An a contrario decision framework for motion detection

Thomas Veit, Frédéric Cao, Patrick Bouthemy

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

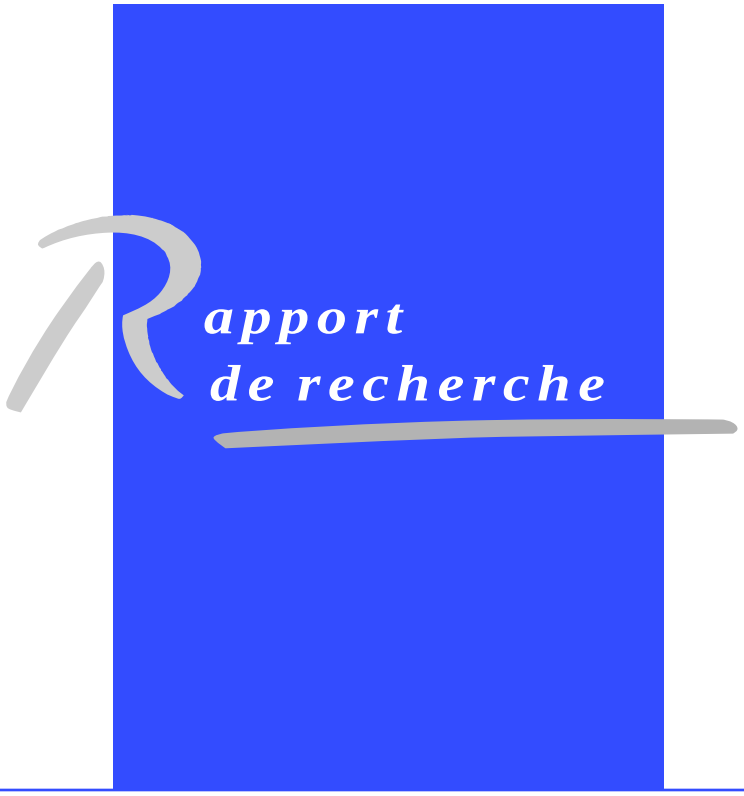
An a contrario *decision framework for motion detection*

Thomas Veit, Frédéric Cao, Patrick Bouthemy

N°5313

Septembre 2004

Systèmes cognitifs



ISSN 0249-6399 ISRN INRIA/RR--5313--FR+ENG

An *a contrario* decision framework for motion detection

Thomas Veit*, Frédéric Cao†, Patrick Bouthemy‡

Systèmes cognitifs
Projet Vista

Rapport de recherche n°5313 — Septembre 2004 — 27 pages

Abstract: Motion detection aims at discriminating between moving objects and a static environment. This task can be seen as the grouping of local motion observations into moving objects. The framework we propose is derived from a perceptual grouping principle, namely the Helmholtz principle. It consists in defining an image model in the absence of moving objects instead of modeling the moving objects. This prevents from any complex model design while enforcing the generality of the approach, since there is no prior to specify on the objects to be detected. Detections are then said to be performed *a contrario*: moving regions appear as low probability events in the "no motion" or *a contrario* model. The modeling framework induced by this approach is compact and handy, since it is simply built on independent identically distributed random variables. Furthermore, computing automatic detection thresholds and attaching a confidence level to each detected moving region is possible through the probabilistic setting of the framework. The resulting detection algorithm is thus truly generic and avoids parameter tuning. The method performance is assessed on various real image sequences.

Key-words: *a contrario* decision, detection, motion

(Résumé : tsvp)

* thomas.veit@irisa.fr
† frederic.cao@irisa.fr
‡ patrick.bouthemy@irisa.fr

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu, 35042 RENNES Cedex (France)
Téléphone : 02 99 84 71 00 - International : +33 2 99 84 71 00
Télécopie : 02 99 84 71 71 - International : +33 2 99 84 71 71

Décision *a contrario* pour la détection de mouvement

Résumé : La détection de mouvement consiste à séparer les objets mobiles d'un environnement statique à partir de mesures de mouvements locales dans une séquences d'images. La séquence peut être acquise par une caméra fixe ou mobile. L'approche que nous proposons s'inspire d'un principe de groupement perceptuel, le principe de Helmholtz. Ce principe décrit des mécanismes de la perception humaine: un événement est perceptuellement significatif s'il contredit un modèle aléatoire. Dans le cas de la détection de mouvement, il s'agit donc de modéliser la situation dans laquelle il n'y a pas d'objets mobiles et non pas les objets mobiles eux-mêmes. Le principal avantage est que le modèle *a contrario* d'absence d'objets mobiles est plus général et plus simple qu'un modèle représentant un objet en mouvement. Ce modèle se base sur des variables aléatoires indépendantes identiquement distribuées. Les détections se font alors *a contrario*: les objets mobiles apparaissent comme des configurations de très faibles probabilités. Cette formulation probabiliste permet de fixer les seuils de détection automatiquement. De plus, un niveau de confiance est associé à chaque détection. L'algorithme ainsi obtenu est général et il évite le réglage de paramètres. La méthode est validée sur des séquences vidéos réelles et variées.

Mots-clé : décision *a contrario*, détection, mouvement

1 Introduction

The detection of independently moving objects in a scene observed by a mobile camera is a central issue in computer vision. Developing reliable, efficient and really automatic solutions has an impact on several important applications. This paper describes a novel approach to detect independently moving regions in an image sequence. Given a region of the image, we want to answer the fundamental question: "Is this region moving ?" by "yes" or "no". To this purpose, the mathematical formalization of a perceptual grouping principle, namely the Helmholtz principle, results in an *a contrario* detection scheme [14]. Applied to motion detection, this means that we do not aim at modeling the moving objects we want to detect but rather the absence of moving objects in an image. The model of the image in which no moving object is present is therefore called *a contrario* model. A careful specification of this model results in bringing moving objects to the fore as events of very low probability (i.e. large deviations to this model). The proposed decision process automates the detection threshold selection and its robustness avoids parameter tuning. A shorter preliminary presentation of the *a contrario* framework applied to motion detection can be found in [44].

Our motivations were to design a general and reliable method for deciding if the gray level variations of a given region of an image were due to camera motion or independent motion (this includes of course the application to video sequences captured by a static camera). We want to automatically derive appropriate thresholds for deciding on the presence of moving objects in the viewed scene. We believe that fixing the detection threshold should not be the result of an empirical process and should not be done a posteriori. Leaving the user with a cursor to balance between false alarms and missed detections is rather a drawback than a necessity. Motion information is salient enough in order to distinguish moving objects from noise.

One way to detect independent motion is motion segmentation. This kind of methods relies on an accurate estimation of the optical flow or more often parametrized motion models. This can be achieved by alternating estimation and segmentation steps or by computing motion layers [19]. Optical flow representation of multiple motions provides an effective visual representation of the dynamic content of an image. However, the accuracy of the optical flow computed on small supports like moving objects is questionable. Our method depends on motion estimation only for the computation of the global dominant image motion. Global motion is identified with camera motion. Once camera motion is compensated, image differencing techniques can be applied.

Image differencing techniques are appealing because of their simplicity. The intensity differences of two consecutive images are sensitive to small registration errors and are not invariant to contrast change. Therefore, the magnitude of the normal residual flow is preferred. However, the thresholding question remains. Despite the apparent simplicity of image differencing methods, the threshold selection question is still an open one. This detection threshold often needs to be set by the user. In our point of view, setting the detection threshold and answering the question on the presence of motion is the task of the detection algorithm and not of the user.

Some methods for selecting the threshold have been proposed. In most cases, a single global threshold is computed via likelihood techniques. However, a single threshold for the whole image is not always sufficient in the presence of multiple motions. Apparent motions can be very different in

their magnitude, thus requiring different thresholds. The presnented method involves an automatic threshold selection adapted to each image region.

Thresholding the image differences alone does not enable to recover the boundaries of moving objects. Let us stress that often only a part of the moving object may display perceptible motion information. Motion information is absent from regions of homogeneous intensity, some can be exploited in textured parts and most of it concentrates on edges. On top of that, edges parallel to the motion direction do not convey motion information. This known as the aperture problem. To recover the boundaries of moving regions, one usually resorts to some kind of regularization. The point is to propagate motion information in a sensible way. Variational approaches, like active contours or Markov Random Fields (MRF) help localizing the contours of moving objects. However, these models implicitly assume the presence of a moving object. Thus, they do not solve the detection issue itself. To recover moving object boundaries, an *a priori* gray level segmentation based on the image intensity level lines is utilized. This way accurate shape information is accessed. The segmentation is independent of the image motion information. This is crucial for the *a contrario* framework.

