Motion Segmentation by Fuzzy Clustering with Automatic Determination of the Number of Motions

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 modified objective criterion is applied to discard non significant classes.

1. Introduction

Motion estimation is a very important issue in image processing. Recently, model-based motion estimation techniques have gained interest against local motion estimation techniques [1, 2]. Indeed, 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 from 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. Such a region 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 their spatial support. These regions or objects are sometimes called *layers* [3].

One of the possible approaches consists in estimating several motions simultaneously, e.g. [4, 5]. In this context, fuzzy clustering techniques have also been used [6]. In this contribution, a motion estimation framework that uses spatio-temporal information is introduced. This formulation spans a multi-dimensional feature space where data belonging to the same motion are concentrated on a hyperplane. Finding the hyperplanes is achieved with the fuzzy c-variety clustering technique [7]. In such an approach the number of motions is an unknown parameter. Therefore, a criterion for the automatic determination of this number is introduced by modifying the fuzzy objective function. This modification takes into account spatial dependencies, which also allows to obtain smooth motion classifications.

2. Motion Estimation Framework

The model-based motion estimation framework used here is based on a spatio-temporal description of motion [8, 9]. 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 apparent motion is actually the instantaneous transformation of the pattern. In order to find the motion, one fixes a motion model, for example translational or affine. 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

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

for each point $\mathbf{r}$ in the image sequence 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 to transform the problem into a well-posed one.

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

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

1015-4651/96 $5.00 © 1996 IEEE
Proceedings of ICPR '96

where a_i's are the elements of the p-dimensional vector $\mathbf{a}$ and $\mathcal{L}_i f(\mathbf{r}), i = 1 \ldots p$ are the elements of the $n \times p$ matrix X. Actually, the kth row of X represents the feature vector $\mathbf{x}_k$ for point $\mathbf{r}_k$. 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 fitting a hyperplane through the origin, in a p-dimensional feature space. No warping is required for solving Equation (2).

3. Fuzzy c-variety clustering

The problem now consists in grouping the data into clusters that correspond to the same motion. By solving Equation (2) for each cluster, one obtains the motion parameters. Here, we propose to use the fuzzy c-variety clustering (FCV), because it is simple and has good convergence properties. This algorithm aims at clustering p-dimensional 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]. In the motion estimation framework presented below, varieties are hyperplanes of dimension $(p-1)$, going through the origin. This leads to the minimization of the following objective function:

$$J_{c,m}(U, \mathbf{n}) = \sum_{i=1}^{c} \left(\sum_{k=1}^{n} u_{ik}^m (d_{ik})^2 \right) \tag{3}$$

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 in our case:

$$d_{ik} = \left(\langle \mathbf{x}_k, \mathbf{n}_i \rangle^2 \right)^{1/2}, \tag{4}$$

where $\mathbf{n}_i$ is the normal vector to variety i. 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 $\mathbf{n}_i$.

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 [4].

The robustness of fuzzy algorithms can be increased by adding a noise class that is characterized by a constant distance δ from all data points to its centroid [10]. δ should be chosen in the order of magnitude of all distances. Thus, atypical points, which are characterized by large distances to all classes, will mainly be attributed to the noise class, and will not influence the estimation of the motions. Here, δ is chosen as the median of the smallest distances to classes in order to make it robust to outliers, namely

$$\delta = \underset{i}{\text{median}} \left(\min_k d_{ik} \right). \tag{5}$$

4. Spatial Constraint for Clustering

Clustering algorithms such as the FCV and the EM are commonly used in many different clustering problems. However, when dealing with images, it is important to preserve the spatial information contained in the inter-relations between neighboring pixels. Such an information is usually ignored by clustering algorithms and noisy segmentation results may be obtained.

