Region-Based Tracking in an Image Sequence *

François Meyer and Patrick Bouthemy

IRISA/INRIA, Campus Universitaire de Beaulieu, 35042 Rennes Cedex, France

Abstract. This paper addresses the problem of object tracking in a sequence of monocular images. The use of regions as primitives for tracking enables to directly handle consistent object-level entities. A motion-based segmentation process based on normal flows and first order motion models provide instantaneous measurements. Shape, position and motion of each region present in such segmented images are estimated with a recursive algorithm along the sequence. Occlusion situations can be handled. We have carried out experiments on sequences of real images depicting complex outdoor scenes.

1 Introduction

Digitized time-ordered image sequences provide an actually rich support to analyze and interpret temporal events in a scene. Obviously the interpretation of dynamic scenes has to rely somehow on the analysis of displacements perceived in the image plane. During the 80's, most of the works have focused on the two-frame problem, that is recovering the structure and motion of the objects present in the scene either from the optical flow field derived between time t and time $t+1$, or from the matching of distinguished features (points, contour segments, ...) previously extracted from two successive images.

Both approaches usually suffer from different shortcomings, like intrinsic ambiguities, and above all numerical instability in case of noisy data. It is obvious that performance can be improved by considering a more distant time interval between the two considered frames (by analogy with an appropriate stereo baseline). But matching problems become then overwhelming. Therefore, an attractive solution is to take into account more than two frames and to perform tracking over time using recursive temporal filtering [1]. Tracking thus represents one of the central issues in dynamic scene analysis.

First investigations were concerned with tracking of points, [2], and contour segments, [3, 4]. However the use of vertices or edges lead to a sparse set of trajectories and can make the procedure sensitive to occlusion. The interpretation process requires to group these features into consistent entities. This task can be more easily achieved when working with a limited class of a priori known objects [5]. It appears that the ability of directly tracking complete and coherent entities should enable to more efficiently solve for occlusion problems, and also should make the further scene interpretation step easier. This paper addresses this issue. Solving it requires to deal with a dense spatio-temporal information. We have developed a new tracking method which takes into account regions as features and relies on 2D motion models.

* This work is supported by MRT (French Ministry of Research and Technology) in the context of the EUREKA European project PROMETHEUS, under PSA-contract VY/85241753/14/Z10.

2 Region Modeling, Extraction and Measurement

We want to establish and maintain the successive positions of an object in a sequence of images. Regions are used as primitives for the tracking algorithm. Throughout this paper we will use the word, "regions", to refer to connected components of points issued from a motion-based segmentation step. The region can be interpreted as the silhouette of the projection of an object in the scene, in relative motion with respect to the camera.

Previous approaches, [6], to the "region-tracking" issue generally reduce to the tracking of the center of gravity of regions. The problem of these methods is their inability to capture complex motion of objects in the image plane. Since the center of gravity of a region in the image does not correspond to the same physical point throughout the sequence, its motion does not accurately characterize the motion of the concerned region.

We proceed as follows. First the segmentation of each image is performed using a motion-based segmentation algorithm previously developed in our lab. Second the correspondence between the predicted regions and the observations supplied by the segmentation process is established. At last a recursive filter refines the prediction, and its uncertainty, to obtain the estimates of the region location and shape in the image. A new prediction is then generated for the next image.

2.1 The Motion Based Segmentation Algorithm

The algorithm is fully described in [7]. The motion-based segmentation method ensures stable motion-based partitions owing to a statistical regularization approach. This approach does not require neither explicit 3D measurements, nor the estimation of optic flow fields. It mainly relies on the spatio-temporal variations of the intensity function while making use of 2D first-order motion models. It also manages to link those partitions in time, but of course to a short-term extent.

When a moving object is occluded for a while by another object of the scene and reappears, the motion-based segmentation process may not maintain the same label for the corresponding region over time. The same problem arises when trajectories of objects cross each other. Labels before occlusion may disappear and leave place to new labels corresponding to reappearing regions after occlusion. Consequently, tracking regions over long periods of time requires a filtering procedure to be steady. A truly trajectory representation and determination is required. The segmentation process will provide only instantaneous measurements. In order to work with regions, the concept of region must be defined in some mathematical sense. We describe hereafter the region descriptor used throughout this paper.

2.2 The Region Descriptor

The region representation

We need a model to represent regions. The representation of a region is not intended to capture the exact boundary. It should give a description of the shape and location that supports the task of tracking even in presence of partial occlusion.