Level set theory and MRFs might seem very different but both rely on some kind of energy minimization. The involved energies are often very similar. In general, the minimization process does not provide the value of the minimum of the energy function. Assessing the quality of the detection is therefore impossible. The proposed detection procedure associates to each detected region a confidence level.

Our approach is the following. Let us first assume that no moving object is present. Then, temporal intensity variations are due to noise only. This noise is supposed uncorrelated. This noise model is our *a contrario* model or background model. In this context, the word "background" has to be understood in its statistical meaning. It does not refer to the background of the scene but to the random image difference behaviour in the absence of motion. Modeling the absence of motion is much simpler than defining a model for the moving objects. Designing generative models for moving objects is possible for specific applications (for example, gait recognition), but it is hardly possible in the general case. If a moving object is present, the values of temporal image changes are highly spatially correlated, strongly contradicting the background model. The probability of such an event within a compact region can be computed according to the *a contrario* model. The whiteness assumption on noise facilitates this probability computation. The probability is very low for a region displaying a coherent change (numerically about 10^{-10}). This leads to a so-called *a contrario* detection. Furthermore, the event probability can be related, through expectation, to an average number of occurrences. This number of occurrences is computed using the *a contrario* model and is called Number of False Alarms (NFA). A confidence level is associated to each region through the NFA. The lower the expectation of occurrences in the *a contrario* model, the more meaningful the detected event. The proposed motion detection algorithm aims at tackling in an automatic way the questions of presence (with evaluation of a confidence level), position and shape of the moving objects contained in the scene.

This paper is organized as follows. In Section 2, previous work on motion detectionis described. Section 3 introduces the *a contrario* decision framework and its mathematical formalization. Sec-

tion 4 presents the proposed motion detection algorithm, and Section 5 concerns its implementation. Experiments are reported in Section 6. Finally, concluding remarks are given in Section 7.

2 Related work on motion detection

Previous approaches to motion detection can be split into two categories: methods based on motion segmentation and methods thresholding image differences. A synopsis of methods from both categories is proposed in [26].

Motion segmentation methods require an accurate estimation of the 2D apparent motion in the image. This is not trivial since computing motion estimation on the various image supports arising from objects in the scene is definitely demanding. However, some specific problems cannot be solved without information on the orientation of motion. For example, apparent motion estimation enables to conclude that the motion of a static background and a static foreground, although different in their 2D projection, are induced by the same camera motion and should therefore not be detected as independently moving. This can be done using parallax and rigidity constraints [17], [32], [43], [42] and [8].

Thresholding methods are applied to temporal image differences. The temporal image differences of static regions and those of moving objects can be statistically modeled. Within the class of thresholding methods, different kinds of image structures can be considered. They may range from decision on single pixels to spatial regularization using MRFs or active contours. The highest level of structure is representing moving objects by their spatio-temporal envelopes. A different form of structure is to consider only edges and to detect those that are moving.

A review of basic methods is given in [22]. Likelihood tests for motion detection are introduced in [16]. More complex methods are proposed in [38] for modeling the spatial distribution of either noise or signal and selecting an appropriate threshold. Markov Random Fields are a natural extension in order to introduce contextual information into the detection scheme [1]. Previous to image differencing, camera-induced motion can also be estimated and compensated. To this end, a 2D parametric motion model is used in [34, 33] along with hierarchical MRFs. In [9] wavelet analysis and robust techniques are introduced to estimate dominant motion and hierarchical MRFs are again exploited.

A higher level of spatial structure can be reached using active contour and the level set theory. For instance the approach described in [36] applies active contours for detection and tracking of moving objects. It relies on the gray level intensities for providing object boundaries. Two methods using level sets implementation are introduced in [25]: one purely based on motion and another enforcing correspondence between motion boundaries and intensity boundaries. Besides, they can distinguish between different moving objects.

Bayesian approaches like MRF and variational methods both rely on energy minimization techniques. In the Bayesian framework, the aim is to at maximizing the posterior probability of a model given the observations. Variational techniques minimize an energy functional yielding a contour evolving according to some constraints. The formulation of energies for both approaches is similar. It consists of a regularization term and a term to fit observations. However, it is not possible to assess the validity of the extremum. In other words, it is not possible to interpret the value of the extremum.

Thus, the best state is reached given the model and the observations, but the quality of this state can not be assessed.

Temporal integration improves quality and stability of the detection. Accumulating motion information over time makes moving objects more salient with regard to noise. This enables to detect small slowly moving objects. Directionally consistent flow is accumulated over time in [45]. A graph to represent moving objects is exploited in [7], and object trajectories are optimal paths in this graph. Spatio-temporal image intensity gradients to create mosaics of the background are used in [37]. Residual motion is then propagated and accumulated without optical flow computation. A threshold allows to balance between false alarms and minimal detectable motion. It is not clear how to set this threshold, unless empirically.

To overcome the sparseness of motion information in images captured by classical sensors, multi-sensors method can be used. For example, thermal infrared (IR) sensors are complementary to Electro-Optical (EO) sensors and help to overcome the ambiguity of motion information in poorly contrasted regions. IR are also less sensitive to illumination changes. Recent work in this field can be found in [20, 31].

As mentioned above, motion information accumulates on edges. The work in [40] concentrates on these highly contrasted features to detect multiple layered motions. In [18] contour fragments are matched. The different transformations issued from those matchings are clustered into background and moving objects. This work is closer to shape matching issues than to motion detection. Both methods rely on a Canny-type edge detector.

Some work has also been devoted to distinguish between motion and changes. Changes are variations of image intensities which do not correspond to real a moving object : shadows and reflections but also aliasing [4].

Finally, the use of perceptual criteria for change detection appears in [23] and [39]. In [23] an *a contrario* framework is applied to detect changes in satellite images of urban landscapes. The considered local change information is the image gradient orientation. In [39], the use of perceptual organization is considered to build the spatio-temporal envelopes of moving objects. The approach is essentially applied to human undergoing fronto-parallel motion in front of a static camera. The envelopes localize the motion information but do not provide shape information.

The work described in this paper is close to thresholding approaches. Camera-induced motion is compensated using 2D parametric motion models. The perceptual grouping principle allows the computation of automatic detection thresholds. Our algorithm does not resort to temporal integration or tracking to improve detection. Detection operates on three frames only. Boundaries of moving objects are retrieved through an image segmentation based on gray-level intensities. Moreover, a confidence level for each detected region is derived through the so-called number of false alarms evaluated according to the *a contrario* model. The lower this number, the more reliable the detected event.

3 *A contrario* decision framework

The *a contrario* decision approach is a mathematical formalization of the so-called Helmholtz perceptual grouping principle. Its application to image analysis has been developed by Desolneux et

al. [11]. It defines perceptually meaningful configurations as large deviations to a random model. In other words, conspicuous events are those that are very unlikely to occur by chance, i.e., in a noise situation. As a consequence, no detection should occur in a pure white noise image. The idea of computing the probability of accidental occurrence was introduced in [3] and more recently in [24]. Similarly, a robust estimator is constructed in [41] assuming that outliers are randomly distributed and inliers are the points which are least likely to have occurred randomly. In the next paragraph, we mathematically formalize this principle, which has been successfully applied to the detection of alignments [10], edges [13], vanishing points [2], good continuation [5], and shape matching [29]. Before doing this, let us outline the main ingredients of this principle. Contrarily to more classical statistical methods, and particularly Bayesian methods, we do not try to design a model of what is sought from the observations. On the contrary, a model of the statistical background is designed which grossly represents the absence of relevant detection. If a particular event, with a very low probability, occurs in this background model, then this event certainly has a better explanation than chance alone. In that case, this naturally leads us to reject the background model and, *a contrario*, to accept this detection as valid or relevant. Let us stress that the events to be detected have to be specified prior to the observation. Of course, these events have to be defined so that they correspond qualitatively to some perceptually meaningful structures.

3.1 From hypotheses testing to *a contrario* decision

In order to introduce the *a contrario* decision framework, let us first consider the more classical hypothesis testing point of view as also proposed in[30] . Let us consider a given region R of the image. Let $C(R)$ be a random variable representing a motion measure for the region. The measure $C(R)$ is positive and increases with the magnitude of the apparent motion on region R. Let $c(R)$ represent a realization of the random variable. We want to answer the question "Is the region R a static region ?" by "yes" or "no". Let us try to give an answer to this question using classical statistical hypotheses testing. The two hypotheses are H_M: "the region R is moving" versus H_S: "the region R is static". Let L_M and L_S be the associated likelihood functions under each hypothesis.

Evaluating tests is usually done using error probabilities. Rejecting hypothesis H_M although the region is moving, results in a mis-detection (type I error). Accepting H_M whereas the region is in fact static, results in a false detection (type II error). The probabilities associated to each error are respectively

$$\alpha = P(\text{reject } H_M | H_M) \quad \text{and} \quad \beta = P(\text{accept } H_M | H_S). \tag{1}$$

The probability of type I error α is called the level or size of the test. The power of the test is defined as $1 - \beta$.

