MOTION ESTIMATION AND SEGMENTATION BY FUZZY CLUSTERING

Benoît Duc, Philippe Schroeter and Josef Bigün

Signal Processing Laboratory, Swiss Federal Institute of Technology
EPFL-Ecublens, CH–1015 Lausanne, Switzerland

ABSTRACT

A layered motion estimation scheme using fuzzy clustering is introduced in this paper. Once motion estimation is performed, a label smoothing is applied by using a modified objective function that takes into account spatial continuity. This allows to suppress noisy classes scattered over the whole image.

1. INTRODUCTION

Motion Estimation is a very important issue in image processing, not only as motion compensation for coding purposes, but also for analysis purposes, where motion detection and motion-based segmentation constitute important clues for surveillance and active vision applications.

Recently, model-based motion estimation techniques have gained interest against local motion estimation techniques [1, 2], because they provide a direct interpretation of the motion in terms of the chosen model, and they are more robust with respect to noise. Of course, they suffer other weaknesses: the determination of an optimal model complexity is not an easy task and is conditioned by convergence properties of algorithms and by the type of motion that is expected to occur. Usually an affine motion is chosen as a compromise when perspective effects are negligible.

Generally speaking, a motion model is intended to describe exactly one motion in a region which is not known a priori and which is large compared to a typical neighborhood used for optical flow computation. In order to deal with realistic scenes with several moving objects, it is necessary to determine motion parameters for each object as well as the regions of the image sequence where this model apply. These regions or objects are sometimes called *layers* [3].

In order to determine such regions of support and the motions, robust estimation has been used. Robust estimators are known to detect outliers of the regression at the same time as they estimate the regression parameters. One takes advantage of this property by interpreting one of the motions as the regression to be found, while layers from other motions become outliers [4]. It would be possible to find all motions by successive robust estimations on the outliers of the previous estimations.

Unfortunately, robust estimators are theoretically limited to a contamination of 50% of outliers. As a consequence, their use for motion estimation is valid only if at each estimation step, one motion dominates all remaining ones. This hypothesis is often not fulfilled.

Therefore, one would like to develop an estimation technique that is able to find several motions simultaneously, as well as their region of support, even if none of them is dominant. This technique must be robust with respect to the actual number of motions, which is not known a priori. The aim of this contribution is to show that fuzzy clustering can fulfill these requirements.

2. MOTION ESTIMATION FRAMEWORK

The model-based motion estimation framework used here is based on a spatio-temporal description of motion [5, 6]. By taking into account more than two successive images to do the estimation, one aims at increasing the robustness of the estimation.

In this approach, an image sequence is interpreted as being generated by a two-dimensional pattern undergoing a transformation through time. The instantaneous apparent motion is actually the instantaneous transformation of the pattern. In order to find the motion, one fixes a motion model complexity, for example translational or affine motion. By doing so, one restricts the search space to a family of transformations that is a Lie group of transformations. The instantaneous transformations are expressed by differential operators. Each transformation in the group is obtained by a linear combination of p basic differential operators $\mathcal{L}_i, i = 1 \dots p$, called infinitesimal generators.

The motion estimation is reformulated as searching for a transformation that leaves the image sequence invariant. In Lie theory a transformation expressed by the infinitesimal generator $\mathcal{L}$ leaves a function f invariant if and only if

$$\mathcal{L}f(x) = 0, \tag{1}$$

for each x where f is defined. Here $\mathcal{L} = \sum_{i=1}^{p} a_i \mathcal{L}_i$, so that the unknowns are actually the a_i's. As the transformation that is looked for is determined up to a scaling factor, the constraint $\sum_{i=1}^{p} a_i^2 = 1$ is added in order to make the problem completely well-posed.

Solving equation (1) in least square sense for n points of interest leads to the minimization of the following objective function:

$$\mathbf{a}^t X^t X \mathbf{a} = 0, \tag{2}$$

where a_i's are the elements of the p-dimensional vector $\mathbf{a}$ and $(\mathcal{L}_i f(\mathbf{x}), i = 1 \dots p)$ are the elements of the $n \times p$ matrix X representing all "features" for all points $\mathbf{x}$. Equation (2) together with the constraint $\|\mathbf{a}\| = 1$ leads to an eigenvalue problem. The solution is given by the eigenvector of $M = X^t X$ corresponding to the smallest eigenvalue. The estimation can also be viewed as a hyperplane fitting through the origin, in a p-dimensional "feature" space.

