HAL open science

# A PDE-based Level-Set Approach for Detection and Tracking of Moving Objects

Nikolaos Paragios, Rachid Deriche

▶ To cite this version:

Nikolaos Paragios, Rachid Deriche. A PDE-based Level-Set Approach for Detection and Tracking of Moving Objects. RR-3173, INRIA. 1997. inria-00073515

HAL Id: inria-00073515
https://inria.hal.science/inria-00073515

Submitted on 24 May 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

# *A PDE-based Level-Set Approach for Detection and Tracking of Moving Objects*

Nikolaos PARAGIOS and Rachid DERICHE

**N° 3173**

Mai 1997

THÈME 3

Rapport de recherche

ISSN 0249-6399

# A PDE-based Level-Set Approach for Detection and Tracking of Moving Objects

Nikolaos PARAGIOS and Rachid DERICHE

Thème 3 — Interaction homme-machine,
images, données, connaissances
Projet Robotvis

Rapport de recherche n ° 3173 — Mai 1997 — 29 pages

**Abstract:** This papers presents a framework for detecting and tracking moving objects in a sequence of images. Using a statistical approach, where the *inter-frame* difference is modeled by a mixture of two Laplacian or Gaussian distributions, and an energy minimization based approach, we reformulate the motion detection and tracking problem as a front propagation problem. The Euler-Lagrange equation of the designed energy functional is first derived and the flow minimizing the energy is then obtained. Following the work by Caselles et al [11] and Malladi et al [23, 24] the contours to be detected and tracked are modeled as geodesic active contours evolving toward the minimum of the designed energy, under the influence of internal and external image dependent forces. Using the level set formulation scheme of Osher and Sethian [29], complex curves can be detected and tracked and topological changes for the evolving curves are naturally managed. To reduce the computational cost required by a direct implementation of the formulation scheme of Osher and Sethian [29], a new approach exploiting aspects from the classical Narrow Band [3] and Fast Marching [33] methods is proposed and favorably compared to them. In order to further reduce the CPU time, a multi-scale approach has also been considered. Very promising experimental results are provided using real video sequences.

**Key-words:** Motion detection, Tracking, Front propagation, Level-set approach, Curve evolution, Deformable contours, Geodesic Active contours, PDE.

*(Résumé : tsvp)*

**This work was funded in part under the VIRGO research network (EC Contract No ERBFMRX-CT96-0049) of the TMR Programme.**

Unité de recherche INRIA Sophia Antipolis
2004 route des Lucioles, BP 93, 06902 SOPHIA ANTIPOLIS Cedex (France)
Téléphone : 04 93 65 77 77 – Télécopie : 04 93 65 77 65

# EDP et Courbes de Niveau pour la Detection et le Suivi d' Objets en Mouvement

**Résumé :** Ce rapport décrit une nouvelle méthode de résolution du problème de la détection et du suivi d'objets en mouvement dans une séquence monoculaire d'images. Une approche variationnelle permettant de reformuler le problème de la détection et du suivi comme un problème de propagation de front est à la base de cette nouvelle méthode. Une énergie à minimiser est associée au principe variationnel que doivent respecter les contours des objets en mouvement. Suivant les travaux de Caselles et al [11] et de Malladi et al [23, 24], l'équation d'Euler-Lagrange, déduite de la minimisation de cette énergie, est alors utilisée afin de déformer les contours initiaux, considérés comme des contours actifs géodésiques qui vont se déplacer vers les objets en mouvement. La résolution de l'EDP par la méthode des courbes de niveau d'Osher et Sethian [29], permet ensuite de mettre en oeuvre de manière efficace le processus d'évolution des contours tout en gérant automatiquement d'éventiuels problèmes de changement de topologie durant la déformation. Ce qui permet de traiter correctement les configurations singuliéres de type fusion d'objets multiples et/ou scission d'objets en plusieurs parts, qui peuvent apparaitre au cours du processus de suivi.

Afin de réduire la complexité calculatoire d'une mise en oeuvre directe, une nouvelle méthode exploitant les aspects les plus positifs des approches rapides connues sous le nom de *Narrow Band* [3] et *Fast Marching* [33] est proposée mise en oeuvre et comparée favorablement á ces deux techniques. Une approche multi-échelles est aussi considérée.

Plusieurs résultats expérimentaux, obtenues à partir de séquences d'images réelles, illustrent les très bonnes performances obtenues par cette nouvelle méthode.

**Mots-clé :** Détection et Suivi d'Objets en Mouvement Approches Variationnelles, Contours Actifs Géodésique, Évolutions de Courbes, Courbes de Niveau, EDP.

# Summary


- **What is the original contribution of this work?**

  Using the curve evolution theory, **we reformulate the motion detection and tracking problem as a front propagation problem**. An energy minimization based approach is proposed for the solution. The Euler-Lagrange equation of the proposed energy is derived, and its associated PDE is then solved using the level set formulation scheme of Osher and Sethian [29] by viewing it as a front propagating with internal and external image dependent speed. **The original scheme we propose may be viewed as a geodesic active motion detection and tracking model which basically attract the given curves to the bottom of a potential well corresponding to the frontiers of the moving objects**. The curves to be tracked are therefore modeled as geodesic active contours evolving toward the minimum of the designed functional.

  Due to the computational cost required by a direct implementation of the formulation scheme of Osher and Sethian [29], **a new approach combining aspects of the Narrow Band [3] and Fast Marching [33] methods is proposed and favorably compared to these two classical methods.** In order to reduce further the CPU time, **a multi-scale approach has also been considered** which is used according to the coarse to fine strategy. Very promising experimental results are provided using real video sequences.

- **Why should this contribution be considered important?**

  The pioneering work of Stanley Osher and J.A. Sethian in developing level set methods for capturing fronts, has been successfully applied to a wide range of applications, including problems in fluid mechanics, combustion, computer animation, image processing, robotic navigation etc. More recently these techniques started to be applied in the computer vision domain including shape from shading, geodesic active contours, skeletons, color scale space, etc. **We show in this article that the level set method can also be applied effectively for solving important vision problems such as motion detection and tracking and we provide promising results. Using the method we propose, complex curves can be tracked and topological changes for the evolving curves are naturally managed. The final result is relatively independent of the curve initialization.**

  **Using our original implementation algorithm, the computational cost is drastically reduced.**

  **The problem addressed is very important from the point of view of the applications.**

Motion detection and Objects Tracking are among the most important issues within a wide class of applications including robotics, automated surveillance, traffic monitoring, medical imagery, etc.

Motion detection and Objects Tracking provide a good basis for the estimation of 3D motion and/or structure from time varying images and thus, provide a rich support to the analysis of time-varying environment. Due to the detailed quantitative information they can provide, and due to the fact that the change of topology is correctly handled, the tracked contours are very well suited to tracking rigid and **non-rigid objects.**

- **What is the most closely related work by others and how does this work differ ?**

  Simultaneously to this work, the idea of applying the curve evolution theory to the tracking problem has been recently presented by Caselles in [10]. However this sequentially three step approach is very different from the unified approch we present in this article. Following their previous work on geodesic active contours, they first start by detecting the contours of the objects to be tracked. An estimation of the velocity vector field along the detected contours is then performed using a completely separate approach, and finally another PDE is designed to move the contours to the boundary of the moving objects. These contours are then used as initial estimate of the contours in the next image, and the process is repeated. Our method starts by expressing an energy functional that include both terms of motion detection and tracking. The associated Euler-Lagrange PDE is then solved using the level set formulation scheme of Osher and Sethian [29]. In order to reduce the CPU time, we have developed and implemented **a new approach that exploits the best aspects of the Narrow Band [3] and Fast Marching [33] methods** . Various real world video sequences have been used to test and validate our approach.

- **How can other researchers make use of the results of this work ?**

  **This work will be of great interest for researchers working on motion analysis, and tracking and curve evolution theory.**

  **The ideas presented in this paper can also be applied to develop different applications such as stereo, shape from shading,..** We are currently investigating the application to the stereo problem; the preliminary results obtained are very promising.

# 1 Introduction

*Detection* and *tracking* of moving objects in a sequence of images are problems arising in numerous applications of computer vision and image coding. This paper deals with these two problems of motion analysis supposing a static scene. That is, we assume that the images are taken by a static observer (camera).

During the last decade, many different approaches have been proposed for the *detection* of moving objects. In the related bibliography, many different frameworks have been proposed concerning the problem formalization. A basic assumption is the absence of camera's motion; otherwise, the problem is more complicated, since both the estimation and compensation of this motion are required.

The simplest moving object detector is obtained using a thresholding technique over the *inter-frame difference*. Decisions are taken independently point by point [16], or over blocks in order to achieve robustness in noise influence [38]. More complex approaches lead to a statistical orientation-formalization of the problem. In this case, the *inter-frame* difference is analyzed as a mixture decomposition problem [37]. Statistical likelihood tests have been also considered [27]. This approach is quite efficient in the case where large displacements appear, or the objects projections are sufficiently texture. Kalman filtering techniques [20], as well as the use of first and second order Markov chains have been applied [9, 17]. In addition Gabor filters have been also designed in order to create spatio-temporal gradient detectors [22].

The spatial Markov Random Fields (MRFs) have been widely used for the problem modelization. The problem is statistically oriented, according to the equivalence between the MRFs and Gibbs distribution [19]. The problem solution is derived by minimizing a global cost function, using global or local minimization techniques [2, 8, 20, 35]. There are some models which propose a more flexible and less expensive (in terms of complexity) way to deal with the problem, by combining the MRFs with thresholding techniques [2]. Deterministic relaxation algorithms often have been also used [28, 30].