Let us define the following likelihood ratio test of level α^*: accept H_M if $\frac{L_M(c(R))}{L_S(c(R))} > h_{\alpha^*}$ and reject it otherwise. The threshold h_{α^*} defining the rejection region is solution of $P\left(\frac{L_M(c(R))}{L_S(c(R))} \leq h_{\alpha^*} | H_M\right) = \alpha^*$. The Neyman-Pearson lemma states that this test is a uniformly most powerful test inside the class of tests of level α^*. The level of the test has to be specified by the user. Moreover, the test requires the knowledge of the likelihood function under the motion hypothesis. This is a major obstacle.

A Bayesian test takes the following form. Given prior probabilities on motion presence P_M and motion absence P_S, the posterior probabilities $P(H_M|c(R))$ and $P(H_S|c(R))$ are accessed. A Bayesian test can then be defined: Accept H_M if $P(H_M|c(R)) > P(H_S|c(R)) \iff P(c(R)|H_M)P_M > P(c(R)|H_S)P_S$ and reject it otherwise. For this Bayesian test, not only is the distribution of motion information over the region R under the motion hypothesis required, but the priors for static or mobile regions have also to be defined.

Our point is that, modeling the spatial behavior of motion information over region R certainly results in a complex distribution because of the strong spatial correlation of motion information on a moving object. In the absence of a generative model for moving objects, other techniques have to be resorted to in order to determine whether the region R is moving or not.

3.2 *A contrario* decision test

Since no model of the motion information over the region under hypothesis H_M is available, let us consider the model in the absence of motion and the probability of false detection $P(\text{accept } H_M|H_S)$. The probability distribution in the absence of motion is much easier to model in a reasonable way. It is sound to assume that the set of pixels corresponding to the region can be then considered as independent identically distributed random variables. Let us define the following test: Accept H_M if $c(R) > c^*(R)$, where $c^*(R)$ is such that $P(C(R) > c^*(R)|H_S) < \gamma$, for $\gamma < 1$ to be specified. This test is said to be an *a contrario* decision. The motion hypothesis is accepted as soon as the observation is very unlikely under the static hypothesis. Similarly to the Neyman-Pearson test, γ is fixed and $c^*(R)$ has to be determined. The subtlety of the argument is that γ can be set automatically.

The probability $P(C(R) > c^*(R)|H_S)$ represents the probability of false detection. Introducing the total number of tested regions N_R gives access to the average number of false alarms.

Definition 1 (ε-meaningful moving region) *Let $\varepsilon > 0$. Let us consider a set of N_R regions independent of the observations. A region R is said to be an ε-meaningful moving region if $c(R) > c^*(R, \varepsilon)$, where $c^*(R, \varepsilon)$ is solution of $c^*(R, \varepsilon) = \min\{c, P(C(R) > c|H_S) < \frac{\varepsilon}{N_R}\}$.*

The *a contrario* decision test is now: Accept the hypothesis H_M for a region R if R is an ε-meaningful moving region. As a consequence, the average number of ε-meaningful moving regions under the hypothesis H_S, that is to say the number of false detections, is less than ε. The proof is straightforward. Let N be the random variable corresponding to the number of ε-meaningful moving regions. Then, $N = \sum_{i=1}^{N_R} \mathbf{1}_{\{R_i \text{ is a } \varepsilon\text{-meaningful region}\}}$, yielding

$$\begin{aligned} \mathbf{E}(N|H_S) &= \sum_{i=1}^{N_R} P(C(R_i) > c^*(R_i)|H_S) \\ &< N_R \frac{\varepsilon}{N_R} \text{ by definition of } c^*(R_i) \\ &= \varepsilon \quad . \end{aligned} \tag{2}$$

The Number of False Alarms (NFA) of a region R is defined as follows.

Definition 2 (NFA of a region) *The number of false alarms is defined as*

$$NFA(R) = N_R \cdot P(C(R) > c(r)|H_S) \tag{3}$$

The function $c \mapsto P(C(R) > c)$ is non-increasing. Thus, if $NFA(R) < \varepsilon$ then the region is ε-meaningful. The number of false alarms of R indicates how false the background model is for the considered region. If $NFA(R)$ is very small, then there is certainly a better explanation of its occurrence than the considered simple random model. Therefore, the region R must be a moving region. The lower $NFA(R)$, the more salient or meaningful the motion of the region. The only parameter in the decision is ε. Nevertheless, it can be shown, and it will apply for our method, that ε can vary a lot with no practical incidence. Indeed, we shall see that only the logarithm of ε plays a role in the detection. As a consequence, replacing ε by 10ε or $\frac{\varepsilon}{10}$ does not change the results. In practice, ε is set to 1, meaning that one detection on average is expected in the background model when testing N_R regions.

4 Motion detection algorithm

In this section, the application of the *a contrario* decision framework to motion detection is presented. The main ingredients of the method are the following. First, the dominant image motion identified with camera motion is compensated. Then, a residual motion measure is defined. By using backward and forward estimation of the dominant motion, occlusion and disocclusion of the background by moving objects are handled. Given a region defined as a set of points, the residual motion measure for each point is compared to a given value. The *a contrario* decision framework does not need this value to be specified exactly, since a whole set of values are tested, while still controlling the number of false detections. Another issue is the temporal size of the detection window. The definition of the NFA can also be adapted to several time steps.

4.1 Dominant motion estimation

The dominant motion between two successive images is here represented by a 2D parametric model w_θ. A quadratic motion model with 8 parameters is chosen. The expression of the motion model is

$$w_\theta(p) = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} + \begin{pmatrix} a_3 & a_4 \\ a_5 & a_6 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a_7 & a_8 & 0 \\ 0 & a_7 & a_8 \end{pmatrix} \cdot \begin{pmatrix} x^2 \\ x \cdot y \\ y^2 \end{pmatrix} \tag{4}$$

with $\theta = (a_1, \ldots, a_8)$, and $p = (x, y)$.

This quadratic motion model was preferred to an affine one because it is exact for planar scenes undergoing rigid motion. The model is fixed for all experiments. The additional computational cost compared to the affine model is negligible with the utilized software. A limitation resides in the ability of the 2D parametric motion model to handle complex scene geometries. Nevertheless, a planar approximation of the background holds in numerous situations. The quadratic motion model cannot deal with scene backgrounds displaying large depth discontinuities compared to the distance

to the camera. If so, static objects at different depths from the one corresponding to the computed dominant image motion are detected as moving with respect to the dominant motion. A posterior step could differentiate between really independently moving objects and objects detected because of a strong parallax.

The estimated dominant image motion is supposed to be caused by camera motion. This assumption is true up to some limitations such as the size of the objects in the scene. For example, if an object covers more than half of the image, the estimated dominant might be the motion of this large object. However, in most cases and in the presence of multiple moving objects, dominant motion is correctly attached to camera motion.

Different methods exist to estimate the parameters of such a motion model. The robust multi-resolution method described in [35] is utilized. It can handle large displacements, it is real-time and accurate [1]. An M-estimator criterion is minimized involving a hard-redescending function ρ (Tukey's biweight function):

$$\hat{\theta}_t = \arg\min_{\theta_t} \sum_p \rho(DFD_{\theta_t}(p, t, t+1)) \ , \tag{5}$$

where the Displaced Frame Difference (DFD) at point p for the motion model to be estimated is given by

$$DFD_\theta(p, t, t+1) = I(p + w_\theta(p), t+1) - I(p, t) \ , \tag{6}$$

$I(p,t)$ and $I(p + w_\theta(p), t+1)$ being the image intensity functions at time t and $t+1$, respectively at point p and $p + w_\theta(p)$. When $p + w_\theta(p)$ is non-integer, the intensity value is bilinearly interpolated. Minimization is achieved using iteratively re-weighted least squares within a Gauss-Newton incremental scheme through a multi-resolution framework.

4.2 Local residual motion observation

Once the camera motion, or more precisely the dominant image motion, is computed, residual image motion information has to be computed. An object will be detected as moving (strictly speaking, as non conforming with the estimated dominant motion), if its residual image motion is large enough. Thus, a real valued measure which is large if and only if residual motion is large is needed. Assuming constancy of gray level along trajectories, the simplest way to check the adequacy of each pixel with the estimated global motion is the DFD given in (6).

However, this quantity is extremely sensitive to spatial intensity gradient. Pixels with a large spatial intensity gradient may display large DFD values, even if the residual motion is low. Indeed, small errors in the dominant motion estimation are enhanced along highly contrasted edges. On the contrary, regions where the spatial intensity gradient is low obviously have low DFD whatever the magnitude of the residual motion.