0-8186-7310-9/95 $4.00 © 1995 IEEE

3. FUZZY C-VARIETY CLUSTERING

The problem now consists in determining layers whose points should be incorporated to the matrix X for a particular motion estimation or, equivalently, which weights to assign to points in case all available points are taken into account. We propose to achieve it by fuzzy clustering. The fuzzy c-variety algorithm for a feature space with dimension p, aims at clustering points that belong to c *linear varieties* V_{i_r}, namely lines ($r = 1$), planes ($r = 2$), or hyperplanes (up to $r = p - 1$), see Bezdek [7]. A linear variety V_{i_r} of dimension r through point $\mathbf{v}_i \in R^p$, spanned by the linearly independent vectors $\{\mathbf{s}_{i1}, \ldots, \mathbf{s}_{ir}\} \subset R^p$, is the set:

$$V_{i_r}(\mathbf{v}_i, \mathbf{s}_{i1}, \mathbf{s}_{i2}, \ldots, \mathbf{s}_{ir}) = \{\mathbf{y} \in R^p \mid \mathbf{y} = \mathbf{v}_i + \sum_{j=1}^{r} t_j \mathbf{s}_{ij}, t_j \in R\} \qquad , \tag{3}$$

The algorithm consists in minimizing the following objective function:

$$J_m(U, \mathbf{v}) = \sum_{i=1}^{c} \left(\sum_{k=1}^{n} u_{ik}^m (d_{ik})^2 \right) \tag{4}$$

with the constraint $\sum_{i=1}^{c} u_{ik} = 1, \forall k$; u_{ik} is the membership weight of point $\mathbf{x}_k$ for variety V_{i_r}; m is the fuzzy exponent, which controls the fuzziness of the final result (the larger m, the fuzzier the clustering); d_{ik} is the distance of point $\mathbf{x}_k$ to the *linear variety* V_{i_r}, that is the distance to its orthogonal projection onto the variety:

$$d_{ik} = [\|\mathbf{x}_k - \mathbf{v}_i\|^2 - \sum_{j=1}^{r} (\langle \mathbf{x}_k - \mathbf{v}_i, \mathbf{s}_{ij} \rangle)^2]^{1/2}. \tag{5}$$

The minimization of the objective function is achieved by an iterative algorithm that successively updates the weights u_{ik} and the linear varieties, i.e. the vectors $\mathbf{v}_i, \mathbf{s}_{i1}, \ldots, \mathbf{s}_{ir}$.

As the estimation of multiple motions is equivalent to fitting hyperplanes through the origin, one applies fuzzy c-variety clustering to multiple motion estimation by taking $r = p - 1$, and by constraining the variable $\mathbf{v}_i$ to be 0. The update of the hyperplanes corresponds actually to motion estimations, and the update of the u_{ik}'s corresponds to the determination of the layers of support, thus showing a similar structure as the EM algorithm used in [8]. It is important to note that fuzzy algorithms can be robustified by adding a noise class that is characterized by a constant distance to the centroid of the class [9].

A hard labeling is currently achieved by attributing each point to the class with maximum belongingness. As values of m are chosen rather close to 1 in experiments (namely $m = 1.4$ for the examples below), this decision is justified because in that case, membership weights are expected to be close to the hard case.

3.1. Determination of the number of motions

The number of classes in clustering algorithms is itself an unknown parameter. Usually in fuzzy clustering, the number of classes is evaluated by comparing cluster characteristics obtained with various values of c according to a criterion such as fuzzy classification entropy or partition coefficient. As these criteria are based on the feature space only, they are not useful in our approach, where the spatial relationship of points is significant. Namely, one would like to obtain compact label regions, as in real image sequences moving objects are expected to be compact.

The approach adopted here consists in imposing a number of classes a priori larger than the actual number of motions. Then a smoothing operation in performed by adding a spatial constraint on membership weights to the objective function, while frozing the motion parameters.