The majority of the related bibliography assumes a static observer; however there are models which are able to cope with a mobile camera by estimating the motion fields, which can be very complex due to motion and depth discontinuities. Moving objects are detected by computing the dominant motion, and by performing motion compensation using this dominant motion. Usually an affine model is used in order to describe this motion [28, 30]. Finally, in order to create a detection map in "real time", multi-scale algorithms have been also proposed [28].

Motion analysis usually includes two elementary processes: measurement and segmentation. Early tracking approaches worked with single points and edge-line segments [14], but these models were very sensitive to occlusions. As a consequence, more complex tracking models were proposed to take into account global primitives such as contours and regions. In order to achieve this, deformable models ("*snakes*") have been widely used [36]. Different types of deformable models have been proposed. Most of them are edge-based [12, 21, 39]. Other approaches use an additional constraint for the contour displacement to optimize the deformable contour [5], while other take into account the measurements of the optical flow in an integrated model [7]. On the other hand, there is a class of approaches optimizing the region contour by maximizing the intensities homogeneity inside the region [13]. Finally, there are models that combine deformable regions and deformable contours [4, 26] and others that combine snake-based tracking approaches with motion-based region segmentation [6].

Simultaneously to this work, the idea of applying the curve evolution theory to the tracking problem has been recently presented by Caselles and Coll in [10]. However this sequentially three step approach is very different from the unified approch we present in this article. Following their previous work on geodesic active contours, they first start by detecting the contours of the objects to be tracked. An estimation of the velocity vector field along the detected contours is then performed using a completely separate approach, and finally another PDE is designed to move the contours to the boundary of the moving objects. These contours are then used as initial estimate of the contours in the next image, and the process is repeated.

This paper describes a unified approach for the *detection* and *tracking* of moving objects. Both problems are stated under a unified model which follows a level-set methodology. The inter-frame difference is modeled as a mixture of two Laplacians or Gaussians distributions. Using this analysis we generate a new image, based on the input images which exhibit large gradient values only around the moving area. The use of geodesic active contours is then introduced and a unified model is proposed for both problems. For the front propagation problem three well-known schemes are used, Classic, Narrow Band and the Fast Marching approaches. A new scheme is proposed, called Hermes, which is also evaluated compared to the existing schemes. In order to achieve a faster algorithm, the front propagation methods are combined with a classic multi-scale approach.

The paper is organized as follows: Section 2 introduces the front propagation problem with the level-set formulation model and the active contour models. Section 3 illustrates the detection and the tracking approach, proposing a unified model for both problems. The existing front-propagation methods are briefly described and evaluated in comparison with our approach in Section 4. Finally, Section 5 presents experimental results, and concluding remarks.

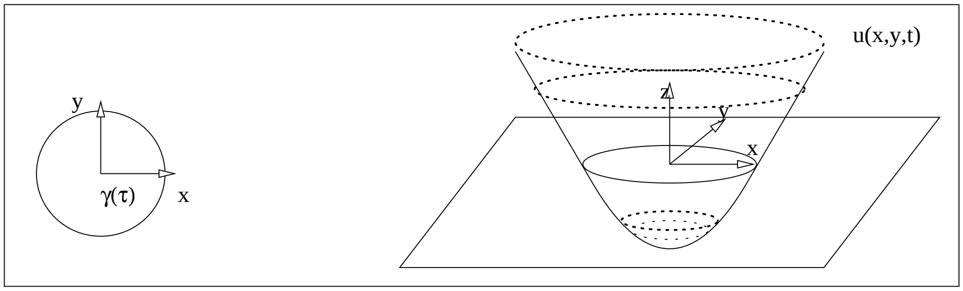



Figure 1: A circle $\gamma(x,y,t)$ and its associated surface $u(x,y,t)$

# 2 Front Propagation Problem - Active Contours

## 2.1 Front Propagation Problem

Let $C_0(p) : [0,1] \rightsquigarrow R^2$ be a close initial parameterized planar curve in a Euclidean plane, and $C(p,t)$ the family of curves which is generated by the movement of $C_0(p)$ in the direction of its inward Euclidean normal vector $\mathcal{N}$. We suppose that the speed of this movement is a scalar function of the curvature $\mathcal{K}$:

$$\left\{ \begin{array}{l} \frac{dC}{dt} = F(\mathcal{K}) \cdot \mathcal{N} \\ C(p,0) = C_0(p) \end{array} \right. \tag{1}$$

In order to implement the curve evolution, according to the above equations we can consider a Lagrangian approach and produce the associated equations of motions for the position vector $(x,y) = C(p_1)$. These positions are updated using a difference approximation scheme. The main drawback of this approach is that the evolving model is not capable to deal with topological changes of the moving front. This could be avoided by introducing the work of Osher and Sethian [29]. According to them, the initial curve $C_0(p)$ is represented by a zero-level set $(u = 0)$ of an initial surface $z$ $(z = u(x,y,t) = 0 \in R^3)$. Using equation (1) and deriving $u(x,y,t) = 0$ with respect to time and space, the following associated equation of motion for the surface $u$ can then easily de derived:

$$\left\{ \begin{array}{l} \frac{du(x,y,t)}{dt} = F(\mathcal{K}) \cdot |\nabla u| \\ u(x,y,0) = u_0(x,y) \end{array} \right. \tag{2}$$

where $|\nabla u|$ denotes the magnitude of the gradient.

There is a connection between the family of moving curves $C(p,t)$ and the family of one parameter moving surfaces $u(x,y,t)$. This connection is achieved by the fact that the level set $u = 0$ yields always the moving front. This contour representation allows to deal with changes of topology of the initial curve [29].

## 2.2 Active Contours

Let $I : [0,a] \times [0,b] \rightsquigarrow R^+$ be a given single input image from which we want to extract the contours of some objects. A classical simplified *snake model* [11], for a given parameter $\lambda$, aims at finding the curve $C(p)$ that minimizes the following energy:

$$E(C(p)) = (1-\lambda)\underbrace{\int_0^1 |C'(p)|^2 dp}_{E_{internal}(C)} + \lambda \underbrace{\int_0^1 g^2(|\nabla I(C(p))|)dp}_{E_{image}(C)} \tag{3}$$

where $\lambda \in [0,1]$ is a positive constant which balances the contribution of the two energy terms. The energy term $E_{internal}(C)$ accounts for the expected spatial properties (*i.e. smoothness*) of the contour while the energy term $E_{image}(C)$ stands for the *attraction* energy term of the curve towards the objects contour. Finally, $g$ is a monotonically decreasing function such that $g(r) \rightsquigarrow 0$ as $r \rightsquigarrow \infty$ and $g(0) = 1$.

Using the snake equation, it can be proved [29] that its minimization leads to a geodesic curve in a Riemmanian space with a new metric [11], and the problem can be shown to be equivalent to the minimization of:

$$E(C(p)) = \int_0^L g(|\nabla I(C(p))|)ds \tag{4}$$

where $ds$ is the Euclidean arc-length element, and $L$ the Euclidean length of $C(p)$. In other words, when we try to detect an object, we try to find the best minimal length that takes into account the image characteristics.

We minimize this new energy form by solving the associated Euler-Lagrange equation. According to this equation, the flow that deforms the initial curve $C(p,0) = C_0(p)$ towards the local minima of (4) is given by a steady state solution of:

$$\frac{dC(p,t)}{dt} = (g(|\nabla I(C(p))|)\mathcal{K}(p,t) - \nabla g(|\nabla I(C(p))|) \cdot \mathcal{N}(p,t))\,\mathcal{N}(p,t) \tag{5}$$

where $t$ denotes the time as the contour evolves, $\mathcal{N}(p,t)$ the inward Euclidean normal vector to the curve $C(p,t)$, and $\mathcal{K}(p,t)$ the Euclidean curvature [29]. Using the level-set formulation, we can assume that $C$ is a level-set of a function $u : [0,a] \times [0,b] \rightsquigarrow \mathbf{R}$. This model is parameter free, as well as topology-free since different topologies of zero level-set correspond to the same topology of $u$ [29]. Based on (5), it can be proved that the steady state solution of this geodesic problem (as described in [34]) is equivalent to:

$$\left\{ \begin{array}{rcl} \frac{du}{dt} & = & g(|\nabla I(C(p,t))|)\mathcal{K}(p,t)|\nabla u| + \nabla g(|\nabla I(C(p,t))|) \cdot \nabla u \\ u(x,y,0) & = & u_0(x,y) \end{array} \right. \tag{6}$$

where $C(p,t)$ is represented by a level-set of $u$ and the value of $\mathcal{K}(p,t)$ comes from the level-sets of $u$,

$$\mathcal{K}(p,t) = \mathbf{div}\left(\frac{\nabla u}{|\nabla u|}\right) \tag{7}$$

In our case $g$ is considered to be: $g(r) = \frac{1}{1+|r|^p}$, $p = 1,2$. Different forms can also be used. Following our previous work in [15], we are in the process to test different functions.

The main difference between the classical snake models and the curve evolution model is the independence of the topology due to the level-set representation. This allows detection of all the objects which appear in the image plane, without knowing their exact number. Additionally, we don't have any problems concerning the contour initialization, due to the fact that the two terms of the velocity magnitude are both equal to zero only at the contours points.