For this reason, a more appropriate measure is the residual normal flow w_{res} magnitude. Its absolute value given by is considered

$$w^{res}_{\hat{\theta}}(p, t, t+1) = \frac{|DFD_{\hat{\theta}}(p, t, t+1)|}{\|\nabla I(p, t)\|}. \tag{7}$$

[1] The corresponding software can be downloaded from www.irisa.fr/vista/Motion2D

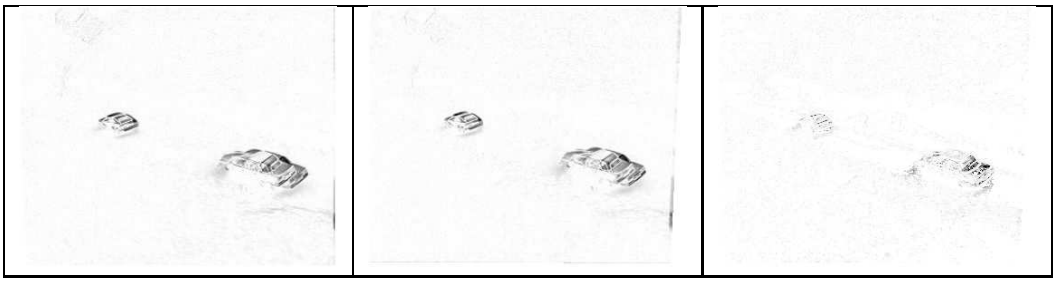


Figure 1: Maps of the backward and forward DFD and of the residual motion measure C_t for time instant $t = 290$ of the road sequence (Fig. 11). Dark pixels correspond to large values. The ghosting effect is significantly reduced.

Let us note that this quantity under its differential form $\frac{\frac{\partial I}{\partial t}}{\|\nabla I\|}$ has the nice property of being contrast invariant (i.e. invariant under transformation of the type $I \rightarrow g(I)$ with g increasing). This will imply that our detection criterion is independent of contrast (up to the dominant motion estimation). This is coherent with the psychophysical argument that image analysis is widely independent of contrast [21, 6].

A large value of the quantity w^{res} indicates that the motion of the corresponding point differs from the estimated dominant motion. High values are essentially generated by moving objects in the scene or noise. In order to avoid division by zero and since regions with low image gradient convey only unreliable information about motion, the points of the image where the spatial gradient magnitude is less than 2 are ignored.

In order to deal with occlusion and disocclusion of the scene background by moving objects, a three-image scheme on images $I(t-1)$, $I(t)$ and $I(t+1)$ is considered. The reference image remains $I(t)$. Two dominant motions are estimated: a forward one from $I(t)$ to $I(t+1)$, leading to a set of parameters θ_t^{t+1}, and a backward one from $I(t)$ to $I(t-1)$, leading to θ_t^{t-1}. The resulting quantity considered is finally

$$C_t(p) = \min(w^{res}_{\theta_t^{t+1}}(p,t,t+1), w^{res}_{\theta_t^{t-1}}(p,t,t-1)) \ . \tag{8}$$

Looking forward and backward in time ensures that a correct motion evaluation can be achieved, since the two pairs of images cannot simultaneously be affected by an occluding or disoccluding situation at the same time. Therefore, taking the minimum of both w^{res} results in removing most of the high w^{res} values due to occlusion (Fig. 1).

The observation C_t is the local observation involved in the *a contrario* decision process for motion detection.

4.3 *A contrario* model for motion detection

In the previous section, the local motion observations C_t that forms the basic elements of the grouping process was defined. Now, the definition of an *a contrario* model on these elements and the

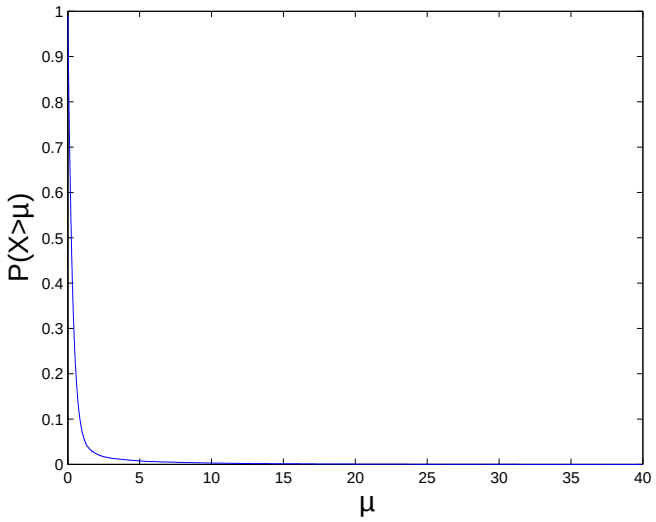


Figure 2: This graph corresponds to the empirical distribution of C_t computed using frames 289, 290 and 291 of the "road" sequence (Fig. 11). The graph represents the probability $P(X > \mu)$ that C_t exceeds a threshold μ.

construction of candidate moving regions have to be defined. Let N_R be the number of candidate regions in the image. The way they are extracted is described in Subsection 5.1.

By now, our aim is first to define the *a contrario* model, that is to say the model corresponding to the absence of moving objects in the scene. In this case it is sound to consider that the observations C_t are independently, identically distributed random variables. Since the general form of the distribution of C_t is unknown (anyway, it is not reasonable to assume that a single distribution could account for all image sequences), the empirical distribution of C_t is considered. Integrating the empirical distribution function (frequency histogram) yields the function $F(\mu) = P(X > \mu)$, where X is a random variable distributed according to the empirical distribution of the observed values C_t (Fig. 2).

Let us consider a threshold μ on the magnitude of the residual normal flow. The threshold selection issue is addressed in the following subsection. Let R be a region of the image of size n, let k be the number of points at which C_t assumes a value larger than μ. Let us define the event E = "At least k points of a region of size n assume a value C_t larger than μ". According to the *a contrario* model, the probability P(E) is

$$P(E) = B(k, n, F(\mu))$$

where $B(k,n,p)$ is the tail of a binomial distribution

$$B(k,n,p) = \sum_{i=k}^{n} \binom{n}{i} p^i (1-p)^{n-i} \quad .$$

Let us now give a first definition of an ε-meaningful moving region.

Definition 3 *For a fixed threshold μ, and a given region R of size n containing k points for which C_t is above threshold μ, define*

$$NFA_1(R) = N_R \cdot B(k,n,F(\mu))$$

where N_R is the total number of regions.

A region is said an ε-meaningful moving region if $NFA_1(R) < \varepsilon$.

As in the general formulation of the *a contrario* framework stated in Section 3, this definition directly implies the following proposition.

Proposition 1 *The expected number of ε-meaningful moving region in the a contrario model is less than ε.*

Proof (Proposition 1). Let $k^*(n,\mu) = \min\{k, B(k,n,F(\mu)) < \frac{\varepsilon}{N_R}\}$. Let $R_j, \quad j = 1,\ldots,N_R$ be the set of all candidate regions, n_j the number of points of R_j and k_j the number of values in R_j larger than μ. Let T_j be the Bernoulli random variable associated to each region, where T_j = 1 if the corresponding region is ε-meaningful and 0 otherwise. The expected number of ε-meaningful moving regions is

$$\begin{aligned}
\mathbf{E}[\sum_{j=1}^{N_R} T_j] &= \sum_{j=1}^{N_R} \mathbf{E}[T_j] = \sum_{j=1}^{N_R} P(T_j = 1) \\
&= \sum_{j=1}^{N_R} P(k_j > k^*(n_j,\mu)) = \sum_{i=j}^{N_R} B(k^*(n_j,\mu)), n, F(\mu)) \\
&< \sum_{j=1}^{N_R} \frac{\varepsilon}{N_R} = \varepsilon,
\end{aligned}$$

which proves the result. ■

The setting of ε allows us to control the number of false detection. For example, fixing ε to 1 as done in practice means that the average number of false detection in a frame distributed according to the *a contrario model* is less than 1. Large deviations theory helps us to approximate the tail of the binomial $B(k,n,p)$. A refined version of the Markov inequality, known as the Hoeffding inequality, gives us an upper bound to the tail of the binomial distribution.

Proposition 2 (Hoeffding inequality [15]) *Let k, n be positive integers with $k \leq n$, and p a real number such that $0 < p < 1$. Then if $r = k/n \geq p$, the inequality holds*

$$B(k,n,p) \leq exp\ \left(nr \ln \frac{p}{r} + n(1-r) \ln \frac{1-p}{1-r} \right) \tag{9}$$

This inequality is used to evaluate an upper bound to the number of false alarms since evaluating $B(k,n,p)$ and more precisely $\binom{n}{k}$ for large n is not possible with classical integer programming.