4. CONSTRAINED FUZZY C-MEANS

In this section, we present a new algorithm that combines the fuzzy c-means [7] and the Gibbs distribution [10] for the classification of data taking into account the spatial connectivity of images. This allows us to reduce the noise and eventually to eliminate noisy classes in the classification result. The algorithm is based on a Maximum *a posteriori* Estimation (MAP) technique but adapted to the fuzzy sets.

Let us define $\mathbf{u}_k$ as the vector of the degree of belongingness of the k-th element of the data set $X = \{x_1, \ldots, x_n\}$, η_k represents a neighborhood system and $\hat{\mathbf{u}}_{\eta_k}$ denotes the provisional estimates of the true membership vectors of the neighbors of $\mathbf{u}_k$ at the current stage of the iteration. The problem can be formulated as follows (details can be found in [11]):

$$\max_k P(\mathbf{u}_k \mid \mathbf{x}_k, \hat{U}) = \max_k P(\mathbf{u}_k \mid \mathbf{x}_k, \hat{\mathbf{u}}_{\eta_k}) \qquad \forall k \tag{6}$$

The equality holds because we assume that the image of membership vectors $U = \{\mathbf{u}_1, \ldots, \mathbf{u}_n\}$ is the realization of a Gibbs distribution. Thus U (and $\hat{U}$) has the properties of Markov Random Fields [10]. Using Bayes's theorem, the problem can be reformulated as follows:

$$\max_k P(\mathbf{x}_k \mid \mathbf{u}_k) P(\mathbf{u}_k \mid \hat{\mathbf{u}}_{\eta_k}) \qquad \forall k \tag{7}$$

If we assume a mixture of Normal distribution for the *posterior* $P(\mathbf{x}_k \mid \mathbf{u}_k)$ and a Gibbs distribution for the *prior* $P(\mathbf{u}_k \mid \hat{\mathbf{u}}_{\eta_k})$

$$P(\mathbf{x}_k \mid \mathbf{u}_k) = (2\pi det(\sigma_N))^{-1/2} \exp \frac{-1}{2} (\mathbf{x}_k - V\mathbf{u}_k)^t \sigma_N^{-1} (\mathbf{x}_k - V\mathbf{u}_k) \tag{8}$$

$$P(\mathbf{u}_k \mid \hat{\mathbf{u}}_{\eta_k}) = \frac{1}{Z} \exp(-\sum_{j \in \eta_k} \beta \kappa_j \|\hat{\mathbf{u}}_j - \mathbf{u}_k\|^2), \tag{9}$$

maximizing the logarithm of equation (7) is similar than minimizing the following expression

$$\frac{1}{2} (\mathbf{x}_k - V\mathbf{u}_k)^t \sigma_N^{-1} (\mathbf{x}_k - V\mathbf{u}_k) + \sum_{j \in \eta_k} \beta \kappa_j \|\hat{\mathbf{u}}_j - \mathbf{u}_k\|^2 \tag{10}$$

$\forall k$, where $V = [\mathbf{v}_1, \ldots, \mathbf{v}_c]$ is the matrix of mean vectors, σ_N is the covariance matrix of the noise process, β is the parameter of the Gibbs distribution and κ_j are weighting factors. The first part of equation (10) is simply a distance between the object $\mathbf{x}_k$ and the centroids $\mathbf{v}_i$ weighted by the vector of membership values $\mathbf{u}_k$. This distance can be compared to the distance $\sum_{i=1}^{c}(u_{ik})^m\|\mathbf{x}_k - \mathbf{v}_i\|^2$ defined in the fuzzy c-means. The second part of equation (10) is a connectivity constraint on the membership values induced by the Gibbs distribution. Note that here we chose $\sum_{j\in\eta_k}\beta\kappa_j\|\hat{\mathbf{u}}_j - \mathbf{u}_k\|^2$ as the sum of the potentials for describing piecewise constant regions, but other energy functions can be used [10].