# 3 Detection and Tracking

## 3.1 Defining the model

Let $D = \{d(x,y), (x,y) \in [0, lines] \times [0, columns]\}$ denote the *inter-frame* gray level difference image with

$$d(x,y) = I(x,y;t+1) - I(x,y;t) \tag{8}$$

The detection problem consists of a "binary" label $\omega(x,y)$ for each pixel on the image grid. We associate the random field $\omega(x,y)$ with two possible events, $\omega(x,y) = static$ (*static: background pixel*), if the observed difference $d(x,y)$ supports the hypothesis for static pixel ($H_0$), and $\omega(x,y) = mobile$ (*Mobile: moving pixel*), if the observed difference supports the alternative hypothesis $H_1$, for mobile pixel. Let $p_{D|static}(d|static)$ (resp. $p_{D|mobile}(d|mobile)$) be the probability density function of the observed inter-frame difference under the $H_0$ (resp. $H_1$) hypothesis. These probability density functions are supposed to be homogeneous, *i.e.* independent of the pixel location and usually they follow the Laplacian or Gaussian law [31], that is:

$$p(d) = \frac{\lambda}{2} e^{-\lambda|d|} \quad , \quad p(d) = \frac{1}{2\sigma\sqrt{\pi}} e^{-\frac{d^2}{2\sigma^2}} \tag{9}$$

Let $P_{static}$ ($P_{mobile}$) be the *a priori* probability of hypothesis $H_0$ ($H_1$). Observed difference values are assumed to be obtained by selecting a label $l \in \{static, mobile\}$ with probability $P_l$, and then selecting a value $d$ according to the probability law $p(d|l)$. Thus, the probability density function is given by

$$p_D(d) = P_{static}\ p_{D|static}(d|static) + P_{mobile}\ p_{D|mobile}(d|mobile) \tag{10}$$

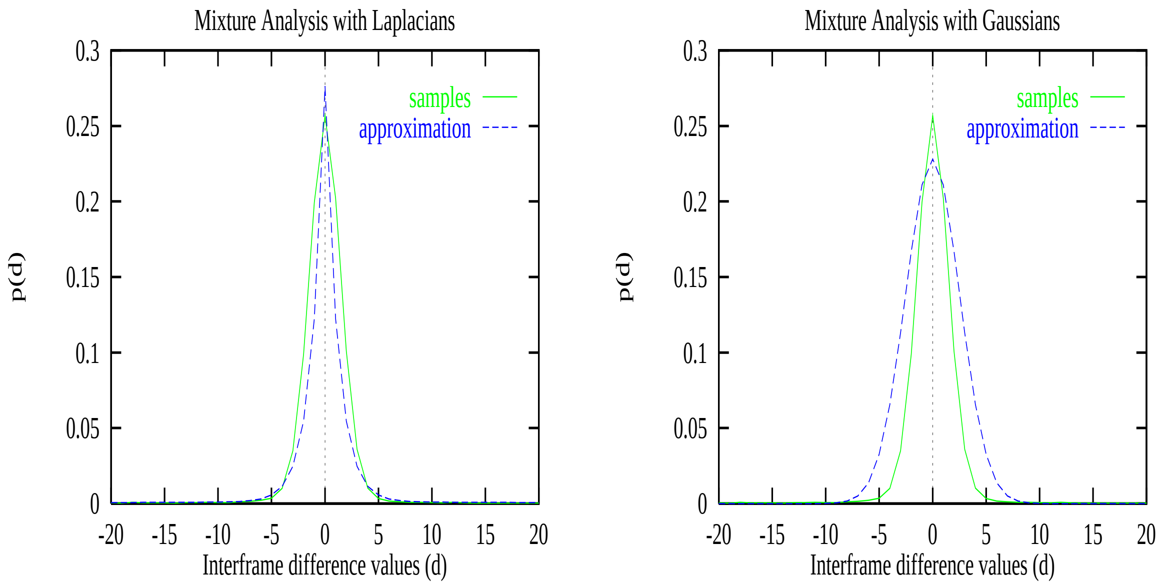



Figure 2: Mixture decomposition for inter-frame difference for *Football match sequence*

In this mixture distribution the principle of Maximum Likelihood is used to obtain an estimation of unknown parameters [18, 25], which are iteratively estimated using the observed distribution of grey level inter-frame differences (fig. 2). An initial estimation in a closed form is calculated using first, second and third order moments of the variable considered [31].

The contour of the *moving area* corresponds to the pixels where there is a **transition** between their labels and the labels of their neighbors. A pixel $(x, y)$ (a site at the image grid) defines the contour of the moving area if its label is **static** (**mobile**) and there is at least a neighborhood pixel $(v, w)$ with the label **mobile** (**static**). For each pixel location $(x, y)$ and a neighborhood pixel $(v, w)$, we define the following energy terms:

$$\begin{aligned} E_{trans}((x,y),(v,w)) &= p(d(x,y)|\mathbf{static}) \cdot p(d(v,w)|\mathbf{mobile}) \\ &\quad + p(d(x,y)|\mathbf{mobile}) \cdot p(d(v,w)|\mathbf{static}) \\ E_{smooth}((x,y),(v,w)) &= p(d(x,y)|\mathbf{static}) \cdot p(d(v,w)|\mathbf{static}) \\ &\quad + p(d(x,y)|\mathbf{mobile}) p(d(v,w)|\mathbf{mobile}) \end{aligned} \tag{11}$$

## 3.2 Detection Part

We apply the Level-Set method for the *detection of moving objects*, by creating a new image based on the input images $I_t, I_{t+1}$. The new image values should have large gradient values

| (a) | (b) |
|---|---|

Figure 3: Generated image for: (a) *Car sequence* (fig. 12(a)), (b) *Autoroute Sequence* (fig. 12(c))

in the boundaries of the moving area. In order to achieve this, we define for each pixel the following measurement:

$$I_{detection}(x,y) = \max_{(v,w)\in n_{g_{(x,y)}}} \left\{ \frac{E_{trans}((x,y),(v,w))}{E_{smooth}((x,y),(v,w))} \right\} \tag{12}$$

where $n_{g_{(x,y)}}$ denotes the neighborhood of pixel $(x,y)$ (second-order with 8 pixels). It is clear from the above definition that the generated image would present large magnitudes only in the pixels around the boundary of the moving area. To avoid the noise influence, we can perform a smoothing operation before the estimation of this image (fig. 3). For instance a median filtering technique could be applied in the image of the *inter-frame difference*. The form of the generated image (12) is empirically selected according to the quality of the obtained results. Many different forms have been tested ($i.e\ I_{detection}(x,y) = \frac{p(d(x,y)|\mathbf{mobile})}{p(d(x,y)|\mathbf{static})}$) but the selected one seems to be the more stable. The goal is to create an image with large gradient values around the moving area. With the new image, we proceed as follows: An initial arbitrary curve is defined at the borders of it. The problem is expressed using the framework of energy minimization, under the conditions of a geodesic active problem; we associate an energy function to the given curve, and we try to find (for a given parameter $\lambda$) the curve $C(p,t)$ that minimizes the following energy:

$$E(C(p)) = \underbrace{(1-\lambda)\int_0^1 |C'(p)|^2 dp}_{E_{internal}(C)} + \underbrace{\lambda \int_0^1 g^2(|\nabla I_{detection}(C(p))|)dp}_{E_{image}(C)} \tag{13}$$

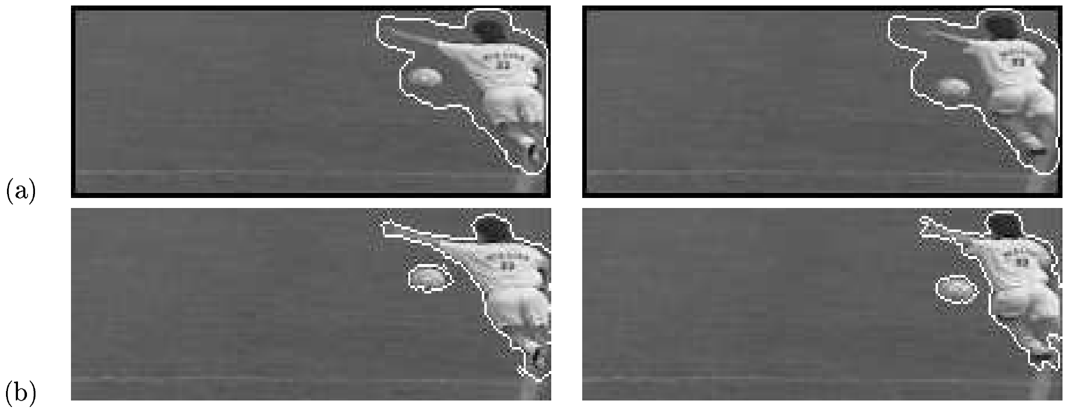


Figure 4: (a) *Detection* of moving objects (projected on the input frames), (b) *Tracking* the moving objects

Using this model and following the work of geodesic active contour we are able to detect the moving areas (fig. 4(a)) if and only if the generated image $I(detection)$ fulfills the conditions assumed during its construction (we assume that this image has large gradient values only close to the borders of the moving area). In the case of non-rigid objects and objects with holes, this result is not equivalent to the moving area, since there are some regions inside the estimated contour which correspond to static image regions. These regions could be easily detected if we continue the curve evolution inside the estimated contour, since we are going to detect the regions inside the moving area that correspond to a static background.

### 3.3 Tracking Part

Complete motion detection is not equivalent to temporal change detection. In other words, the moving estimated area between two successive images corresponds to the union of the moving object locations in these images. The goal is to track the objects in these images. To achieve this, the following modification of the snake model is required:

$$\begin{aligned} E(C(p)) &= \underbrace{(1-\lambda)\int_0^1 |C'(p)|^2 dp}_{\mathbf{E}_{internal}(C)} + \\ & \underbrace{\lambda \int_0^1 (\gamma\, \overbrace{g(|\nabla I_{detection}(C(p))|)}^{\text{detection term}} + (1-\gamma)\, \overbrace{g(|\nabla I_t(C(p))|)}^{\text{tracking term}})^2 dp}_{\mathbf{E}_{image}(C)} \end{aligned} \quad (14)$$

The energy term $E_{detection}$ forces the curve to fit the moving area, avoiding edges or static objects. On the other hand, since the curve is around the moving area, this term is close to zero. The other term, $E_{tracking}$, is used for the curve evolution until the curve reaches the exact location of the moving object. We assume that around the object there are no edges or a texture background , otherwise, the method cannot be applied (fig. 4(b)). Finally, $\gamma \in [0,1]$ is a parameter which balances the contribution of the detection and the tracking term. Selecting a $\gamma$ value close to 0, we push the model to detect the moving objects, while with a value close to 1 we have a classic geodesic active contour model. In the general case $\gamma$ must have a value close to 0.5.