We choose to represent regions with some of its boundary points. The contour is sampled in such a way that it preserves shape information of the silhouette. We must select points that best capture the global shape of the region. This is achieved through a polygonal approximation of the region. A good approximation should be "close" to the original shape and have the minimum number of vertices. We use the approach developed by Wall and Danielson in [8]. A criterion controls the closeness of the shape and the polygon.

The region can be approximated accurately by this set of vertices. This representation offers the property of being flexible enough to follow the deformations of the tracked silhouette. Furthermore this representation results in a compact description which decreases the amount of data required to represent the boundary, and it yields easily tractable models to describe the dynamic evolution of the region.

Our region tracking algorithm requires the matching of the prediction and an observation. The matching is achieved more easily when dealing with convex hull. Among the boundary points approximating the silhouette of the region, we retain only those which are also the vertices of the convex hull of the considered set of points. It must be pointed out that these polygonal approximations only play a role as "internal items" in the tracking algorithm to ease the correspondence step between prediction and observation. It does not restrict the type of objects to be handled as shown in the results reported further.

The region descriptor

This descriptor is intended to represent the silhouette of the tracked region, all along the sequence. We represent the tracked region with the same number of points during successive time intervals of variable size. At the beginning of the interval we determine in the segmented image the number of points, n, necessary to represent the concerned region. We maintain this number fixed as long as the distance, defined in 2.3, between the predicted region and the observation extracted from the segmentation is not too important. The moment the distance becomes too large, the region descriptor is reset to an initial value equal to the observation. This announces the beginning of a new interval.

We can represent the region descriptor with a vector of dimension $2n$. This vector is the juxtaposition of the coordinates (x_i, y_i) of the vertices of the polygonal approximation of the region : $[x_1, y_1, x_2, y_2, \ldots, x_n, y_n]^T$.

2.3 The Measurement Vector

Measurement definition

We need a measurement of the tracked region, in each image, in order to update the prediction generated by the filter. The measurement is derived from the segmented image. For a given region we would like a measurement vector that depicts this region with the same number of points as the region descriptor. This number remains constant throughout an interval of frames. The shape of the tracked region may change. The region may be occluded. Thus the convex hull of the segmented region does not provide enough information. We will generate a more complete measurement vector related to the segmented region. The idea is illustrated in Fig. 1. If the segmentation algorithm provides us with only a partial view of the region, the "remaining part" can be inferred as follows. Let us assume that the prediction is composed of n points, and that the boundary of the region obtained by the segmentation is represented by m points, (if the silhouette of the observation is occluded we have $m \leq n$). We will move the polygon corresponding to the prediction in order to globally match it with the convex hull of the observation composed of m points. We finally select the n points of the correctly superimposed polygon onto the observation, as the measurement vector. The measurement coincides indeed with the segmented region, and if the object is partially occluded, the measurement still gives an equivalent complete view of the silhouette of the region.

Consequently this approach does not require the usual matching of specific features which is often a difficult issue. Indeed the measurement algorithm works on the region taken as a whole.

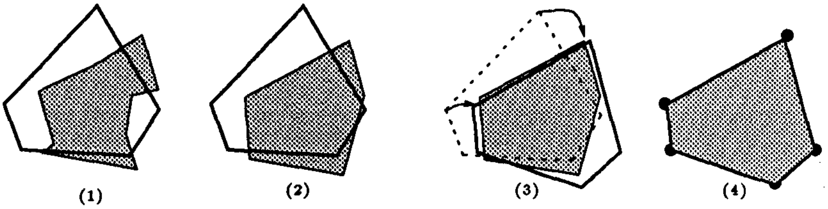

Fig. 1. The measurement algorithm : (1) Observation obtained by the segmentation (grey region), and prediction (solid line) ; (2) Convex hull of the observation ; (3) Matching of polygons ; (4) Effective measurement : vertices of the grey region.

Measurement algorithm

If we represent the convex hull of the silhouette obtained by the segmentation and the prediction vector as two polygons, the problem of superimposing the observation and the prediction reduces here to the problem of matching two convex polygons with possibly different number of vertices.

Matching is achieved by moving a polygon and finding the best translation and rotation to superimpose it on the other one. We did not include scaling in the transformation, otherwise in the case of occlusion the minimization process will scale the prediction to achieve a best matching with the occluded observation. A distance is defined on the space of shapes, [9], and we seek the geometrical transformation that minimizes the distance between the two polygons. If P_1 and P_2 are two polygons, T the transform applied on the polygon P_2, we minimize f with respect to T:

$$f(T) = m(P_1, T(P_2)) = \sum_{M_1 \in P_1} d(M_1, T(P_2))^2 + \sum_{M_2 \in P_2} d(T(M_2), P_1)^2 \qquad (1)$$

The function f is continuous, differentiable. It is also convex with respect to the two parameters of the translation. Thus conjugate-gradient methods can be used to solve the optimization problem.

