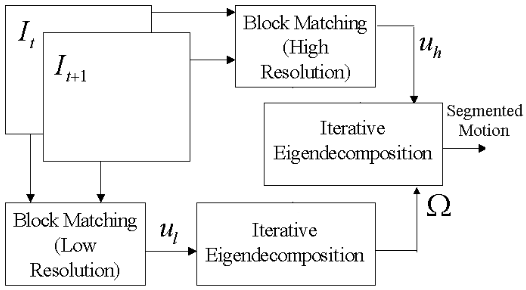

Figure 1: Motion segmentation system.

- We obtain the a high resolution field of motion vectors U_H using blocks of size k-pixels and a low-resolution motion field U_L using blocks of size $2k$ pixels.
- We apply our clustering algorithm to the low resolution motion field U_L. We note the number of clusters N_L detected.
- We make a second application of our clustering algorithm to the hight-resolution motion field U_H. Here we select only the first N_L eigenvalues of the motion-vector similarity matrix as cluster centres.

In this way we successively perform the motion estimation at low and high resolution. The number of clusters detected at low resolution is used to constrain the number of permissible high resolution clusters. This allows the high-resolution clustering process to deal with fine detail motion fields without succumbing to noise. There is scope to extend the method and develop a pyramidal segmentation strategy. The structure of the hierarchical system can be seen in Fig 1.

6 Experiments

We have conducted experiments on motion sequences with known ground truth. In Figure 3 we show some results obtained with five frames of the well-known "Hamburg Taxi" sequence. The top row shows the hand-labelled ground-truth segmentation for the motion sequence. The second row shows the corresponding image frames from the motion sequence. In the third and fourth rows we respectively show the low resolution and high resolution block motion vectors. The low-resolution uses 16×16 pixel blocks to perform motion correspondence and compute the motion vectors; for the high resolution motion field the block size is 8×8 pixels. The fifth row shows the moving objects segmented from the motion field using pairwise clustering. In each frame there are 3 clusters which match closely to the ground truth data shown. In fact, the three different clusters correspond to distinct moving vehicles in the sequence. These clusters again match closely to the ground-truth data. It is interesting to note that the results are comparable to those reported in [19] where a 5 dimensional feature vector and a neural network was used. The proposed algorithm converges in an average of four iterations.

In Table 1 we provide a more quantitive analysis of these results. The table lists the fraction of the pixels in each region of the ground truth data which are misasigned by the clustering algorithm. The best results are obtained for the chest-region, the taxi and the far-left car, where the error rate is a few percent. For the far-right car and the head of the Trevor White, the error rates are about 10%. The problems with the far-right car probably relate to the fact that it is close to the periphery of the image.

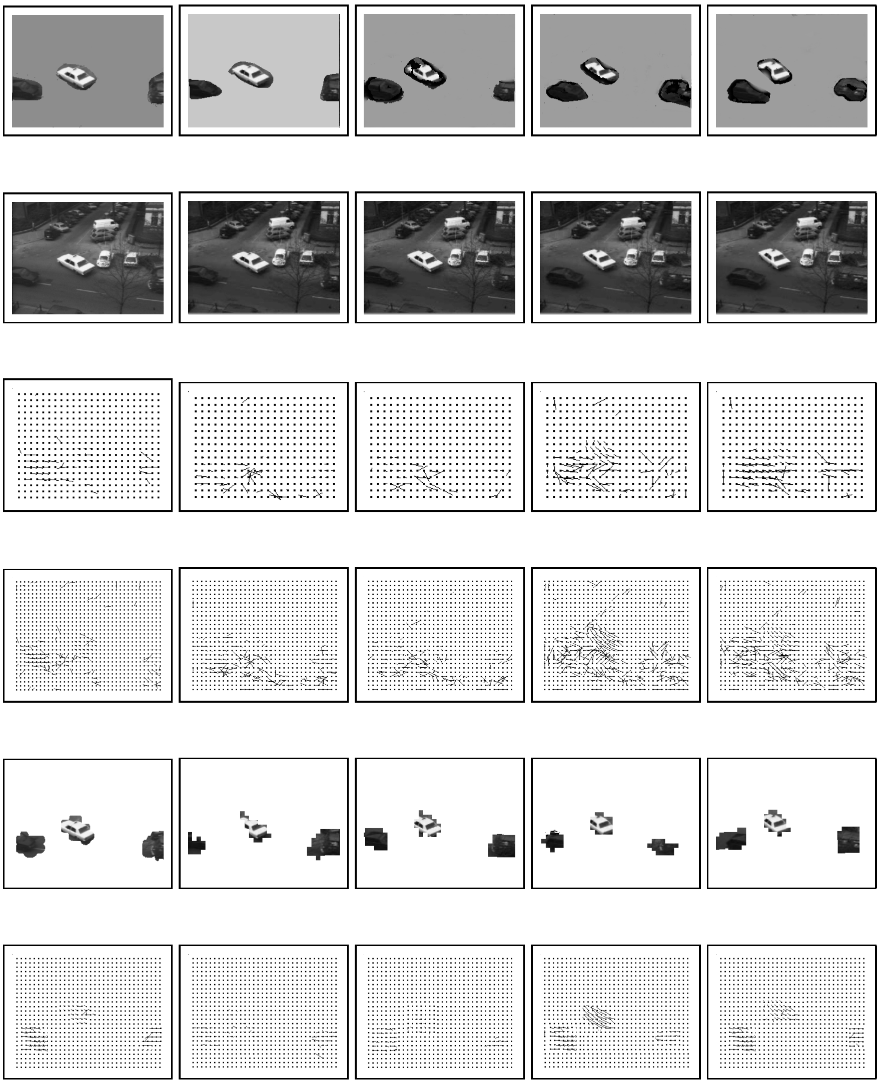

Figure 2: Top row: ground truth for the 1st, 4th, 8th, 12th and 16th frame of the "Hamburg Taxi" sequence; Second row: original frames ; Third and fourth row: resolution motion maps; Fifth row: Final motion segmentation; Bottom row: smoothed motion maps.