Following the work on geodesic active contours presented in the previous section, this minimization problem is then transformed into a problem of geodesic computation, and the associated Euler-Lagrange PDE is then solved using the level set formulation scheme of Osher and Sethian [29] by viewing it as a front propagating with internal and external image dependent speed :

$$\begin{aligned} \frac{du}{dt} \;=\; & \Big\{\gamma \cdot \Big(g(|\nabla I_{detection}(C(p,t))|) \cdot \mathcal{K}(p,t) + \nabla g(|\nabla I_{detection}(C(p,t))|) \cdot \tfrac{\nabla u}{|\nabla u|}\Big) \\ & + (1-\gamma) \cdot \Big(g(|\nabla I_t(C(p,t))|) \cdot \mathcal{K}(p,t) + \nabla g(|\nabla I_t(C(p,t))|) \cdot \tfrac{\nabla u}{|\nabla u|}\Big)\Big\} \, |\nabla u| \end{aligned} \tag{15}$$

This resulting PDE equation (acting on $u$ evolution) is then solved using techniques borrowed from hyperbolic conservations laws [29], and using the approaches presented in the next section.

# 4 Front Propagation Algorithms

A direct implementation approach of equations such as (6) and (15) involves the re-estimation of the characteristic image of all the level set pixels (not simply the zero level set corresponding to the front itself). This front evolution method is computationally very expensive, due to many useless operations that are performed during the front propagation (especially in pixels which are out of interest). In order to overcome this drawback two different methods have been proposed: (i) the "Narrow Band" method that works with a small percentage of pixels (those which are around to the latest estimation of the contour) [3], (ii) the "Fast Marching" method which drastically reduces the required cost but implies serious limitations concerning its applications [33]. According to the "Fast Marching" method the level set tube and the contour are designed step by step. We propose a new method, where we combine the advantages of the existing methods and design a fast approach suitable to a large variety of applications. Short descriptions of the existing approaches as well as for the new approach are following.

### 4.1 Narrow Band Approach

The key idea is to deal only with pixels which are close to the latest estimation of the zero level-set contour in both directions (inwards and outwards). This is known as Narrow Band Approach [3]. Since the curve evolution is smoothly performed according to the Euler-Lagrange equations, the use of pixels which are far away from the current contour does not effect the evolving process. Thus we can work only with pixels around to the current contour estimation. In that case a set of narrow band points is defined around the latest contour estimation and the evolving Euler-Lagrange equation is performed only for these points (fig. 11). The problem is that the contour position changes dynamically (from iteration to iteration), as well as with the narrow band pixels. The estimation of the contour position from iteration to iteration increases dynamically the cost (in terms of complexity), thus the contour position is re-estimated only in cases where the contour is very close to the borders of the narrow band. There is a mechanism which permits to decide fast and without large computational cost the cases where the contour has to re-estimated. A significant cost reduction is achieved through this approach (compared to the classic method), but certainly the cost still remains considerable.

### 4.2 Fast Marching Approach

Consider a special case of a front moving with a speed $\mathcal{F} = \mathcal{F}(x, y)$, $\mathcal{F} > 0$. Let now consider a monotonically advancing front whose level-set equation is of the form: $\frac{du}{dt} = \mathcal{F}(x, y)|du|$. Let $T(x, y)$ be the time at which the curve crosses the site $(x, y)$. In this time the surface $T(x, y)$ satisfies the equation: $|\nabla T| \cdot \mathcal{F} = 1$. This equation simply says that the gradient of arrival time surface is inversely proportional to the speed of the front.

Using the above equation and an approximation for the gradient norm $|\nabla T|$, (proposed in [3]) we are looking for the solution of:

$$\left[\mathbf{max}\left(\mathbf{max}\left(D_-^x T, 0\right), -\mathbf{min}\left(D_+^x T, 0\right)\right)^2 + \mathbf{max}\left(\mathbf{max}\left(D_-^y T, 0\right), -\mathbf{min}\left(D_+^y T, 0\right)\right)^2\right] = \frac{1}{F_{xy}^2} \tag{16}$$

where

$$\begin{array}{llll} D_-^x T(x,y) = \frac{T(x,y)-T(x,y-1)}{2} & , & D_+^x T(x,y) = \frac{T(x,y+1)-T(x,y)}{2} \\ D_-^y T(x,y) = \frac{T(x,y)-T(x-1,y)}{2} & , & D_+^y T(x,y) = \frac{T(x+1,y)-T(x,y)}{2} \end{array} \tag{17}$$

Concerning the gradient approximation (eq. (16)), different forms could also be used ([32]). Since according to (16), information are propagated in "one way" (that is from smaller to larger values of $T$), it is possible to build the solution outwards of the smallest time value $T$. The idea is to sweep the front ahead in an upwind fashion (by considering a set of pixels in narrow band around the existing front), and to march this narrow band forwards (freezing the values of existing pixels and bringing new ones into narrow band structure).

```
SetInitialContour()
SetNarrowBandPixels()
SetFarAwayPixels()
while ( ExistNarrowBandPixels() )
{
        site = SelectSmallestTValue()
        active = TagNonAliveNeighbors(site)
        MoveFarAwayPixels(active)
        RestimateTValues(active)
}
exit()
```

Figure 5: Fast Marching Algorithm

The algorithm (**fig.** 5) is composed of two basic steps: the **initialization** and the **marching** one. Initially, a user-defined contour initialization takes place. All the pixels belonging to the contour are set to *Alive* and their $T$ value to zero. For all *Alive* pixels we select their neighbors, set them as *Narrow Band* pixels and initialize their $T$ value equal to $\frac{1}{F}$. The rest of the pixels are assumed to be *FarAway* pixels with a $T$ value close to infinite.

During the **marching** step, the *Narrow Band* pixel with the smallest $T$ value is selected and set *Alive* in each iteration. The idea is to propagate the information from the lowest values to the biggest ones. All the neighborhood sites of the selected pixel which are *FarAway* are moved to *Narrow Band*. A re-estimation of $T$ values (according to the quadratic equation (16)) for the *Narrow Band* neighborhood pixels takes place. The algorithm stops when all pixels have been labeled *Alive*.

Despite the fact that *Fast Marching* requires low computational cost it has a **limited set of applications**. The **main drawback is the necessity to have an only positive or negative speed function during the front evolution**. Additionally, this algorithm can be used only for cases with location-dependent speed function. As a consequence, we **can't use this method in a case where there is a curvature-dependent speed function, a very popular case which appears in a large variety of image vision applications**, such as the one we are considering in this article (motion detection and tracking applications).

## 4.3 Hermes Algorithm

We propose in this section a new approach that combines the existing ones and is capable of propagating fronts without any limitations on their speed function. An alternative idea for

```
SetInitialContour()
SetActivePixels()
while ( LargestVelocity() !=0 )
{
        site = SelectLargestVelocity()
        EvolveLevelSet(site,neighbors)
        AddNeighborsActive(site)
        ReestimateVelocity(site,neighbors)
        if ( ActivePixels > Threshold)
        {
                SmoothCharImage()
                FindContourPosition()
                SetActivePixels()
        }
}
exit()
```

Figure 6: Hermes Algorithm

the propagation of a given front is to work with the "strongest" point at each step (**fig.** (6)). Since the Euler-Lagrange equation has been defined for the level set evolution (equation (15)), the contour depends on the time step and the velocity which varies from point to point. For the general case, we write

$$\mathbf{u}^{t+1}_{(x,y)} = \mathbf{u}^{t}_{(x,y)} + \mathcal{V}(x,y,\mathbf{u})dt \tag{18}$$

Since the velocity $\mathcal{V}(x,y,\mathbf{u})$ in many cases is estimated according to image characteristics, there are some pixels for which the front evolves in a faster way compared to the others. The key idea of this approach, is to evolve the contour according to the velocity values.

We propose an algorithm which at each step selects the pixel with the largest velocity and evolves its characteristic value (according to eq. (18)). First we initialize the contour and we set both the contour pixels and their neighbors as *active points*. We select from the active pixels the one with the largest velocity and we evolve it for a certain number of iterations as well as the values of the neighborhood pixels. Since there are modified image values, there are pixels which are affected (in terms of current velocity). For these pixels, we estimate their velocity. If there are neighborhood pixels which are not *active* we label them as *active*. If the pixel value changes sign we remove this pixel from *Active* ones. Additionally, there is a special velocity cost term, which counts only for the selection of the "strongest" pixel (not for the level-set evolution), and depends on the number of times at which this pixel has

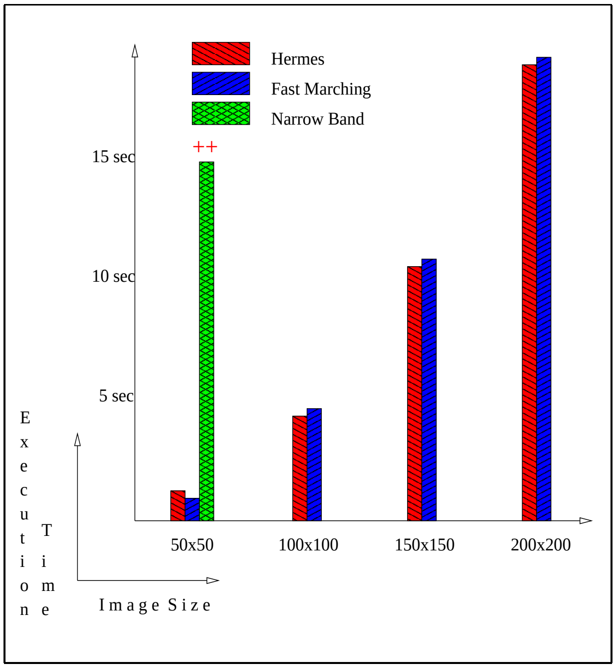



Figure 7: Computational Cost Diagram, for *Front Propagation* algorithms

been selected. Periodically we find the contour position in order to avoid the creation of a large set of *Active* pixels and we perform a smooth operation in the characteristic image, since this image is partially motivated. The algorithm stops when all the contour points (according to the latest estimation) have velocities close to zero, or there are no more *Active* pixels. The selection of the termination criterion cannot be done automatically, which could be considered as a drawback.