3 The Region-Based Tracking Algorithm

A previous version of the region-tracking algorithm, where each vertex of the region could evolve independently from the others, with constant acceleration, is proposed in [10]. The measurement is generated by the algorithm described in Sect. 2.3. A Kalman filter gives estimates of the position of each vertex. Though the model used to describe the evolution of the region is not very accurate, we nevertheless have good results with the method. We propose hereafter a more realistic model to describe the evolution of the region. More details can be found in [10].

Our approach has some similarities with the one proposed in [11]. The authors constraint the target motion in the image plane to be a 2D affine transform. An overdetermined system allows to compute the motion parameters. However, the region representation and the segmentation step are quite different and less efficient. Besides their approach does not take into account the problems of possible occlusion, or junction of trajectories. We propose an approach with a complete model for the prediction and update of the object geometry and kinematics.

We make use of two models : a geometric model and a motion model, (Fig. 2). The geometric filter and the motion filter estimate shape, position and motion of the region from the observations produced by the segmentation. The two filters interact : the estimation of the motion parameters enables the prediction of the geometry of the region in the next frame. The shape of the region obtained by the segmentation is compared with the prediction. The parameters of the region geometry are updated. A new prediction of the shape and location of the region in the next frame is then calculated.

When there is no occlusion the segmentation process assigns a same label over time to a region ; thus the correspondence between prediction labels and observation labels is easy. If trajectories of regions cross each other, new labels corresponding to reappearing regions after occlusion will be created while labels before occlusion will disappear. In this case more complex methods must be derived to estimate the complete trajectories of the objects.

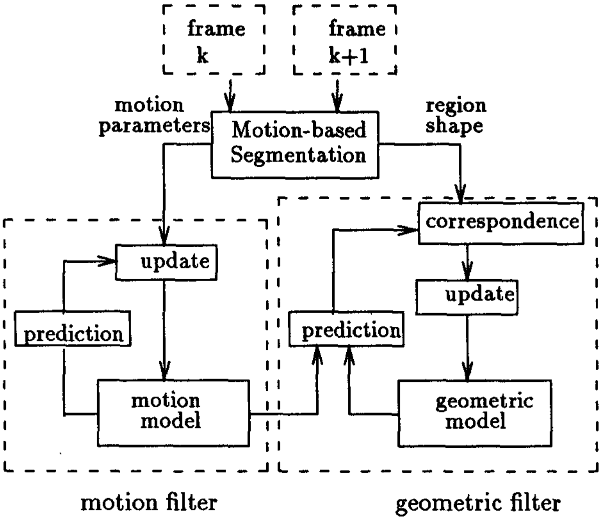

Fig. 2. The complete region-based tracking filter

3.1 The Geometric Filter

We assume that each region R_i in the image at time $t+1$ is the result of an affine transformation of the region R_i in the image at time t. Hence every point $(x(t), y(t)) \in R$

at time t will be located at $(x(t+1), y(t+1))$ at time $t+1$, with :

$$\begin{pmatrix} x \\ y \end{pmatrix}(t+1) = \Phi(t)\begin{pmatrix} x \\ y \end{pmatrix}(t) + \underline{\mathrm{b}}(t) \tag{2}$$

The affine transform has already been used to model small transformation between two images, [11]. The matrix $\Phi(t)$ and the vector $\underline{\mathrm{b}}(t)$ can be derived from the parameters of the affine model of the velocity field, calculated in the segmentation algorithm, for each region moving in the image. Let $M(t)$ and $\underline{\mathrm{u}}(\mathrm{t})$ be the parameters of the affine model of the velocity within the region R. We have :

$$\forall (x,y) \in R, \quad \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix}(t) = M(t)\begin{pmatrix} x \\ y \end{pmatrix}(t) + \underline{\mathrm{u}}(t)$$

Even if 2nd order terms generally result from the projection in the image of a rigid motion, they are sufficiently small to be neglected in such a context of tracking, which does not involve accurate reconstruction of 3D motion from 2D motion. Affine models of the velocity field have already been proposed in [12] and [13]. The following relations apply :

$$\Phi(t) = I_2 + M(t) \text{ and } \underline{\mathrm{u}}(t) = \underline{\mathrm{b}}(t) \tag{3}$$