The following result gives an asymptotic approximation of $k^*(n,\mu) = \min\{k, B(k,n,F(\mu)) < \frac{\varepsilon}{N_R}\}$, the minimal number of points for a region to be detected.

Proposition 3 (asymptotic behavior of $k^*(n,\mu)$ [12]) *When $N_R \to \infty$ and $n \to \infty$ such that $\frac{n}{(\ln N_R)^3} \to \infty$, one has*

$$k(n,\mu) = n \cdot F(\mu) + \sqrt{2 \cdot F(\mu) \cdot (1 - F(\mu)) \cdot \left(\ln \frac{N_R}{\varepsilon} + O(\ln \ln N_R) \right)} \tag{10}$$

The number of points $k^*(n,\mu)$ only depends on the logarithm of $\varepsilon$and N_R. This means that the detection results are robust to changes of ε and N_R.

An interesting consequence is that there is a minimal size for a region to become meaningful. This minimal region size can be computed explicitly and is about ten pixels.

4.4 Multiple thresholds on residual motion

The value of the threshold μ on residual motion cannot be fixed, since the results are certainly sensitive to this choice. Actually, the *a contrario* framework allows us not to choose any particular μ, but to test a whole set of values. The counterpart is to divide ε by the number of tested values. Since the detection depends on the logarithm of ε, the number of tested values has a low influence.

Using multiple thresholds turns out in a new definition of an ε-meaningful moving region.

Definition 4 *Given a set of thresholds μ_i, $i = 1, \ldots, N_\mu$, and a region R_i of size n_i containing k_i points for which C_t is above threshold μ_i, define*

$$NFA_2(R) = N_R \cdot N_\mu \cdot \min_{i=1,\ldots,N_\mu} B(k_i, n_i, F(\mu_i))$$

where N_R is the total number of regions and N_μ the numbers of thresholds.

A region is said an ε-meaningful moving region if $NFA_2(R) < \varepsilon$.

Again, ε-meaningful moving regions according to this second definition occur less than ε times on average for observations distributed according to the background model.

Proposition 4 *The expected number of ε-meaningful regions in the a contrario model using multiple thresholds is less than ε.*

Proof (proposition 4). Let $k_{n,\mu,\varepsilon} = \inf\{k \quad \text{s.t.} \quad B(k,n,F_t(\mu) \leq \frac{\varepsilon}{N_r N_\mu}\}$. For any region R_j of size n_j and a threshold μ_i, let $T_{j,i}$ be the associated Bernoulli random variable which equals 1 if C_t assumes values larger than μ_i at at least $k_{n_j,\mu_i,\varepsilon}$ points among the n_j and 0 otherwise. Then, $P(T_{j,i}=1)$ equals $B(k_{n_j,\mu_i,\varepsilon}, n_j, F_t(\mu_i))$ which is less than $\frac{\varepsilon}{N_r N_\mu}$. Let $T_j = 1$ if R_j is ε-meaningful and 0 otherwise. Then $T_j = 1 \Leftrightarrow \exists j,\ T_{j,i} = 1$. The expected number of ε-meaningful regions is

$$\begin{aligned}
\mathbf{E}\left(\sum_{j=1}^{N_r} T_j\right) &= \sum_{j=1}^{N_r} \mathbf{E}(T_j) \leq \sum_{j=1}^{N_r}\sum_{j=1}^{N_\mu} \mathbf{E}(T_{j,i}) \\
&= \sum_{j=1}^{N_r}\sum_{i=1}^{N_\mu} P(T_{j,i}=1) \leq \sum_{j=1}^{N_r}\sum_{i=1}^{N_\mu} \frac{\varepsilon}{N_r N_\mu} = \varepsilon. \quad \blacksquare
\end{aligned} \tag{11}$$

In other terms, a region is ε-meaningful with multiple thresholds, if there is at least one threshold for which it is $\frac{\varepsilon}{N_\mu}$-meaningful in the sense of Definition 3.

When choosing the threshold, the corresponding probability function $F(\mu) = P(X > \mu)$ has to be considered. Since there is a large majority of small values of C_t and only a few large values, the graph of this function is decreasing very steeply and then flattens. As a consequence, choosing regularly spaced thresholds results in probabilities that are very close for large thresholds. Therefore, regularly spaced probabilities are considered and the corresponding thresholds are derived. Starting from a minimal threshold $\mu^* = 1$ with corresponding probability $p^* = F(\mu^*)$, the other thresholds are those having probability $F(\mu_i) = p^* \frac{i}{N_\mu}, \quad i = 1, \ldots, N_\mu - 1$.

4.5 Temporal detection window size

As previously said in Section 2, the detection of small slowly moving objects requires the integration of motion information over time. Indeed, what makes small slowly moving objects perceptible is the persistence and coherence in time of the motion. As specified up to this point, the detection algorithm might fail in detecting small objects undergoing slow motion. They do not represent a sufficient deviation to the *a contrario model* in order to become meaningful.

The algorithm works on three successive images; I_{t-1}, I_t and I_{t+1}. It can be adapted to the detection of small slowly moving objects if applied to more temporally distant images; $I_{t-\delta t}$, I_t, $I_{t+\delta t}$. Comparing images at more distant time instants results in increasing the apparent residual motion of the moving objects. Of course, δt has to be chosen relatively small (1 to 5) depending on the magnitude of the estimated global motion. The size δt of the detection window could be related directly to the dominant motion magnitude.

Similarly to multiple thresholds, the definition of the NFA can be adapted and the expected number of false alarms is still under control.

5 Implementation issues

5.1 Candidate regions / spatial segmentation

The motion detection test is fully specified in the previous section. Now, relevant candidate regions on which we can apply the motion detection test have to be defined. The first set of candidate regions considered is one of the simplest: square blocks of different sizes. The second set of candidate regions aims at fitting more precisely to the image content. It is based on the meaningful level lines in the image at time instant t.

5.1.1 Quadtree segmentation

A simple spatial image partitioning is to divide the image into square blocks while considering blocks of different size. One way of doing this is to construct a tree of square blocks of dyadic sizes, i.e. a quadtree.

The main advantage resides in the simplicity of the segmentation which implies low computation time. Moreover, blocks of different sizes can adapt to some extent to image content. It is then possible to detect smaller moving objects at a fine resolution of the quadtree, while larger objects can be handled at a lower resolution.

However, this simplicity means also a lack of flexibility. The positions of the considered regions are fixed and cannot fit the moving object location. No shape information is contained in the blocks, no temporal consistency of the detected regions is guaranteed either.

Experiments using quadtree segmentation make obvious that often only a small part of a moving object really carries motion information. The absence of motion information on moving objects is mainly due to uniform intensity areas and to the aperture problem (the orientation of the spatial intensity gradient is collinear to the motion direction and thus no motion information is available). This means detecting the entire object projection using motion information only is not always reachable.

5.1.2 Maximal meaningful level lines

Since one cannot rely on motion information only, better candidate regions than the blocks provided by a quadtree segmentation are needed. Testing regions corresponding to objects or part of objects in the image instead of testing blocks enables to propagate the motion information in a more clever way.

Many segmentation methods based on intensity levels exist and can be used as an input to the defined *a contrario* motion detection criterion. The one that is utilized is the maximal meaningful level lines method (Fig. 3), which for our purpose outperforms classical segmentation methods. This image segmentation method based on gray-level information was developed by Desolneux et al.[13]. It was preferred because of its specific advantages.

Level lines are not to be confused with the level set theory and active contours. Level lines can be defined as the isophote lines of gray levels. A more general definition is to consider level lines as the boundaries of upper or lower level sets:

$$\chi_\lambda = \{p / I(p) \leq \lambda\} \quad \text{and} \quad \chi^\lambda = \{p / I(p) \geq \lambda\} \ . \tag{12}$$

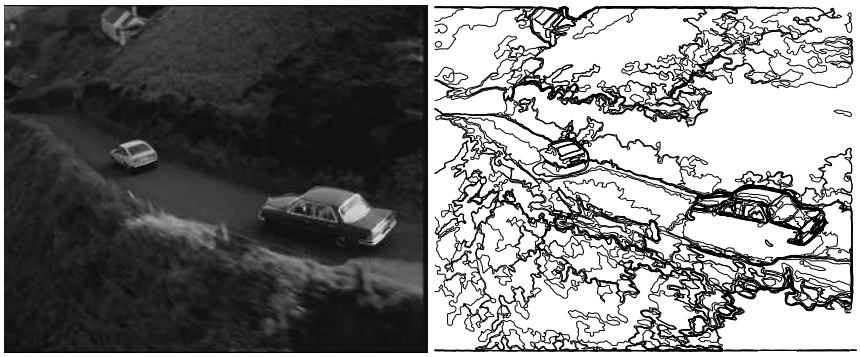


Figure 3: Original image on the left. On the right, the 194 maximal meaningful level lines (out of 18500 level lines).