In a statistical framework, this problem is addressed by using Markov Random Fields (MRF) which can model the *a priori* assumption that spatial regions are homogeneous and show a certain degree of compactness [11]. In this case, MRF are used to impose additional constraints in order to obtain homogeneous piece-wise contiguous regions. By analogy, we use the concept of MRF in order to modify the objective function of the FCV so as to preserve the spatial continuity of images. The application of the MRF concept to the FCV allows to cluster the data in the feature space but also guarantees the spatial connectivity of images.

The objective function of the FCV is modified by adding a term of neighbourhood energy and is written

$$J = J_{m,c}(U, \mathbf{v}) + \beta \sum_{j \in \eta_k} \|\mathbf{u}_j - \mathbf{u}_k\|^2, \tag{6}$$

where η_k denotes the neighbourhood of point k, and $\mathbf{u}_k$ is the vector of the degrees of belongingness of the k-th element of the data set. The second term is analogous to the contribution of the prior model in the log-likelihood function of a Maximum A Posteriori estimation, which is usually given by a Gibbs distribution.

The addition of this smoothness constraint allows to address two important aspects of motion-based segmentation: (i) the automatic estimation of the number of motions, and (ii) smooth motion labelling.

4.1. Determination of the Number of Motions

As for most of the input parameters of clustering algorithms, the number of classes c has to be specified. However, with the introduction of a smoothness constraint we can expect to determine this number automatically. Indeed, a closer look at the objective function (6) shows that it is composed of two contributions that evolve in an opposite way when the number of classes increases. The first contribution, i.e. the within group sum of squared errors, decreases with c since all distances d_{ik} can only become smaller. On the contrary, the contribution induced by the smoothness constraint increases with c since neighboring pixels are more

likely to belong to different classes. Thus, we can expect that the combination of both contributions will reach a minimum for a particular value of c.

The following steps are needed to estimate the optimal number of classes:

- Motions are estimated using the FCV by imposing a number of classes c a priori larger than the actual number of motions. As a result, some obtained motions are meaningless, and their layer of support are usually scattered on the whole image. From this point, motion parameters $\mathbf{n}_i, i = 1 \ldots c$ are kept fixed.
- Estimation of β. The "correct" number of motions may only be obtained with an appropriate choice of β. Here, we choose $\beta = \delta^2$, in order to be dimensionally consistent with the first term.
- The number of classes is determined and by adding incrementally motion classes. This is done in the following way:
  - Compute the modified objective function (6) for each class separately, and keep the class which provides the minimum value.
  - Repeat the same operation with two classes, including the best class of the previous step, and keep the best pair only if the corresponding modified objective value is smaller than the one obtained at the previous step.
  - Repeat the same operations with an increasing number of classes, as long as the modified objective value decreases.

Of course, one could also think of a dual approach, starting with all available motions and discarding meaningless classes one after the other. This approach has also been investigated, but is more costly and does not always work as well as the proposed one. This behaviour will be investigated further.

4.2. Motion labeling

At this point, we have estimates of the motion parameters. As no spatial constraint has been used so far, the motion labeling of the scene resulting from the FCV is still noisy. Smoother results can be obtained by minimizing the modified objective function (6), in the case where the class parameters, i.e. $\mathbf{n}_i, i = 1 \ldots c$, corresponding to the optimal value of c, are kept fixed. In this case, only the partition coefficients are allowed to vary. By analogy with the classical fuzzy c-variety algorithm, a closed form solution for updating the membership values can be obtained for $m = 2$ and is expressed by

$$u_{ik} = \left(\sum_{s=1}^{c} \frac{d_{ik}^2 + \alpha}{d_{sk}^2 + \alpha} \right)^{-1} \cdot \left(1 + \beta \sum_{s=1}^{c} \frac{\sum_j (\hat{u}_{ij} - \hat{u}_{sj})}{d_{sk}^2 + \alpha} \right) \quad (7)$$

where $\alpha = \beta \sum_j 1$ is a constant and $j \in \{\eta_k, k\}$. More details on this *constrained* fuzzy algorithm can be found in [6]. At this point, the labeling is still fuzzy. A hard labeling is achieved by attributing each point to the class with maximum belongingness.