The key issue for an efficient version of the Fast Marching algorithm (resp. Hermes algorithm) lies on a fast way of locating the grid point among the Narrow Band points (resp. among the Active points) with the smallest value for $T$ (resp. with the biggest velocity). For this reason a variation of a heapsort algorithm is used. Initially all the narrow points are sorted in a heapsort (so that the smallest member can be easily located). When a point is removed from the heapsort, the values of its neighbors are recomputed, and the results are bubbled upwards until they reach their correct locations. Moreover, whenever we want to add a point to the heapsort, we put it at the end and we process it in the same way.

**The proposed algorithm can deal with cases at which** *Fast Marching* **cannot to be used**. It is completely independent on the form of the speed function, that is capable

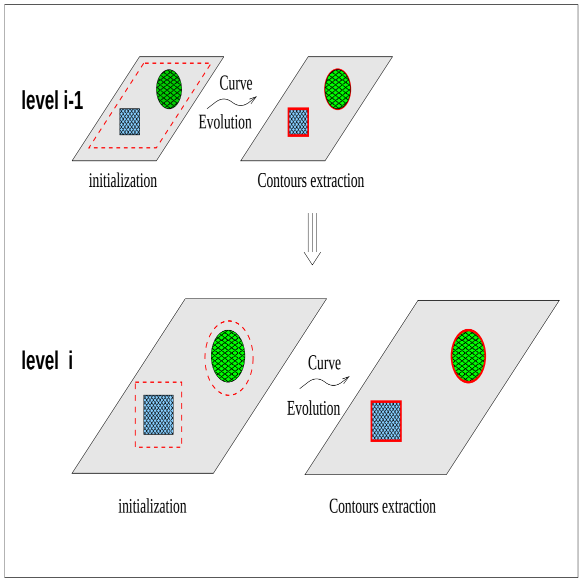



Figure 8: A multi-scale approach for geodesic active contours

to cope with a large variety of level-set applications in image processing (*i.e* curvature based speed functions, etc.)(fig. 14). Additionally, an efficient implementation has high convergence rate and can determine the result very quickly. In order to test the behavior of the proposed approach, we used it for the problem of geodesic active contour detection together with Narrow Band and Fast Marching. We used a synthetic image with four objects where we added gaussian noise. According to our experiments (fig. 7) Hermes provides a little bit less computational cost than *Fast Marching*, while the cost of Narrow Band is at least ten times more. The cost of Hermes and Fast Marching algorithms seems to be proportional to the image size.

## 4.4 Multi-scale Approach

In order to reduce further the computational cost, we propose a multi-scale technique (fig. 8), which can be used combined with the front propagation algorithms. Thus, a Gaussian pyramid of images is built upon the full resolution image and similar geodesic contour problems are defined through the different levels. This multi-resolution structure is then utilized according to a coarse-to-fine strategy. In other words, an extrapolation of the current contour from level with low resolution to levels with finer contour configuration takes place. Thus the current values of $u$ in the image grid are copied and replicated in order to initialize

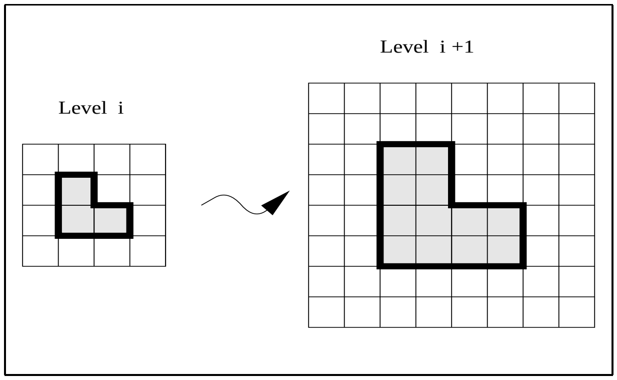



Figure 9: Multi-scale approach: Going from a level to the next (extrapolation scheme)

the values of $u$ at the next level (fig. 9). This extrapolation scheme (initialization of $u$ image) is used as an initial contour, and a new contour evolution is performed. Usually this technique is applied to a pyramid with one or two levels.

# 5 Concluding Remarks

Real-word video sequences have been used to test and validate the proposed approach. For the detection part, we obtained very satisfactory results (fig. 11, 12) . The generated image, based on the inter-frame difference mixture analysis, fits exactly to the assumptions done during its creation. In addition, the use of level-set approach allows to deal with a large variety of objects movements.

The *tracking* part doesn't always give the same quality of results as the detection part. In the case where the moving objects are surrounded by a smooth area, the quality of the solution is very close to the optimal (fig. 15, 16). On the other hand, there are some limitations for the model, since it is not capable to deal with cases where there is a texture background (with edges) closed to the objects. Concerning the level-set implementation two different well-known approaches have been implemented and a new one is proposed. These approaches are evaluated concerning their computational cost [1] and their set of applications (fig. 10). In order to further reduce computational cost a multi-scale approach is combined with the front propagation algorithms We can say that the use of multi-scaling reduces significantly the required cost (fig. 10), especially for *Narrow Band* algorithm. In addition, we could say that the CPU time reduction is proportional to the number of levels, in which

[1] A SPARCstation20 with 64MB memory and a CPU at 75 MHz has been used

| Sequence (size) | levels | Narrow Band | Fast Marching | Hermes |
|---|---|---|---|---|
| Player (208x80) | 0 | 11.3 min | 11.5 sec | 5.8 sec |
| | 1 | 2.7 min | 6.4 sec | 3.1 sec |
| | 2 | 1.4 min | 4.6 sec | 3.3 sec |
| Autoroute (*128x128*) | 0 | 14.8 min | 11.2 sec | 8.1 sec |
| | 1 | 3.8 min | 4.8 sec | 3.4 sec |
| | 2 | 2.2 min | 3.8 sec | 3.3 sec |
| Kollning (*256x256*) | 0 | $30^+$ min | 50.4 sec | 32.8 sec |
| | 1 | 7.6 min | 18.6 sec | 12.2 sec |
| | 2 | 3.5 min | 12.4 sec | 8.9 sec |
| Football match (*408x144*) | 0 | $30^+$ min | 40.5 sec | 30.1 sec |
| | 1 | 7.1 min | 18.4 sec | 12.9 sec |
| | 2 | 3.4 min | 13.2 sec | 9.8 sec |

Figure 10: Computational Cost (CPU time)

the approach has been used. A crucial remark is that the selection of velocity parameters affects significantly on the execution time, especially for the Hermes algorithm.

Summarizing, we have considered a level-set approach for *detection* and *tracking* in image sequences. The main contribution of our approach is a framework for detecting and tracking moving objects in a sequence of images using of level-set methodology. The detection and the tracking are achieved through a simple statistical model, where there is no necessity of optical flow estimation. Based on the inter-frame difference, we are able to create an image where the detection and tracking can be viewed as a geodesic computation problem. The construction of this image is based on the analysis of the inter-frame difference as a mixture of two Laplacian (or zero-mean Gaussian) distributions. A **new, very fast algorithm**, - **which can be used under any case of evolving speed** - for the front propagation problem, is proposed and compared with the existing well-known methods, which have been also implemented. In order to further reduce the computational cost we use a multi-scale approach combined with the different algorithms of front propagation, which permits to track moving objects very fast.

Various experimental results (in MPEG format), including the ones shown in this article, can be found at:

*http://www.inria.fr/robotvis/personnel/nparagio/demos.html*

(a)

(b)