The collection of all level lines yields a complete representation of the image. This means that given the whole set of level lines, one can exactly reconstruct the original image. The computation of the level lines of an image can be achieved using the Fast Level Set Transform (FLST) algorithm [27, 28]. However, the number of level lines in an image is huge, about 10^5 (most of them are very small, a few pixels). The method proposed in [13] selects only the significantly contrasted level lines among the whole set. This is done by applying the Helmholtz principle. The minimal value of the spatial intensity gradient along a level line and its length is taken into account. The *a contrario* model is based on the empirical distribution of the spatial intensity gradient.

Several advantages make this segmentation appealing for our detection algorithm. First of all, level lines define closed regions which can directly be exploited as support for our candidate regions. No edge linking is needed as it is the case with a Canny-type detector. Secondly, the detection of meaningful, contrasted level lines is based on an *a contrario* detection model and therefore involves no parameter tuning. This enhances the generality of the motion detection algorithm. The shape information provided by level lines is accurate and stable enough. Objects, or parts of objects can be precisely described and shape information is preserved over time. Maximal meaningful level lines also inherit from the tree structure of level lines. This has some computational advantages. Finally, this segmentation is invariant to affine contrast changes making it robust to illumination variations.

Using a gray-level-based image segmentation also implies some limitations . Since our candidate regions correspond to contrasted parts of the image, moving objects that are not contrasted enough might not be detected. This problem is not specific to maximal meaningful level lines. Detecting poorly contrasted moving objects is a challenge on its own. Poor spatial intensity gradient distribution causes the sparseness of reliable motion information. A specific drawback of level lines is that a maximal meaningful level line may enclose part of the static background and part of a moving object. It is not clear which decision to take when this happens. Experiments show that this type of lines can be detected.

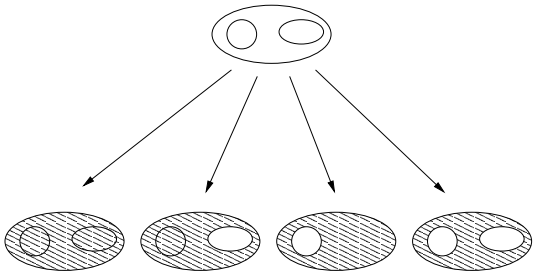


Figure 4: Upper row: three regions defined by three level lines. Lower row: possible pixel assignments for the largest region. Either the support is the whole interior or one or several inner regions are suppressed from the support. The inclusion of an inner region can also made be dependent of its ε-meaningfulness.

There is no ideal image segmentation. Both considered segmentations have their own advantages depending on what the output of the detection process should be. If the main interest is to localize moving objects grossly but quickly, the quadtree structure should be used. If recovering the shape of moving objects is important and available motion information sparse, then the level line segmentation should be preferred, since it provides a better representation of the content of the image. Accessing shape information implicitly enhances temporal coherence of the detected moving regions.

5.2 Tree structure and recursive NFA computation

The two presented segmentations involve a tree structure. This tree structure is exploited for the computation of NFAs. Nevertheless, since the regions of the proposed segmentations are nested, the detection of a large static region due to the fact that it contains a smaller moving region has to be avoided. Each pixel should be involved in the detection of one region only: the smallest moving region containing it.

For both segmentations, there is an ambiguity on the pixels which should be involved in the NFA computation of a given candidate region. As for quadtree segmentation, each pixel belongs to several square blocks. For maximal meaningful level lines, a region defined by a level line encompasses other regions if other level lines are enclosed.

There are several possibilities to define the support of a region containing other region (Fig. 4). The first one is to assign all the inner pixels of a given region to this region. However, as soon as a candidate region contains a moving object it can be detected as a moving region, even if the region is much larger than the moving object. Another possibility is to suppress pixels belonging to one or several inner regions from the overall region.

We decide to make the region construction dependent on the detection decision on inner regions (see Fig. 5). This results in an adaptive region construction process combining the two strategies described in the previous paragraph. Taking advantage of the tree structure, the *a contrario* motion

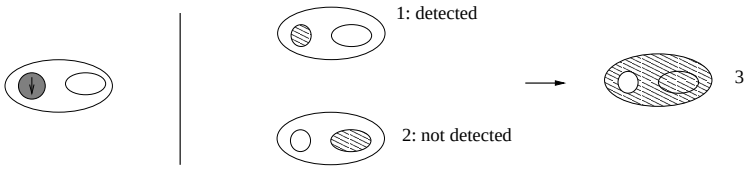


Figure 5: Defined motion detection supports for three nested level-lines. On the left, three level lines. The shaded one corresponds to a moving object. The other two lines are supposed static. In the middle, the two smallest regions are tested (considered pixels are hachured). The two small regions include no other level lines, their whole interior is therefore considered (Regions 1 and 2). Among the two small regions, only the one corresponding to case 1 is detected. It is then removed from the largest region tested in 3 (right).

detection scheme starts from the leaves of the tree. There is no ambiguity for a region that does not contain any other region. If the candidate region is detected as a moving region, then the pixels involved are removed from further computation. If the region is not detected as moving then its pixels are attributed to the next including region. This prevents large regions containing moving objects from being detected. It also allows a more detailed description of moving regions since small regions are tested first. Another advantage of this approach is that candidate regions grow until they are large enough to become significant.

5.3 Overview of the motion detection algorithm

The implementation builds up on the tools described in the previous section. Let us point out that the motion detection algorithm works with three images of a sequence. Loading the three images, I_{t-1}, I_t and I_{t+1} is the first step. Then, the hierarchical spatial segmentation is computed in the reference image I_t. The next step is to estimate the two parametric motion models, the forward one from I_t to I_{t+1} and the backward one from I_t to I_{t-1}. Then, the residual motion quantities C_t are computed. The histogram of the C_t values is computed as well as the empirical probabilities $P(C_t > \mu)$. The thresholds μ_i, $i = 1, \ldots, N_\mu$, are then computed as explained in Subsection 4.4.

Using Definition 4 and given the total number of candidate regions and the number of considered thresholds, one can compute the NFA of any candidate region. NFA computations are performed recursively using the scheme described in Subsection 5.2 to handle nested regions. Starting from the smallest regions, points of a detected region are removed from larger regions.

The whole image sequence can be processed by shifting one image forward. There is consequently only one new image to load and one forward motion model to estimate. The next backward motion model can be derived from the previous forward motion model. The rest of the process remains the same (image segmentation, residual motion computation, NFA computation).

6 Experimental results

The proposed method was tested on several outdoor and indoor sequences. These sequences involve both rigid and articulated motions, static and moving camera. Experiments with the two spatial segmentations described in the previous section are reported (Fig. 6, 8 and 11). The threshold ε on the number of false alarms is set to 1 for all sequences.

6.1 Highway sequence

The first video sequence depicts an outdoor scene corresponding to a highway scene (Fig. 6). The camera is static. A pedestrian stopped his car on the right emergency lane and is running across the highway from right to left. As explained in Subsection 4.5, the case of small slowly moving objects is handled here. The motion detection algorithm is therefore applied with a time step δt equal to 2.

The middle column of Fig. 6 shows the motion detection results using the quadtree segmentation. At time instant $t = 166$, the running pedestrian is detected with NFA of 10^{-1} for the upper body part and 10^{-2} for the lower body part. These NFAs are not so low compared to the detection threshold $\varepsilon = 1$. The size of the pedestrian projection in the image is small and his displacement speed is about 1 pixel per frame. This explains why the observed motions are not very meaningful. At time instant $t = 204$, the pedestrian is detected along with an approaching car. NFAs are equal to 10^{-2} for the car and 0.9 to 0.5 for the pedestrian. The pedestrian is slowing down when approaching the central strip, so that his motion is even less meaningful than before. The detected moving car is in the distance which explains that its projection conveys moderate meaningfulness because of its size and low motion in the image. Let us point out that the size of the detected blocks is smaller when the pedestrian is running than when it is walking. This is coherent with the *a contrario* detection scheme. In order to reach the same level of meaningfulness, a smaller region has to display larger motion. Conversely, slowly moving regions need to be large enough in order to be detected.

The third column in Fig. 6 contains the detected moving regions using the image segmentation based on the meaningful level lines. The motion detection results are consistent with those using the quadtree segmentation, but now the pedestrian is well described in terms of shape by two closed level lines. The head size is below the minimal detectable size. Level lines do not only convey shape description of the objects but they also improve motion detectability. Since the candidate regions fit much better to the objects, the NFA decreases. At time instant $t = 166$, NFAs for the pedestrian upper and lower body parts decrease from 10^{-1} and 10^{-2} with the quadtree segmentation, to 10^{-3} and 10^{-6} using the level line description. At time instant $t = 204$ when the pedestrian is moving slower, NFAs decrease from 0.9 and 0.5 to 0.4 for the upper body part and 10^{-7} for the legs. The NFA for the legs is much lower since they are moving much faster than the upper part of the body.