5. Results

Firstly, we want to illustrate the ability of the robust fuzzy c-variety algorithm to detect a dominant motion. For that purpose, we designed a synthetic image sequence (see Figure 1) in which the upper part undergoes a superposition of a scaling and a translation, and the lower part a translation to the right. By applying the robust FCV algorithm for one class ($c = 1$), and with $m = 1.4$ (this value was used on all simulations), one obtains the motion parameters in Table 1 and the labeling displayed in Figure 1. It can be noticed that, without the addition of a spatial constraint, the labeling results are noisy. As expected, the found motion corresponds to the dominant motion, whereas the other part of the image is incorporated into the noise class.

	dominant motion	other motion	found motion
dx	0.0	1.0	$-3.410^{-}3$
dy	-0.64	0.0	-0.63
s	-10^{-2}	0.0	-910^{-3}
r	0.0	0.0	-7.610^{-5}
c_1	0.0	0.0	3.210^{-4}
c_2	0.0	0.0	-3.410^{-5}

Table 1. True and obtained motion parameters from the robust fuzzy algorithm, by asking for one class, for the synthetic image sequence with two motions. Affine parameters are: dx, dy, translation along $x-$ an $y-$ axis, s, scaling, r, rotation and c_1, c_2, shearing factors respectively.

Using the procedure described in Section 4.1, with an initial choice of 8 classes, one obtains as expected two motions. Furthermore, they are in good correspondence with

Figure 1. *Left*: **one image of the synthetic sequence with two motions.** *Right*: **layer for the dominant motion: the image shows the partition coefficients u_{i1}. Border pixels are discarded from computation. Grey levels indicate the value of u_{i1}, with black meaning 0 and white 1. Due to the small value of m (1.4), only few pixels are far from 0 or 1. Black pixels indicate points considered as outliers.**

the actual ones (see Table 2) compared to motions in Table 1. The corresponding layers without smoothing constraint and the final motion labeling obtained with the CFCV algorithm of Section 4.2 are presented in Figure 2. The small misclassified regions can be easily removed, for example with a median filtering. Such points are due to the fact that the segmentation is based only on motion information, so that pixels of homogeneous regions may be classified randomly.

	first motion	second motion
dx	$2.68\ 10^{-3}$	1.02
dy	$-6.30\ 10^{-1}$	$-2.94\ 10^{-3}$
s	$-9.90\ 10^{-3}$	$1.43\ 10^{-5}$
r	$-8.63\ 10^{-5}$	$1.45\ 10^{-4}$
c_1	$2.33\ 10^{-4}$	$-5.42\ 10^{-5}$
c_2	$3.55\ 10^{-6}$	$-1.57\ 10^{-4}$

Table 2. Motion parameters obtained from the fuzzy clustering approach with automatic determination of the number of motions.

An example of a real sequence is shown in Figure 3. In this case, the algorithm started with 18 motions and ended to estimate 3 classes. The corresponding label image, obtained with the CFCV algorithm, corresponds well to the expected motion segmentation. Improved boundaries could be obtained by incorporating luminance information (e.g. [12]), but this is not the scope of this paper.