For the n vertices $(x_1, y_1), \ldots, (x_n, y_n)$ of the region descriptor we obtain the following system model :

$$\begin{pmatrix} x_1 \\ y_1 \\ \vdots \\ x_n \\ y_n \end{pmatrix}(t+1) = \begin{bmatrix} [\Phi(t)] & & 0 \\ & \ddots & \\ 0 & & [\Phi(t)] \end{bmatrix}\begin{pmatrix} x_1 \\ y_1 \\ \vdots \\ x_n \\ y_n \end{pmatrix}(t) + \begin{bmatrix} I_2 \\ \vdots \\ \vdots \\ I_2 \end{bmatrix}\underline{\mathrm{b}}(t) + \begin{bmatrix} \zeta_1 \\ \vdots \\ \vdots \\ \zeta_n \end{bmatrix}(t)$$

where I_2 is the 2×2 identity matrix. $\Phi(t)$ and $\underline{\mathrm{b}}(t)$ have been defined above in (3). $\zeta_i = [\zeta_i^x, \zeta_i^y]^T$ is a two dimensional, zero mean Gaussian noise vector. We choose a simplified model of the noise covariance matrix. We will assume that :

$$cov(\zeta_1, \ldots, \zeta_n) = \sigma_\zeta^2 I_{2n}$$

where I_{2n} is the $2n \times 2n$ identity matrix. This assumption enables us to break the filter of dimension $2n$ into n filters of dimension 2.

The matrix $\Phi(t)$ and the vector $\underline{\mathrm{b}}(t)$ accounts for the displacements of all the points within the region, between t and $t+1$. Therefore the equation captures the global deformation of the region. Even though each vertex is tracked independently, the system model provides a "region-level" representation of the evolution of the points.

For each tracked vertex the measurement is given by the position of the vertex in the segmented image. The measurement process generates the measurement as explained in Sect. 2.3.

The following system describes the dynamic evolution of each vertex (x_i, y_i) of the region descriptor of the tracked region. Let $\underline{\mathrm{s}}(t) = [x_i, y_i]^T$ be the state vector, and $\underline{\mathrm{m}}(t)$ the measurement vector which contains the coordinates of the measured vertex,

$$\begin{cases} \underline{\mathrm{s}}(t+1) = \Phi(t)\underline{\mathrm{s}}(t) + \underline{\mathrm{b}}(t) + \zeta(t) \\ \quad \underline{\mathrm{m}}(t) \;\; = \;\; \underline{\mathrm{s}}(t) \qquad\qquad\quad + \nu(t) \end{cases} \tag{4}$$

$\zeta(t)$ and $\nu(t)$ are two sequences of zero-mean Gaussian white noise. $\underline{\mathrm{b}}(t)$ is interpreted as a deterministic input. $\Phi(t)$ is the matrix of the affine transform. We assume that the above linear dynamic system is sufficiently accurate to model the motion of the region in the image. We want to estimate the vector $\underline{s}(t)$ from the measurement $\underline{m}(t)$. The Kalman filter [14] provides the optimal linear estimate of the unknown state vector from the measurements, in the sense that it minimizes the mean square estimation error and by choosing the optimal weight matrix gives a minimum unbiased variance estimate. We use a standard Kalman filter to generate recursive estimates $\hat{\underline{s}}(t)$.

The first measurement is taken as the initial value of the estimate, Hence we have $\hat{\underline{s}}(0) = \underline{m}(0)$. The covariance matrix of the initial estimate is set to a diagonal matrix with very large coefficients. This expresses our lack of confidence in this first value.

3.2 The Kinematic Filter

The attributes of the kinematic model are the six parameters of the 1st order approximation of the velocity field. These variables are determined with a least-squares regression method. Therefore these instantaneous measurements are corrupted by noise and we need a recursive estimator to convert observation data into accurate estimates. We use a Kalman filter to perform this task. We work with the equivalent decomposition :

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \frac{1}{2} \begin{pmatrix} div + hyp1 & hyp2 - rot \\ rot + hyp2 & div - hyp1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a \\ b \end{pmatrix}$$

This formulation has the advantage that the variables *div*, *rot*, *hyp*1 and *hyp*2 correspond to four particular vector fields that can be easily interpretated, [7].

The measurement is given by the least square estimates of the six variables. We have observed on many sequences that the correlation coefficients between the six estimates are negligible. For this reason, we have decided to decouple the six variables. The advantage is that we work with six separate filters.