Let us remind here that, the lower the NFA, the larger the deviation to the *a contrario* model, and therefore, the more meaningful with respect to motion detection. The NFA can be directly used as a confidence level. Much lower NFAs can be reached for larger moving objects. The more motion information, the more confident the detection.

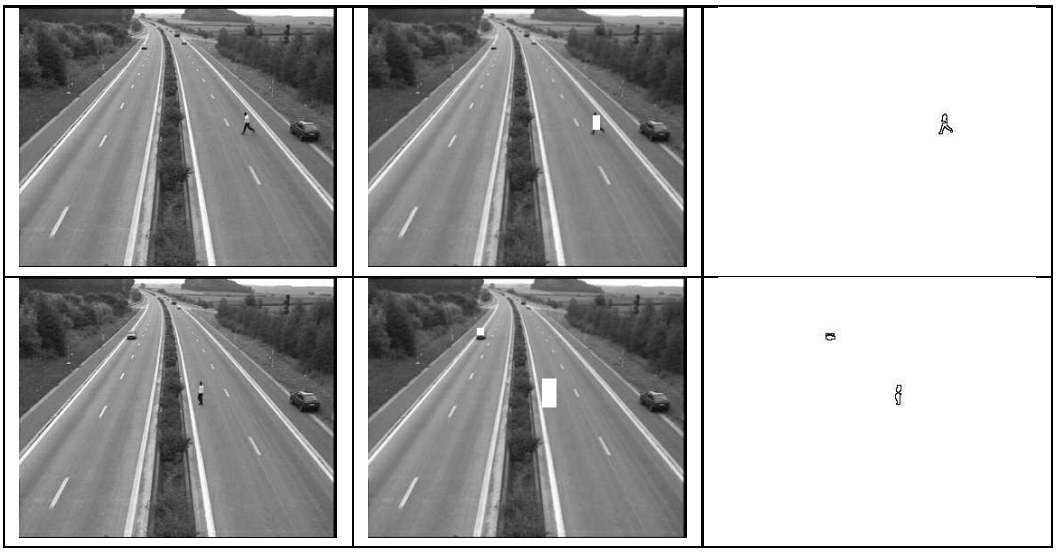


Figure 6: Highway sequence. The original sequence is composed of 340 frames. The two presented results are located at two different time instants: $t = 166$ (upper row) and $t = 204$ (lower row). The first column displays the original images. The middle column contains motion detection results (white blocks) using the quadtree segmentation. NFAs for the detected blocks at time instant $t = 166$ are 10^{-1} for the upper square block and 10^{-2} for the lower square block on the pedestrian. At time instant $t = 204$, NFAs are 10^{-2} for the block on the car, 0.9 for the upper block on the pedestrian and 0.5 for the lower block on the pedestrian. The third column presents motion detection results using the meaningful level lines segmentation. At time instant $t = 166$, NFAs are 10^{-3} for the region representing the upper part of the pedestrian and 10^{-6} for the region representing the legs. At time instant $t = 204$, NFAs are 10^{-2} for the car, 0.4 for the upper body part and 10^{-7} for the legs of the pedestrian.

6.2 Table tennis sequence

The second video sequence depicts a table tennis player. The camera is zooming out. The global dominant motion is estimated using the quadratic 2D motion model of Eq. (4). This model performs well (Fig. 7) although the hypothesis of planar background is violated by the table. Detection based on quadtree segmentation recovers the moving parts of the body. Using the quadtree segmentation, the body of the player is split into square blocks of different sizes. At time instant $t = 66$, the hand is undersegmented and the small blocks have NFAs ranging from 10^{-2} to 10^{-5}. The rest of the body is split into larger blocks that also include part of the background. NFAs are about 0.2. Again the segmentation with level lines allows us to get better motion detection results, since the candidate regions fit well to the player's contours. 12 regions formed by meaningful level lines are detected as moving for the player. NFAs reach 10^{-12} for the regions enclosing the player's body. The head is quasi static and therefore not detected.

At time instant $t = 75$, the behaviour of the quadtree segmentation is the same as previously. The player is now going to hit the ball and is therefore moving faster. As a consequence, smaller NFAs are observed. Motion is concentrated on the racket and on the hips of the player. NFAs for blocks corresponding to these regions are about 10^{-7}. With the level line segmentation, the player is segmented into 19 regions delimited by meaningful level lines which are detected as moving. Again, NFAs increase and are about 10^{-20} for regions on the players body. Let us note that, using the level lines segmentation, the head of the player is recovered. At this point of the sequence, the head is indeed moving but only slowly. This explains that it is not detected using blocks. The contribution of the segmentation using meaningful level lines to enforce spatial coherence of the motion detection is obvious here.

6.3 Road sequence

The last video sequence contains two cars driving down a road. The scene is shot from a helicopter. The camera tracks the cars driving down the road. Camera motion is complex and combines translation and rotation. The global motion estimation using the 2D quadratic motion model (Eq. (4) again performs well (see Fig. 9). Let us remind that the 8-parameter motion model is exact for a planar surface undergoing rigid motion. The planar assumption on the scene background is valid here.

Let us first comment the results obtained using the quadtree segmentation (Fig. 11, middle column). First of all, both cars are detected in all frames. In these experiments motion information is concentrated on the back of the cars (see Fig. 10). This is a consequence of the aperture problem. The spatial intensity gradient is collinear to the motion direction on the back of the car while it is orthogonal on the side. For the same reasons, some residual motion information is present at the front of the car.

The projections of the cars in the image sequence are almost at the same position since the camera is tracking the cars. The blocks corresponding to the cars are almost the same for the successive frames. This gives an impression of stability of the shape of the detected region.

For the large blocks, 32×32 pixels on top of the left car in the first frame, the NFA is 0.95 and is in fact almost not meaningful with respect to motion detection. NFAs for the left car blocks are

about 10^{-5} and for the right car blocks about 10^{-10}, which supplies high confidence on these two detections.

Using the maximal meaningful level line segmentation, the two cars are still detected. Let us note the important stability of the detected regions over time. Let us stress that the algorithm works only with three frames and that no temporal regularization is applied on the detected regions. The stability of the detected moving regions is also due to the stability of the image content which is well represented by the maximal meaningful level line segmentation. The level line segmentation preserves shape information over time. A more precise description of the lower part of the right car is impossible due to the absence of contrast and gray level saturation. Minimal NFAs are about 10^{-3} for the small car regions on the left and about 10^{-30} for the dark car regions on the right for all the five frames represented in Fig. 11.

Average total computation times for 352×288 images are about 2 seconds using the quadtree segmentation and about 4 seconds using the level line segmentation. The computation of the segmentation represents half of the total computation time : 1 second for the quadtree segmentation, 2 seconds for the level line segmentation. These results were obtained on a workstation with a 2.4 GHz processor.

An overall conclusion on experimental results is that there are almost no false alarms. The moving objects are detected and, using the level lines segmentation, the contours of moving objects are precisely located. Each moving region detection is associated with a confidence level through its NFA. Let us emphasize the variety of the content of the sequences. Detected moving objects differ in their size, nature and velocity. Camera motions range from static to complex combination of rotation and translation. These results illustrate the generality of our motion detection algorithm.

7 Conclusions

A general, original and efficient method for detecting moving regions in image sequences has been described. An automatic detection criterion is obtained using the *a contrario* decision approach. Three points characterize this algorithm. First of all, the proposed algorithm answers the questions about presence and position of motion. Second, no parameter tuning is required. The upper bound ε on the number of false alarms is set once and for all to 1. And finally, each detection is associated with a confidence level. These three points are essential for numerous applications and the effective use of image processing algorithms in practice.

An extension of this work is to track the detected moving regions. The first step is to initialize trajectories by detecting the temporal coherence of detected structures in time. Again, an *a contrario* model approach would be useful. Detecting the temporal coherence of moving regions in time would also enable to group regions according to their motion and would fill the gap between region representation and object representation. A successful initialization of trajectories would then allow to use tracking methods, as Kalman filtering or particle filtering to follow objects undergoing occlusion or trajectory crossing along the image sequence.

Acknowledgments

This work was partly supported by Région Bretagne and the IST European project LAVA. The “road” sequence was provided by INA.

References

[1] T. Aach and A. Kaup. Bayesian algorithms for change detection in image sequences using Markov random fields. *Signal Processing: Image Communication*, 7(2):147–160, 1995.