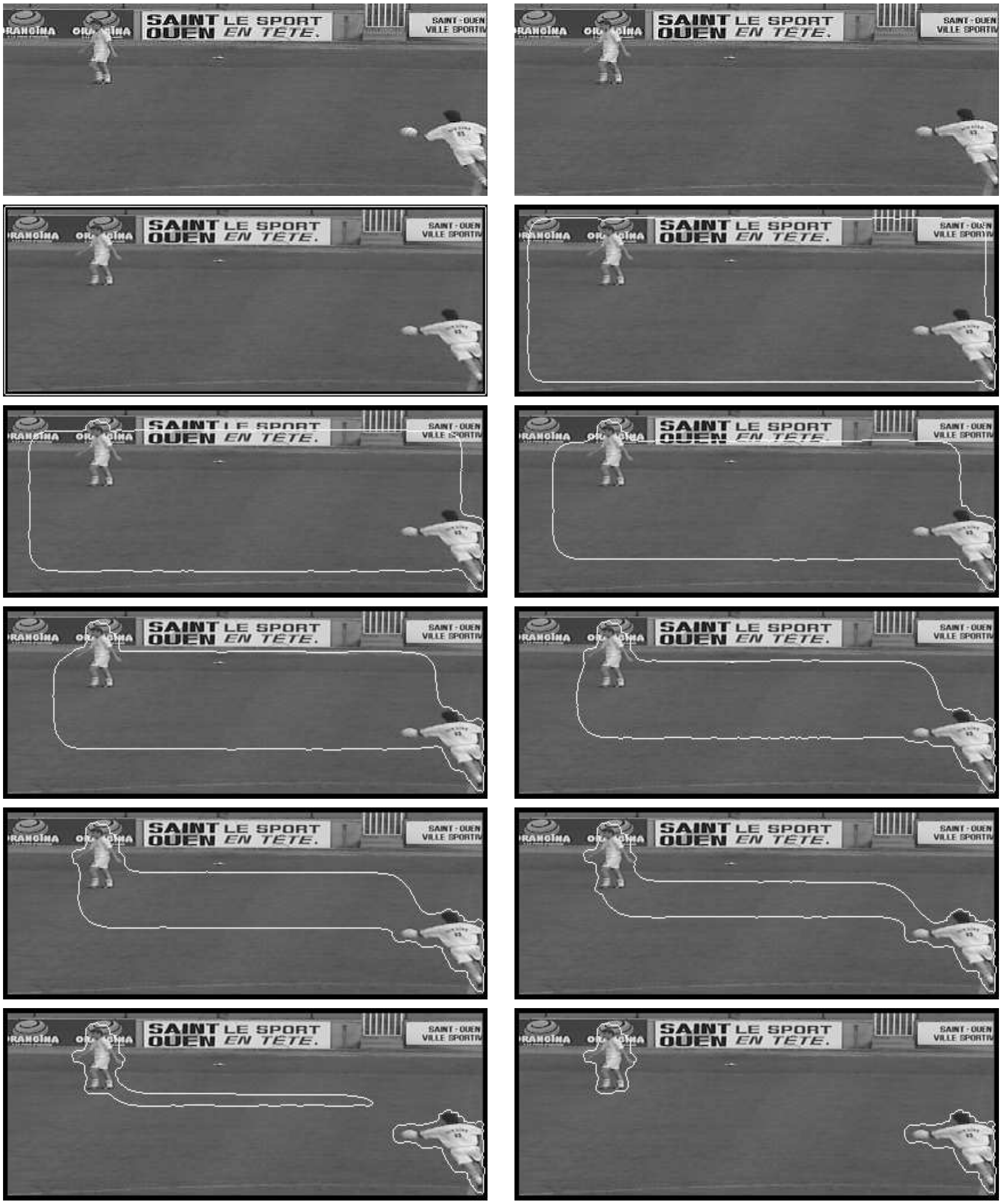
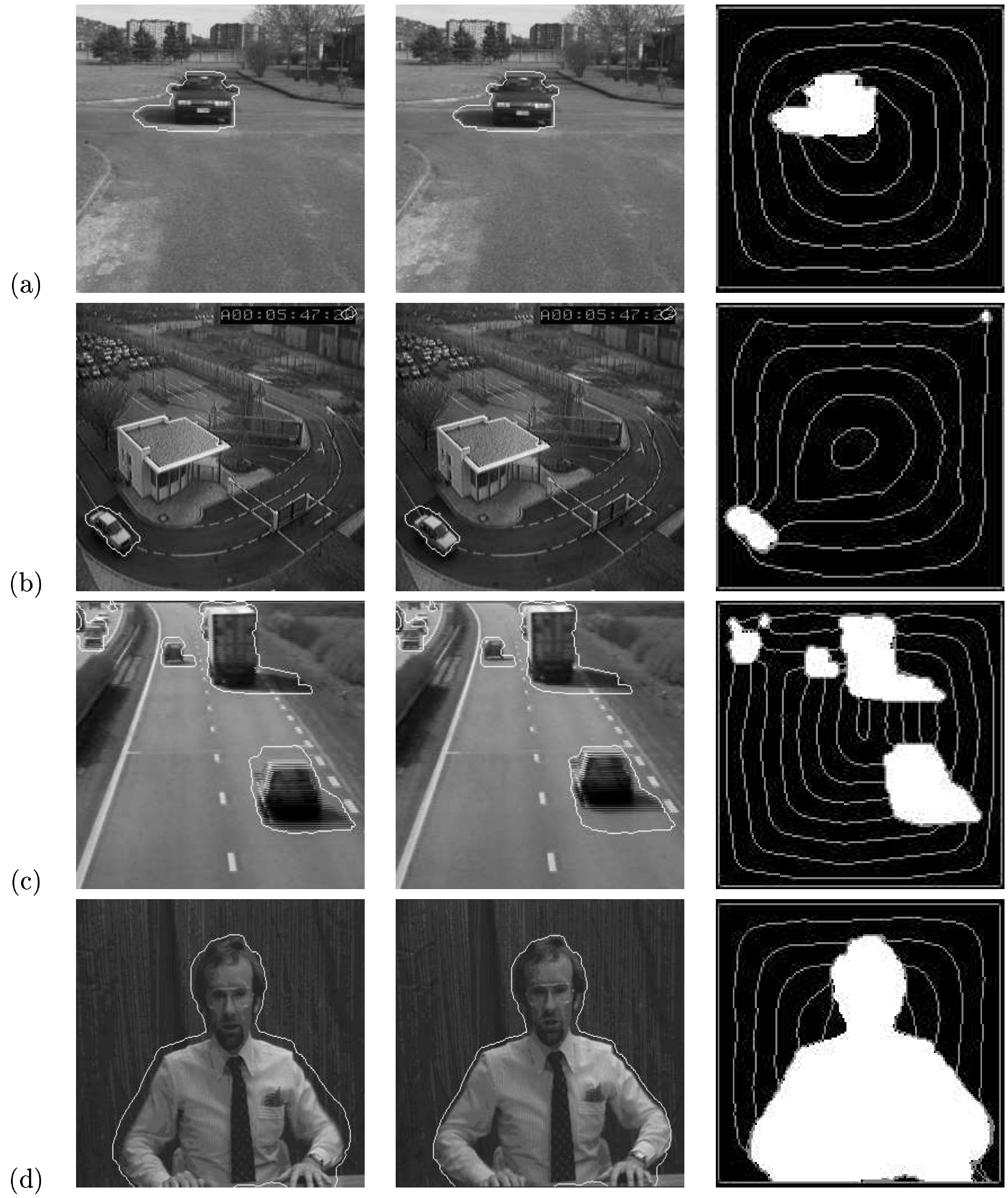



Figure 11: *Detection of moving objects* for *Football match Sequence*. (a) Input frames, (b) Curve evolution projected at the first frame (left to right).



(a) (b) (c) (d)

(1) (2) (3)

Figure 12: (a) *Autoroute Car* sequence, (b) *Kollning* sequence, (c) *Highway* sequence, (d) *Trevor-White* sequence (1), (2) Input Frames & **Moving Area Detection** (white line), (3) **Curve evolution**

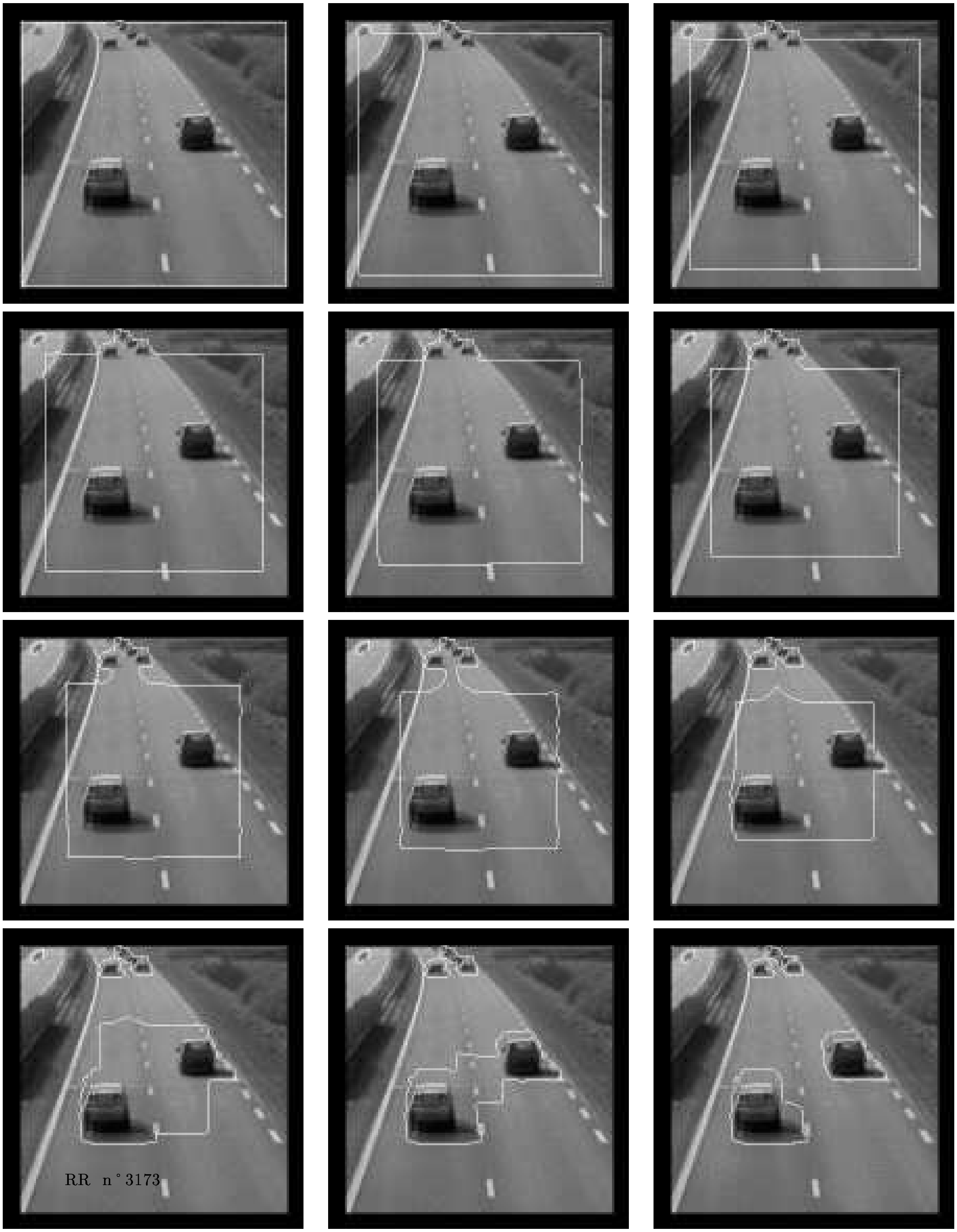


Figure 13: *Curve evolution* projected at the first image for *Autoroute Sequence* (left to right) with Fast Marching Algorithm.