In the absence, in the general case, of any explicit simple analytical function describing the evolution of the variables, we use a Taylor-series expansion of each function about t. After having experimented with different approximations, it appears that using the first three terms performs a good tradeoff between the complexity of the filter and the accuracy of the estimates. Let $\underline{\theta}(t) = [\alpha(t), \dot{\alpha}(t), \ddot{\alpha}(t)]^T$ be the state vector, where α is any of the six variables : a, b, *div*, *rot*, *hyp*1 and *hyp*2. $z(t)$ is the measurement variable. We derive the following linear dynamic system :

$$\begin{cases} \underline{\theta}(t+1) = A\underline{\theta}(t) + \xi(t) \\ z(t) = C(t)\underline{\theta}(t) + \eta(t) \end{cases} \text{with } A = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \quad C = [1 \ 0 \ 0] \quad Q = \sigma_Q^2 \begin{bmatrix} \frac{1}{36} & \frac{1}{12} & \frac{1}{6} \\ \frac{1}{12} & \frac{1}{4} & \frac{1}{2} \\ \frac{1}{6} & \frac{1}{2} & 1 \end{bmatrix}$$

$\xi(t)$ and $\eta(t)$ are two sequences of zero-mean Gaussian white noises of covariance matrix Q, and variance σ_η^2 respectively.

3.3 Results

We present in Fig. 3 the results of an experiment done on a sequence of real images. The polygons representing the tracked regions are superimposed onto the original pictures at time t_1, t_9, and t_{12}. The corresponding segmented pictures at the same instants are presented on the right. The scene takes place at a crossroad. A white van is comming from the left of the picture and going to the right (Fig. 3a). A black car is driving behind the van so closely that the segmentation is enable to split the two objects (Fig. 3d). A white car is comming from the opposite side and going left. The algorithm accurately tracks the white car, even at the end of the sequence where the car almost disappears behind the van (Fig. 3e and f). Since the segmentation process delivers a single global region for the van and the black car (Fig. 3d), the filter follows this global region. Thus the tracked region does not correspond exactly to the boundary of the van. This example illustrates the good performance of the region-based tracking in the presence of occlusion. An improved version of the method, where the kinematics parameters are estimated using a multiresolution approach is being tested. More experiments are presented in [10].

4 Conclusion

This paper has explored an original approach to the issue of tracking objects in a sequence of monocular images. We have presented a new region-based tracking method which delivers dense trajectory maps. It allows to directly handle entities at an "object-level". It exploits the output of a motion-based segmentation. This algorithm relies on two interacting filters : a geometric filter which predicts and updates the region position and shape, and a motion filter which gives a recursive estimation of the motion parameters of the region. Experiments have been carried out on real images to validate the performance of the method. The promising results obtained indicate the strength of the "region approach" to the problem of tracking objects in sequences of images.

References

1. T.J. Broida, R. Chellappa. Estimating the kinematics and structure of a rigid object from a sequence of monocular images. *IEEE Trans. PAMI*, Vol.13, No.6:pp 497–513, June. 1991.
2. I. K. Sethi and R. Jain. Finding Trajectories of Feature Points in a Monocular Image Sequence. *IEEE Trans. PAMI*, Vol. PAMI-9, No 1:pp 56–73, January 1987.
3. J.L. Crowley , P. Stelmaszyk, C. Discours. Measuring Image Flow by Tracking Edge-Lines. *Proc. 2nd Int. Conf. Computer Vision, Tarpon Springs, Florida*, pp 658–664, Dec. 1988.
4. R. Deriche, O. Faugeras. Tracking Line Segments. *Proc. 1st European Conf. on Computer Vision, Antibes*, pp 259–268, April 1990.
5. J. Schick, E.D. Dickmanns. Simultaneous estimation of 3d shape and motion of objects by computer vision. *Proceedings of the IEEE Workshop on Visual Motion, Princeton New-Jersey*, pp 256–261, October 1991.
6. G. L. Gordon. On the tracking of featureless objects with occlusion. *Proc. Workshop on Visual Motion, Irving California*, pp 13–20, March 1989.
7. E. François, P. Bouthemy. Multiframe-based identification of mobile components of a scene with a moving camera. *Proc. CVPR, Hawaii*, pp 166–172, June 1991.
8. Karin Wall and Per-Erik Danielsson. A fast sequential method for polygonal approximation od digitized curves. *Computer Vision, Graphics and Image Processing*, 28:pp 220–227, 1984.
9. P. Cox, H. Maitre, M. Minoux, C. Ribeiro. Optimal Matching of Convex Polygons. *Pattern Recognition Letters*, Vol 9 No 5:pp 327–334, June 1989.
10. F. Meyer, P. Bouthemy. Region-based tracking in an image sequence. *Research Report in preparation, IRISA/INRIA Rennes*, 1992.
11. R.J. Schalkoff, E.S. McVey. A model and tracking algorithm for a class of video targets. *IEEE Trans. PAMI*, Vol.PAMI-4, No.1:pp 2–10, Jan. 1982.
12. P.J. Burt, J. R. Bergen, R. Hingorani, R. Kolczynski, W.A. Lee, A. Leung, J. Lubin, H. Shvaytser. Object tracking with a moving camera. *IEEE Workshop on Visual Motion*, pp 2–12, March 1989.
13. G. Adiv. Determining three-dimensional motion and structure from optical flow generated by several moving objects. *IEEE Trans. PAMI*, Vol 7:pp 384–401, July 1985.
14. Arthur Gelb. *Applied Optimal Estimation.* MIT Press, 1974.