Following equation (10), the constrained fuzzy c-means (CFCM) can be derived by minimizing a new objective function $J = J(U, \mathbf{v})$

$$J = \sum_{k=1}^{n}\left(\sum_{i=1}^{c} u_{ik}^m (d_{ik})^2 + \sum_{j\in\eta_k}\beta\kappa_j\|\hat{\mathbf{u}}_j - \mathbf{u}_k\|^2\right) \quad (11)$$

where $(d_{ik})^2 = (\mathbf{x}_k - \mathbf{v}_i)^t A(\mathbf{x}_k - \mathbf{v}_i)$ with matrix A semi-positive definite. The problem is to minimize equation (11) under the constraint $\sum_{i=1}^{c} u_{ik} = 1$. Although the columns of matrix U are not independent, we will carry on the minimization process on each component independently, thus minimizing:

$$\sum_{k=1}^{n}\min\left(\sum_{i=1}^{c} u_{ik}^m (d_{ik})^2 + \sum_{j\in\eta_k}\beta\kappa_j\|\hat{\mathbf{u}}_j - \mathbf{u}_k\|^2\right) \quad (12)$$

This does not guaranty that we will reach the same minimum as the one we would obtained by minimizing equation (11). However, it allows us to obtain a simple solution for determining the degrees of belongingness. Following the same steps than the ones used for deriving the fuzzy c-means (see [7]), a closed form solution for updating the membership values can be obtained for $m = 2$ and is expressed by

$$\begin{aligned} u_{ik} &= \left(\sum_{s=1}^{c}\frac{d_{ik}^2 + \alpha}{d_{sk}^2 + \alpha}\right)^{-1} \\ &\cdot \left(1 + \beta\sum_{s=1}^{c}\frac{\sum_j \kappa_j(\hat{u}_{ij} - \hat{u}_{sj})}{d_{sk}^2 + \alpha}\right) \end{aligned} \quad (13)$$

where $\alpha = \beta\sum_j \kappa_j$ is a constant and $j \in \{\eta_k, k\}$. The steps of the CFCM are the same than the ones of the fuzzy c-means except that the membership values are updated using equation (13). If $\beta = 0$ (no spatial constraint), equation (13) is simply the relation used in the fuzzy c-means for updating the membership values. As in the FCM, the values of the means are expressed by (for $m = 2$)

$$\mathbf{v}_i = \sum_{k=1}^{n} u_{ik}^2 \mathbf{x}_k \Big/ \sum_{k=1}^{n} u_{ik}^2 \quad (14)$$

For $m \neq 2$, no closed form for updating the u_{ik} can be obtained. Note also that the index $j \in \eta_k$ but also $j = k$. This means that we also take into account the distance $\|\hat{\mathbf{u}}_k - \mathbf{u}_k\|^2$ in the minimization process. The parameter β controls the degree of the spatial constrain. High values of β eliminates the noise but also fine details. A way to estimate β is proposed in [11]). However, this parameter can also be set through experiments. The CFCM algorithm can be used as a clustering algorithm thus letting the mean vectors $\mathbf{v}_i$ vary at each iterations, or it can be used as a classification algorithm. In this case, the mean vectors are first estimated (for instance using the fuzzy c-means) and stay fixed during the iterations of the CFCM (only the values of the u_{ik} are updated).

Found motions		Actual motions	
u	v	u	v
0.001	0.001	0.0	0.0
-0.002	1.020	0.0	1.0
2.010	-0.030	2.0	0.0
-3.12	0.044	-3.0	0.0
0.010	-1.117		

Table 1: Translational motions parameters obtained on the image sequence in Figure 1. Except for the fifth motion, found motions are in good correspondence with actual motions.

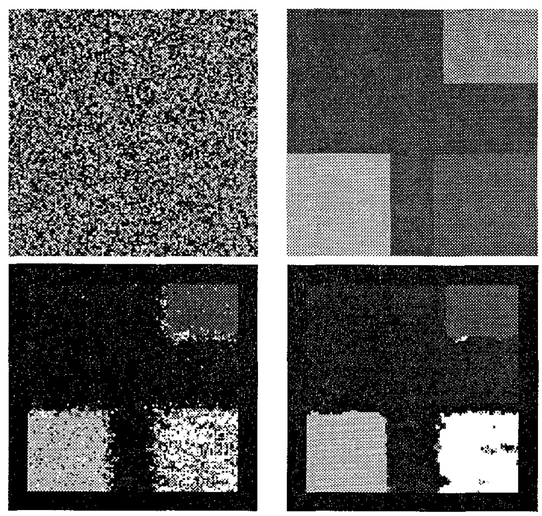

Figure 1: Simultaneous motion estimation with five linear varieties on a synthetic sequence (translations of random patterns). Top left figure: image 8 of the 16-image sequence. Top right figure: mask showing the actual regions of support for the motions. Bottom left figure: layers obtained by assigning each point to the class with the maximum membership ($= arg(max_i(u_{ik}))$). Bottom right figure: layers after label refinement . The black frames around bottom images just reflect the fact that border points have not been taken into account.