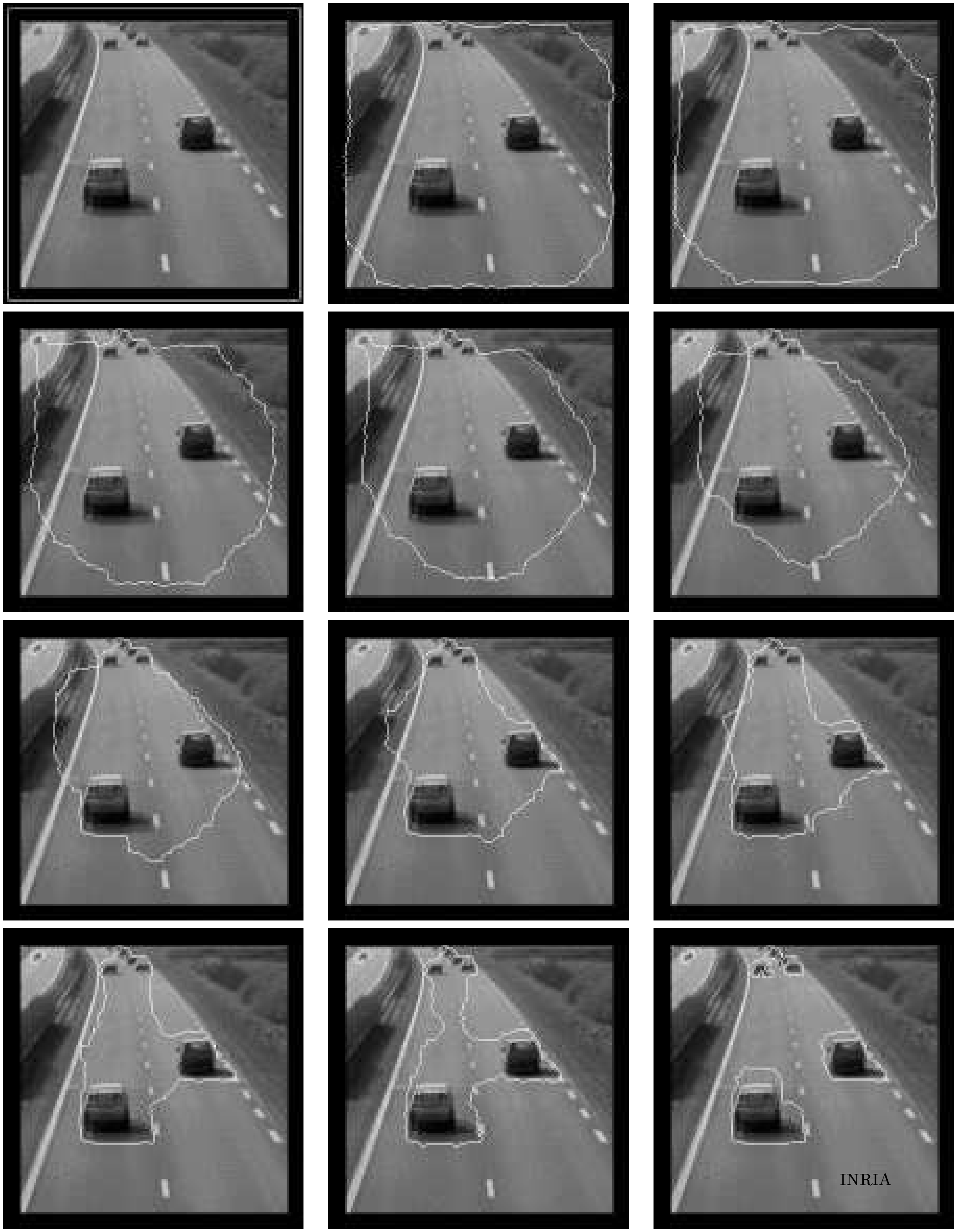


Figure 14: *Curve evolution* projected at the first image for *Autoroute Sequence* (left to right) with Hermes Algorithm.

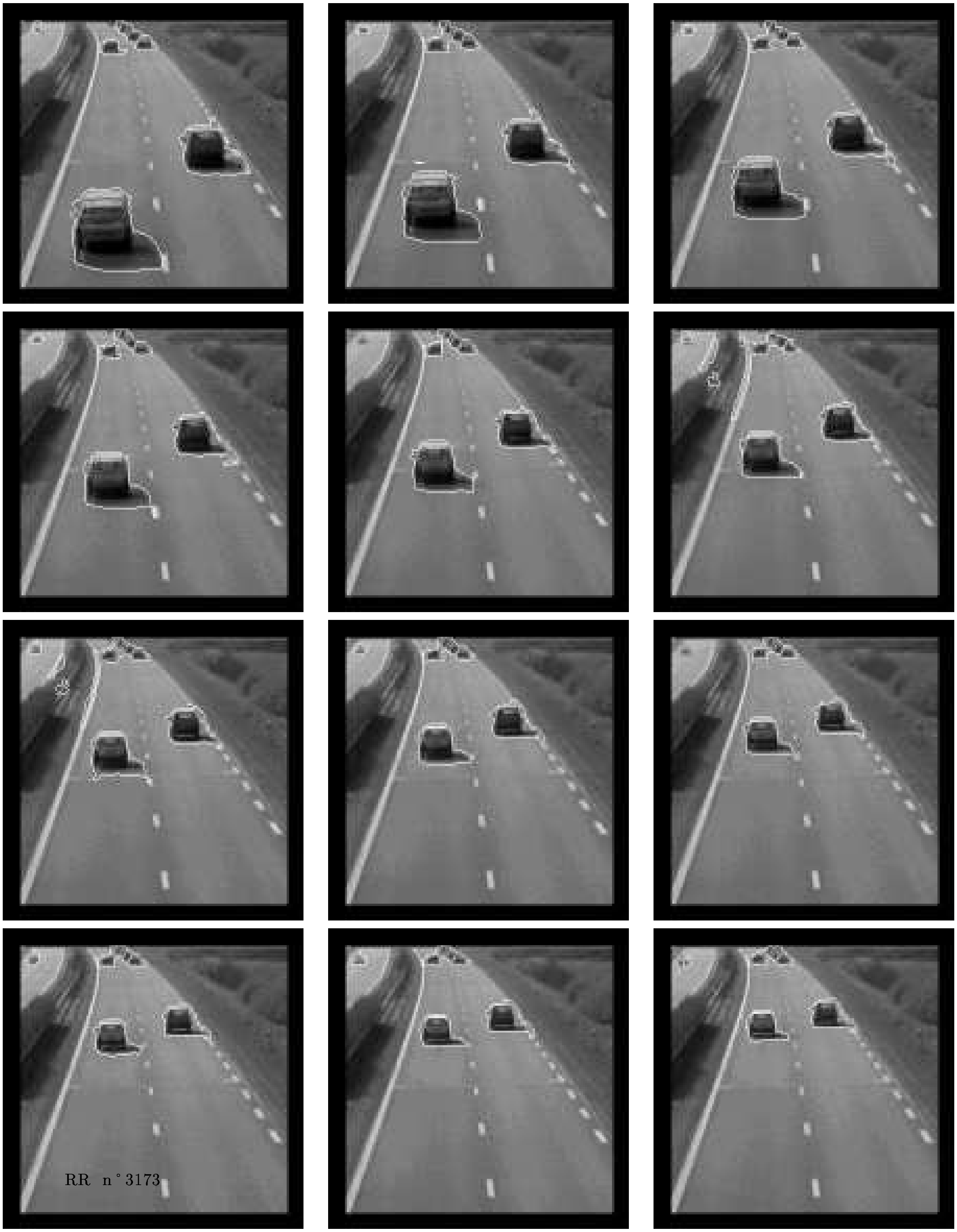


Figure 15: *Tracking* for *Autoroute Sequence* (left to right).