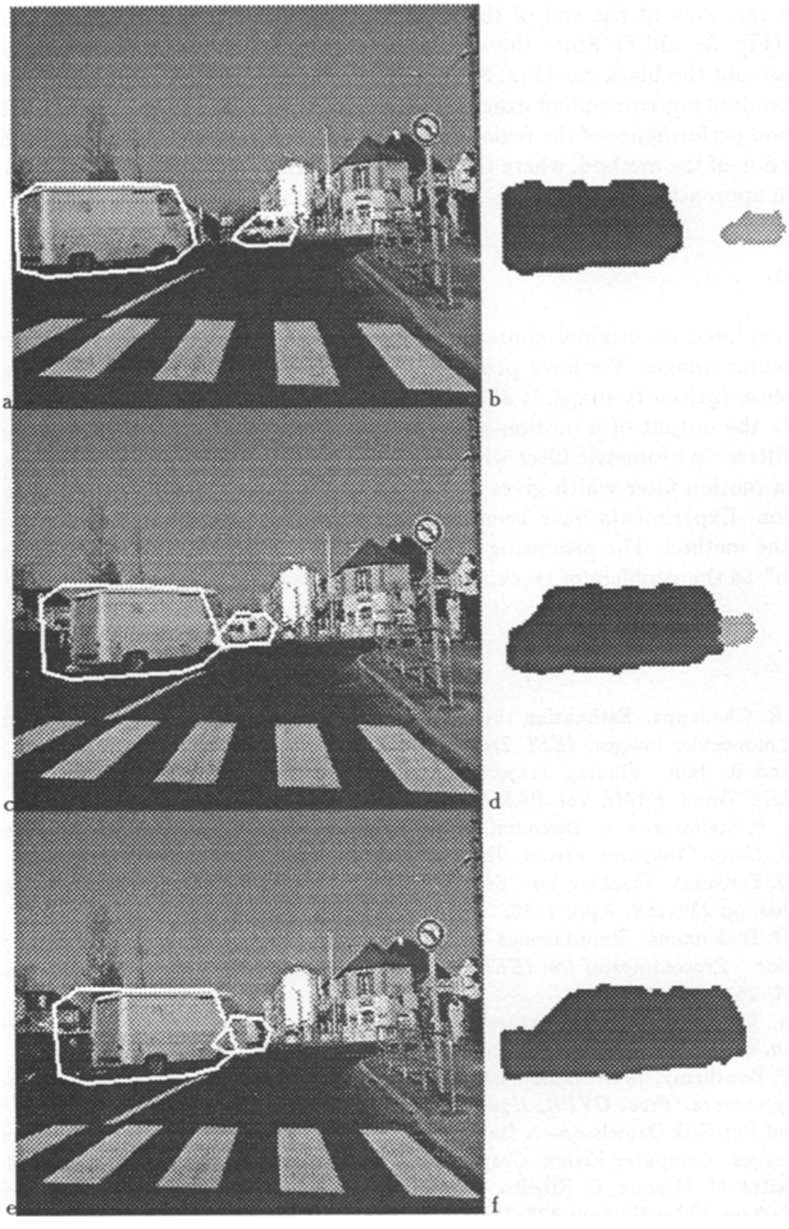

Fig. 3. Left : original images at time t_1, t_9, t_{12} with tracked regions. Right : segmented images at the same instants