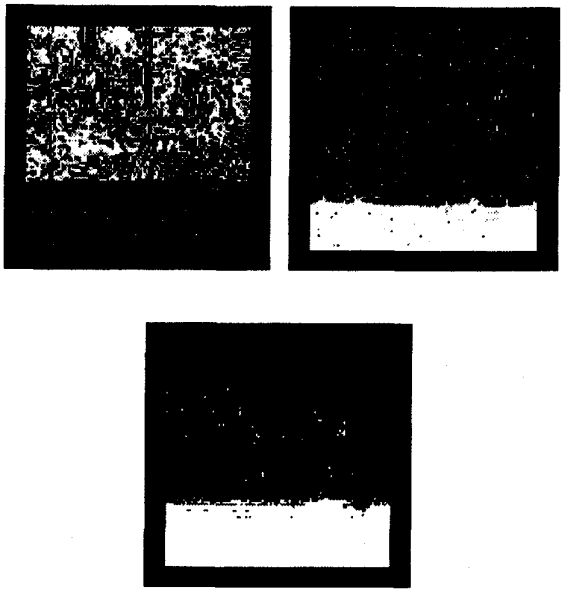

Figure 2. Synthetic motions. *Top*: **partition coefficients for both classes retained at the end.** *Bottom*: **labeling obtained from the fuzzy layers. Pixels at the boundary of the image were discarded from the computation.**

6. Conclusion

In this paper, a method for simultaneous estimations of multiple motion models and of their layers of support has been presented. The FCV algorithm was modified by adding a spatial smoothness term to the objective function, which allows to determine the number of classes automatically, and to improve the motion labeling. The estimated parameters and the motion segmentation results are encouraging.

A point that should be studied further is the comparison of this method with other studies based on Minimum Description Length (MDL, [13]) or the Akaike criterion [14].

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.

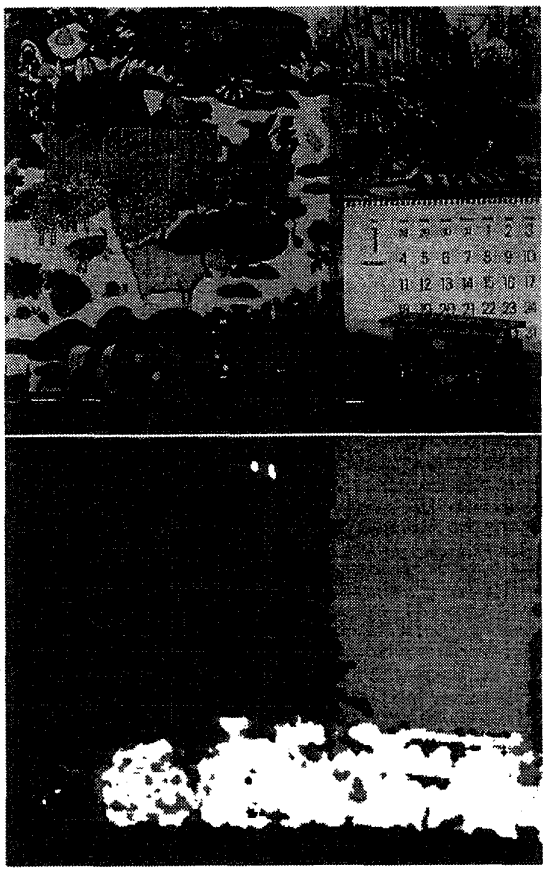

Figure 3. Motion segmentation from the "mobile and calender sequence". *Left*: image 10 of the sequence. *Right*: label image with 3 classes, after median filtering

[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 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.

[5] J.M. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models in complex image sequences. In *7th European Conference on Signal Processing*, Edinburgh, Scotland, September 1994.

[6] B. Duc, Ph. Schroeter, and J. Bigün. Motion estimation and segmentation by fuzzy clustering. In *1995 IEEE International Conference on Image Processing*, volume III, pages 472–475, Washington D.C., October 23-26 1995.

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

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

[9] 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.

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

[11] 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.

[12] P. Schroeter and S. Ayer. Multiple-frame based segmentation of moving objects by combining luminance and motion. In *Seventh European Signal Processing Conference*, pages 22–25, Edimburgh, Scottland, September 1994.

[13] J. Rissanen. A universal prior for integers and estimation by minimum description length. *The Annals of Statistics*, 11(2):416–431, 1983.

[14] H. Bozdogan and S.L. Sclove. Multi-sample cluster analysis using akaike's information criterion. *Annals of the Institute of Statistical Mathemathics*, 36:163–180, 1984.