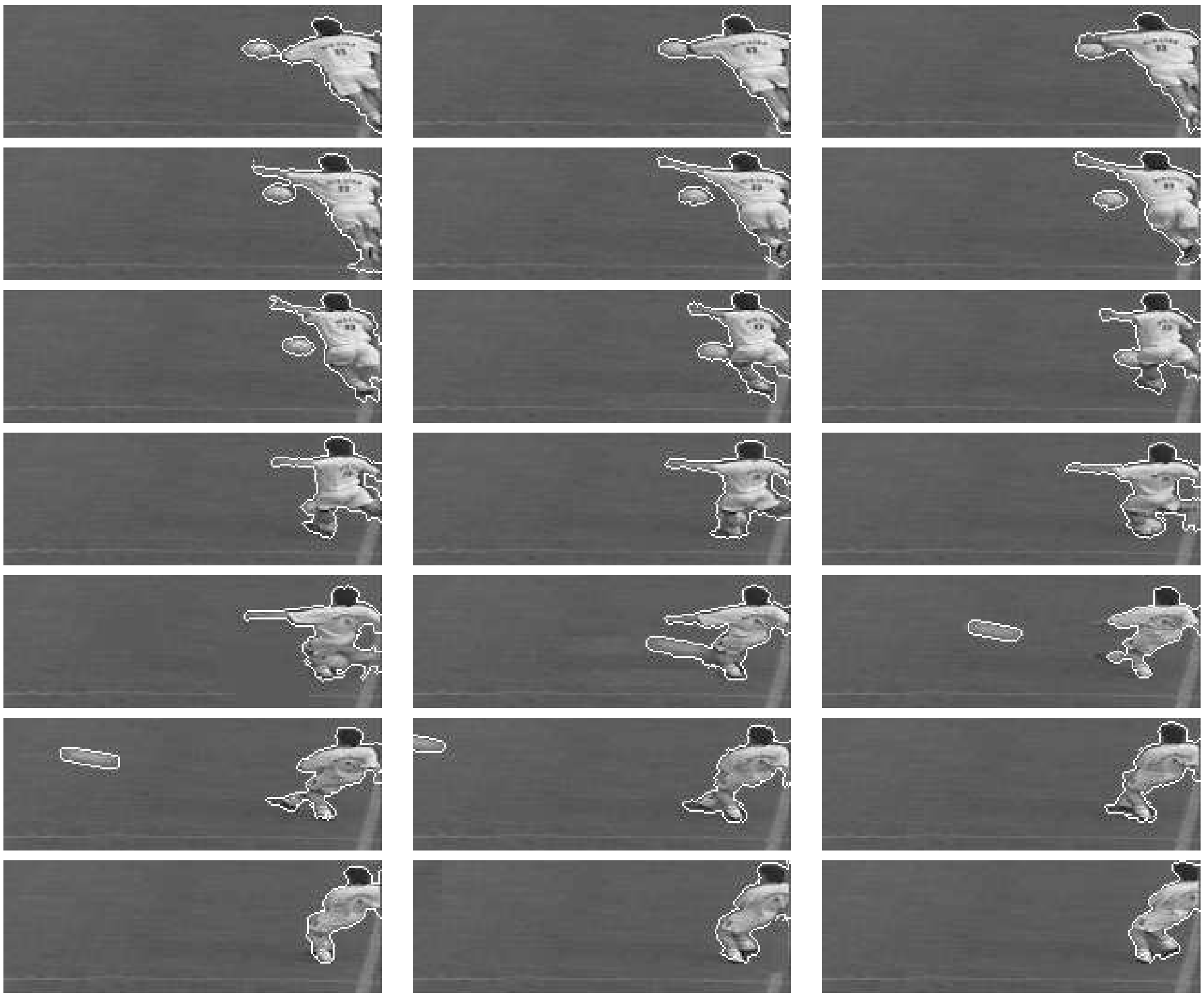


Figure 16: *Tracking* for *Football match Sequence* (left to right).

## References

[1] Boston, MA, June 1995. IEEE Computer Society Press.

[2] T. Aach and A. Kaup. Bayesian algorithms for adaptive change detection in image sequences using markov random fileds. *Signal Processing: Image Communication*, 7:147–160, 1995.

[3] D. Adalsteinsson and J. A. Sethian. A Fast Level Set Method for Propagating Interfaces. *Journal of Computational Physics*, 118(2):269–277, 1995.

[4] B. Bascle, P. Bouthemy, R. Deriche, and F. Meyer. Tracking complex primitives in an image sequence. In *Proceedings of the 12th IAPR International Conference On Pattern Recognition*, pages 426–431, Jerusalem, Israel, oct 1994.

[5] B. Bascle and R. Deriche. Energy based methods for 2D curve tracking, reconstruction, and refinement of 3D curves and applications. In *SPIE Proceedings, Vol. 2031. Geometric Methods in Computer Vision*, San-Diego, California,USA, July 12-13 1993.

[6] Bénédicte Bascle and Rachid Deriche. Region tracking through image sequences. In *Proceedings of the 5th International Conference on Computer Vision* [1], pages 302–307.

[7] M. Berger. How to track efficiently piecewise curved contours with a view to reconstructing 3d objects. In *12th ICPR*, volume 1, pages 32–36, Jerusalem, Israel, October 1994.

[8] Martin Bischel. Segmenting simply connected moving objects in a static scene. *IEEE Transactions on Pattern Analysis Machine Intelligence*, pages 1138–1142, November 1994.

[9] C. Cafforio and F. Rocca. Methods for measuring small displacements of television images. *IEEE Transactions on Information Theory*, IT-22:1973–1979, 1976.

[10] V. Caselles and B. Coll. Snakes in Movement. *SIAM Journal on Numerical Analysis*, 33:2445–2456, December 1996.

[11] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. In *Proceedings of the 5th International Conference on Computer Vision* [1], pages 694–699.

[12] I. Cohen, N. Ayache, and Sulger P. Tracking points on deformable objects using curvature information. In *Proc. 2nd ECCV, Santa Margherita Ligure, Italy*, May 1992.

[13] R. Deriche and T. Blaszka. Recovering and characterizing image features using an efficient model based approach. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 530–535, New-York, June 1993. IEEE Computer Society, IEEE.

[14] Rachid Deriche and Olivier Faugeras. Tracking line segments. *Image and Vision Computing*, 8(4):261–270, November 1990. A shorter version appeared in the Proceedings of the 1st ECCV.

[15] Rachid Deriche and Olivier Faugeras. Les EDP en Traitement des Images et Vision par Ordinateur. *Traitement du Signal*, 13(6), 1996. A preliminary version appeared as an INRIA Research Report, RR-2697, November 1995.

[16] N. Diehl. Object-oriented motion estimation and segmentation in image sequences. *IEEE Transactions on Image Processing*, 3:1901–1904, February 1990.

[17] J. Driessen, J. Biemond, and D. Boekee. A pel-recursive segmentation algorithm for motion compensated image sequence coding. *Proceedings in ICASSP*, pages 1901–1904, April 1989 (New York).

[18] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. New York: Willey-Interscience, 1973.

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

[20] K. Karmann, A. Brandt, and R. Gerl. Moving object segmentation based on adaptive reference images. *Signal Processing: Theories and Applications*, V:951–954, 1990.

[21] M. Kass, A. Witkin, and D. Terzopoulos. SNAKES: Active contour models. *The International Journal of Computer Vision*, 1:321–332, January 1988.

[22] S.P. Liou and R. Jain. Motion detection in spatio-temporal space. *Computer Vision, Graphics, and Image Processing*, 45:227–250, 1989.

[23] R. Malladi, J. A. Sethian, and B.C. Vemuri. Shape modeling with front propagation: A level set approach. *PAMI*, 17(2):158–175, February 1995.

[24] R Malladi, J.A. Sethian, and B.C. Vemuri. A topology independent shape modeling scheme. *SPIE*, 2031:246, 1993.

[25] G. McLachlan, D. Peel, and W. Whiten. Maximum likelihood clustering via normal mixture model. *Signal Processing: Image Communication*, 8:105–111, 1996.

[26] F. Meyer and P. Bouthemy. Region-based tracking in an image sequence. In *Proc. 2nd ECCV, Santa Margherita Ligure, Italy*, pages 476–484, May 1992.

[27] H.-H. Nagel, G. Socher, H. Kollnig, and M. Otte. Motion boundary detection in image sequences by local stochastic tests. In J.-O. Eklundh, editor, *Proceedings of the 3rd European Conference on Computer Vision*, pages 305–315, Vol.II, Stockholm, Sweden, May 1994.

[28] J-M. Odobez and P. Bouthemy. Robust multi-resolution estimation of parametric motion models. *Visual Communication and Image Representation*, pages 348–365, December 1995.

[29] S. Osher and J. Sethian. Fronts propagating with curvature dependent speed : algorithms based on the Hamilton-Jacobi formulation. *Journal of Computational Physics*, 79:12–49, 1988.

[30] N. Paragios, P. Perez, G. Tziritas, C. Labit, and P. Bouthemy. Adaptive detection of moving objects, using multiscale techniques. *Proceedings of ICIP'96*, 1:525–528, 1996 (Laussane).

[31] N. Paragios and G. Tziritas. Detection and location of moving objects, using deterministic relaxation algorithms. *Proceedings of ICPR'96*, 1:201–205, 1996 (Vienne).

[32] Elisabeth Rouy and Agnes Tourin. A Viscosity Solutions Approach to Shape-from-Shading. *SIAM J. Numer. Anal.*, 29(3):867–884, June 1992.

[33] J. A. Sethian. A Fast Marching Level Set Method for Monotonically Advancing Fronts. In *Proc. Nat. Ac. Science*, volume 93, pages 1591–1694, 1996.

[34] J. A. Sethian. *Level Set Methods*. Cambridge University Press, 1996.

[35] Z. Sivan and D. Malah. Change detection and texture analysis for image sequence coding. *Signal Processing: Image Communication*, 6:357–376, 1994.

[36] Demetri Terzopoulos. Regularization of inverse visual problems involving discontinuities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:413–424, 1986.

[37] G. Tziritas and C. Labit. *Motion Analysis for image sequence coding*. Elsevier, 1994.

[38] O. Wenstop. Motion detection from image information. *Proceedings in Scandinavian Conference on Image Analysis*, pages 381–386, 1983 (Copenhagen).

[39] A. Yuille, D. Cohen, and P. Hallinan. Facial feature extraction by deformable templates. Technical Report 88-2, Harvard University, 1990. Harvard Robotics Laboratory.

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
ISSN 0249-6399