Sequence	Cluster	Average % Error
Ham. Taxi	Taxi	5.4 %
Ham. Taxi	Far Left Car	4.8 %
Ham. Taxi	Far Right Car	11.9 %

Table 1: Average error percentage for the two image sequences.

Finally, in Figure 4 we show the total probability mass $R_\omega^{(n)} = \sum_{\imath \in V} s_{i\omega}^{(n)}$ for each cluster ω as function of iteration number n. This plot shows that the cluster memberships converge in about 3 iterations.

7 Conclusions

We have described a maximum-likelihood framework for segmenting fields of motion-vectors into independantly moving objects. The proposed algorithm locates pairwise clusters of motion blocks using the similarity of their motion-vectors. We develop a log-likelihood function under the assumption that the similarity weights for motion blocks belonging to the same pairwise cluster follow a Bernoulli distribution. We develop an algorithm for iteratively updating the pairwise clusters of motion blocks. The algorithm is reminiscent of the EM algorithm. In the E-step we update the cluster membership probabiliies for individual motion blocks. In the M-step, the updated cluster membership probabilities are used to make revised estimates of the pairwise similarities of the motion blocks. When applied to real world image sequences, the method is capble of delivering segmentations with error rates of a few pecent. One of the advantages of the method over the conventional EM lagorithm, widely used in motion segmentation experiments, is that the number of motion components (i.e. moving objects) is controlled using the eigen-modes of the matirx of pairwise similarity weights for the motion blocks. This avoids the need to search for the optimal number of mixture components, and, to control the splitting or merging of existing components.

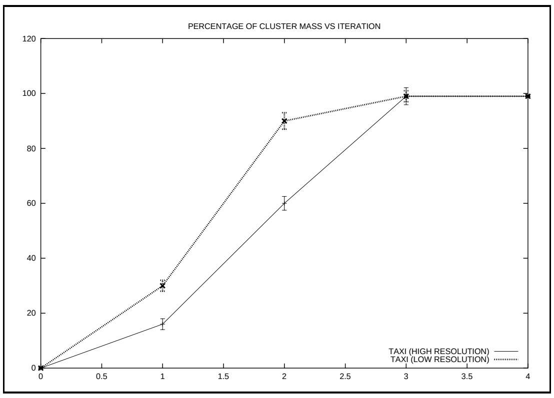

Figure 3: Fractional cluster mass versus iteration number.

References

[1] F. Heitz and P. Bouthemy. Multimodal estimation of discontinuous optical flow using Markov random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(12):1217–1232, 1993.

[2] E. Dubois J. Konrad. Bayesian estimation of motion vector fields. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(9):910–927, 1992.

[3] A. D. Jepson W. J. MacLean and R. C. Frecker. Recovery of ego-motion and segmentation of independent object motion using the EM Algorithm. In *Proceedings of the British Machine Vision Conference*, pages 175–184, 1994.

[4] M. J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, 1996.

[5] E. H. Adelson and Y. Weiss. A unified mixture framework for motion segmentation: Incorporating spatial coherence. In *Proc. IEEE Computer Vision and Pattern Recognition*, pages 321–326, 1996.

[6] N. Laird A. Dempster and D. Rubin. Maximum-likehood from incomplete data via the em algorithm. *J. Royal Statistical Soc. Ser. B (methodological)*, 39:1–38, 1977.

[7] M. Isard and A. Blake. A mixed-state condensation tracker with automatic model-switching. In *ICCV98*, pages 107–112, 1998.

[8] N. Vlassis and A. Likas. A kurtosis-based dynamic approach to gaussian mixture modeling. *IEEE Trans. in Systems, Man and Cybernetics*, 29(4):393 – 399, 1999.

[9] J. Risanen. *Stochastic Complexity in Statistical Enquiry*. World Scientific, 1989.

[10] Z. Ghahramani N. Ueda, R. Nakano and G. E. Hinton. Smem algorithm for mixture models. *Neural Computation*, 12(9):2109–2128, 2000.

[11] Fan R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.

[12] S. Sarkar and K. L. Boyer. Quantitative measures of change based on feature organization: Eigenvalues and eigenvectors. *Computer Vision and Image Understanding*, 71(1):110–136, 1998.

[13] J. Shi and J. Malik. Normalized cuts and image segmentations. In *Proc. IEEE CVPR*, pages 731–737, 1997.

[14] P. Perona and W. T. Freeman. Factorization approach to grouping. In *Proc. ECCV*, pages 655–670, 1998.

[15] Y. Weiss. Segmentation using eigenvectors: A unifying view. In *IEEE International Conference on Computer Vision*, pages 975–982, 1999.

[16] J. S. Shyn C. H. Hsieh, P. C. Lu and E. H. Lu. Motion estimation algorithm using inter-block correlation. *IEE Electron. Lett.*, 26(5):276–277, 1990.

[17] T. Hofmann and M. Buhmann. Pairwise data clustering by deterministic annealing. *IEEE Tansactions on Pattern Analysis and Machine Intelligence*, 19(1):1–14, 1997.

[18] J. S. Bridle. Training stochastic model recognition algorithms can lead to maximum mutual information estimation of parameters. In *NIPS 2*, pages 211–217, 1990.

[19] A. G. Bors and I. Pitas. Optical flow estimation and moving object segmentation based on Median RBF network. *IEEE Trans. on Image Processing*, 7(5):693–702, 1998.