5. RESULTS

The validity of fuzzy c-variety clustering for motion estimation is first illustrated in Figure 1 where four translational motions are present. Table 1 shows that motion estimates are in good correspondence with actual motions. Robustness with respect to the number of classes c is obtained in the sense that one asked for more motions (five) than there are actually and the algorithm provided four coherent motion estimates with good regions of support and a fifth motion that corresponds to no actual motion but with a small region of support that consists of scattered points. The smoothing constraint added to the objective function once motion parameters are frozen allows to discard this region: after the smoothing procedure, no point in the image actually belongs to this class.

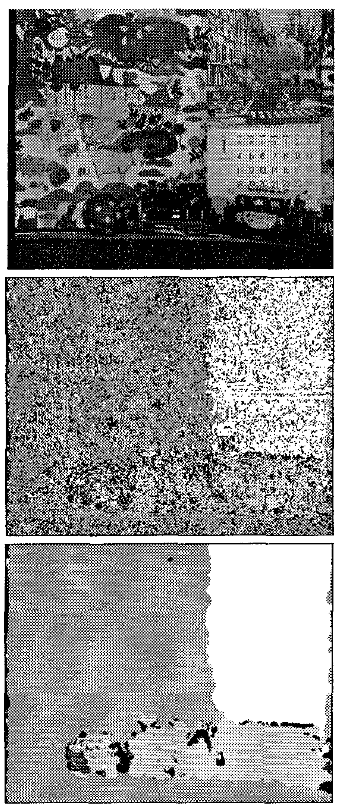

Figure 2: Motion Segmentation on "mobile and calendar" sequence. Centre: Labels before smoothing. Right: Labels after smoothing . Six classes were used. After smoothing, only four classes remain.

A real sequence example is shown in Figure 2. Here again, the smoothing constraint allows to get rid of most of the noise. However, no class has completely disappeared, although some classes have very small support.

6. CONCLUSION

In this paper, a method for simultaneous estimations of multiple motion models and of their layers of support has been presented. Segmentation results based on motion layers are encouraging. Other experiments [4] indicate that incorporating intensity information improves the motion boundaries, and future work will be done in this direction. Another promising direction for determining the number of classes consists in minimizing a cost function that takes into account not only the spatial compactness, but also the objective function [8].

7. REFERENCES

[1] B. K. P. Horn and E. J. Weldon. Direct methods for recovering motion. *International Journal of Computer Vision*, 2(1):51–76, 1988.

[2] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *Second European Conference on Computer Vision*, pages 237–252, Santa Margherita, Italy, May 1992.

[3] T. Darrell and A. P. Pentland. Cooperative robust estimation using layers of support. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):474–487, May 1995.

[4] S. Ayer, P. Schroeter, and J. Bigün. Segmentation of moving objects by robust motion parameter estimation over multiple frames. In *Third European Conference on Computer Vision*, volume 2, pages 316–327, Stockholm, Sweden, May 1994.

[5] B. Duc. Motion estimation using invariance under group transformations. In *12th International Conference on Pattern Recognition*, pages 159–163, Jerusalem, October 9-13 1994.

[6] J. Bigün, G. H. Granlund, and J. Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):775–790, August 1991.

[7] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum Press, New York, 1981.

[8] S. Ayer and H.S. Sawhney. Layered representation of motion video using robust maximum-likelihood estimation of mixture models and MDL encoding. In *Fifth International Conference on Computer Vision*, Boston, MA, USA, June 1995.

[9] R. N. Dave. Characterization and detection of noise in clustering. *Pattern Recognition Letters*, 12:657–664, November 1991.

[10] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[11] H.W. Choi, D.R. Haynor, and Y. Kim. Partial volume tissue classification of multichannel magnetic resonance images - a mixel model. *IEEE Transactions on Medical Imaging*, 10:395–407, September 1991.