[2] A. Almansa, A. Desolneux, and S. Vamech. Vanishing point detection without any a priori information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(4):502–507, 2003.

[3] F. Attneave. Some informational aspects of visual perception. *Psychological Review*, 1954.

[4] M. J. Black, D. J. Fleet, and Y. Yacoob. Robustly estimating changes in image appearance. *Computer Vision and Image Understanding*, 78(1):8–31, 2000.

[5] F. Cao. Application of the Gestalt principles to the detection of good continuations and corners in image level lines. *Computing and Visualisation in Science*, 7:3–13, 2004.

[6] V. Caselles, T. Coll, and J. Morel. Topographic maps and local contrast changes in natural images. *International Journal of Computer Vision*, 33(1):5–27, 1999.

[7] I. Cohen and G. Medioni. Detecting and tracking moving objects for video surveillance. In *IEEE Conf. Computer Vision and Pattern Recognition*, Fort Collins CO, June 1999.

[8] G. Csurka and P. Bouthemy. Direct identification of moving objects and background from 2D motion models. In *7th International Conference on Computer Vision*, pages 566–571, Kerkyra, Greece, 1999.

[9] C. Demonceaux and D. Kachi-Akkouche. Motion detection using wavelet analysis and hierarchical Markov models. In *First International Workshop on Spatial Coherence for Visual Motion Analysis*, Prague, May 2004.

[10] A. Desolneux, L. Moisan, and J. Morel. Meaningful alignments. *International Journal of Computer Vision*, 40(1):7–23, 2000.

[11] A. Desolneux, L. Moisan, and J. Morel. Maximal meaningful events and applications to image analysis. *Annals of Statistics*, 31(6):1822–1851, 2003.

[12] A. Desolneux, L. Moisan, and J. Morel. *A theory of digital image analysis*. Lecture Notes in Mathematics, 2005.

[13] A. Desolneux, L. Moisan, and J.-M. Morel. Edge detection by Helmholtz principle. *Journal of Mathematical Imaging and Vision*, 14(3):271–284, May 2001.

[14] A. Desolneux, L. Moisan, and J.-M. Morel. A grouping principle and four applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(4):508–513, April 2003.

[15] W. Hoeffding. Probability inequalities for sum of bounded random variables. *Journal of the American Statistical Association*, (58):13–30, 1963.

[16] Y. Z. Hsu, H.-H. Nagel, and G. Rekers. New likelihood test methods for change detection in image sequences. *Computer Vision, Graphics, and Image Processing*, 26:73–106, 1984.

[17] M. Irani and P. Anandan. A unified approach to moving object detection in 2D and 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(6):577–589, June 1999.

[18] V. Jain, B. B. Kimia, and J. L. Mundy. Segregation of moving objects using elastic matching. In *Workshop on Spatial Coherence for Visual Motion Analysis*, Prague, May 2004.

[19] S. X. Ju, M. J. Black, and A. D. Jepson. Skin and bones: Multi-layer, locally affine, optical flow and regularization with transparency. In *Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96)*, pages 307–314, 1996.

[20] J. Kang, K. Gajera, I. Cohen, and G. Medioni. Detection of moving objects from overlapping EO and IR sensors. In *IEEE Conference on Computer Vision and Pattern Recognition*, Washington, DC, 2004.

[21] G. Kanizsa. *La Grammaire du Voir*. Diderot, 1996. Original title: *Grammatica del vedere*. French translation from Italian.

[22] J. Konrad. Motion detection and estimation. In A. C. Bovik, editor, *Handbook of Image and Video Processing*. Academic Press, 2000.

[23] J. L. Lisani and J.-M. Morel. Detection of major changes in satellite images. In *IEEE International Conference on Image Processing*, Barcelona, September 2003.

[24] D. G. Lowe. *Perceptual Organisation and Visual Recognition*. Kluwer Academic, 1985.

[25] A.-R. Mansouri and J. Konrad. Multiple motion segmentation with level sets. *IEEE Transactions on Image Processing*, 12(2):201–220, 2003.

[26] A. Mitiche and P. Bouthemy. Computation and analysis of image motion: A synopsis of current problems and methods. *International Journal of Computer Vision*, 19(1):29–55, 1996.

[27] P. Monasse. *Morphological Representation of Digital Images and Application to Registration*. PhD thesis, Université Paris IX Dauphine, 2000.

[28] P. Monasse and F. Guichard. Fast computation of a contrast invariant representation. *IEEE Transactions on Image Processing*, 9(5):860–872, 2000.

[29] P. Musé, F. Sur, F. Cao, and Y. Gousseau. Unsupervised thresholds for shape matching. In *IEEE International Conference on Image Processing*, Barcelona, September 2003.

[30] P. Musé, F. Sur, F. Cao, Y. Gousseau, and J. Morel. An a contrario decision method for shape element recognition. *Work in progress*, 2005.

[31] S. Nadimi and B. Bhanu. Physics-based cooperative sensor fusion for moving object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, Washington, DC, 2004.

[32] R. C. Nelson. Qualitative detection of motion by a moving observer. *International Journal of Computer Vision*, 7(1):33–46, 1991.

[33] J. Odobez and P. Bouthemy. Separation of moving regions from background in an image sequence acquired with a mobile camera. In H. Li, S. Sun, and H. Derin, editors, *Video Data Compression for Multimedia Computing*, chapter 8, pages 283–311. Kluwer Academic Publisher, 1997.

[34] J.-M. Odobez and P. Bouthemy. Detection of multiple moving objects using multiscale Markov random fields. In *1st IEEE International Conference on Image Processing*, volume 2, pages 257–261, Austin, Texas, November 1994.

[35] J.-M. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *Journal of Visual Communication and Image Representation*, 6(4):348–365, 1995.

[36] N. Paragios and R. Deriche. Geodesic active contour and level sets for the detection and tracking of moving objects. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(3):266–280, March 2000.

[37] R. Pless, T. Brodksy, and Y. Aloimonos. Detecting independent motion: The statistics of temporal continuity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):768–773, August 2000.

[38] P. L. Rosin. Thresholding for change detection. *Computer Vision and Image Understanding*, 86:79–95, May 2002.

[39] S. Sarkar, D. Majchrzak, and K. Korimilli. Perceptual organization based computational model for robust segmentation of moving objects. *Computer Vision and Image Understanding*, 86:141–170, 2002.

[40] P. Smith, T. Drummond, and R. Cipolla. Layered motion segmentation and depth ordering by tracking edges. *IEEE Transactions on Pattern Analysis and Machine Intellingence*, 26(4):479–494, April 2004.

[41] C. V. Stewart. Minpran: A new robust estimator for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(10):925–937, October 1995.

[42] W. B. Thompson, P. Lechleider, and E. R. Stuck. Detecting moving objects using the rigidity constraint. *IEEE Trans. Pattern Analysis and Machine Intelligence.*, 15(2):162–166, 1993.

[43] W. B. Thompson and T.-C. Pong. Detecting moving objects. *International Journal of Computer Vision*, 4:39–57, 1990.

[44] T. Veit, F. Cao, and P. Bouthemy. Probabilistic parameter-free motion detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, Washington, DC, 2004.

[45] L. Wixson. Detecting salient motion by accumulating directionally-consistent flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):774–780, August 2000.

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)
http://www.inria.fr
ISSN 0249-6399

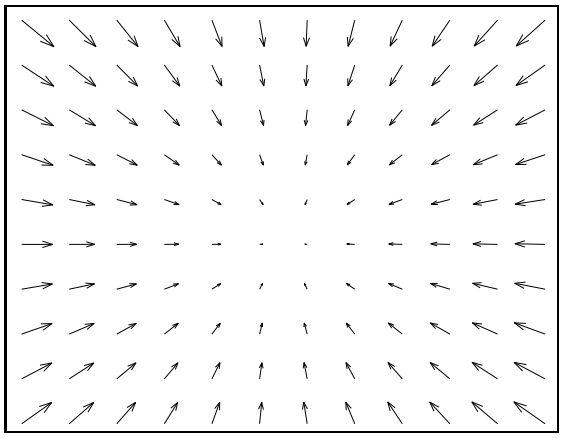


Figure 7: Estimated dominant motion for the sequence "Table tennis" between image at $t = 66$ and $t = 67$. The camera is zooming out. The computed quadratic motion model is represented by the velocity vectors evaluated at a given sampling rate over the image grid (with a scale factor of 5). Values of the parameters are $a_1 = 0.0386$, $a_2 = 0.0604$, $a_3 = -0.0181$, $a_4 = -0.0003$, $a_5 = -0.00171$, $a_6 = -0.01945$, $a_7 = 0.00001$, $a_8 = 0.00001$. The camera motion has been correctly estimated despite the presence of the moving player.

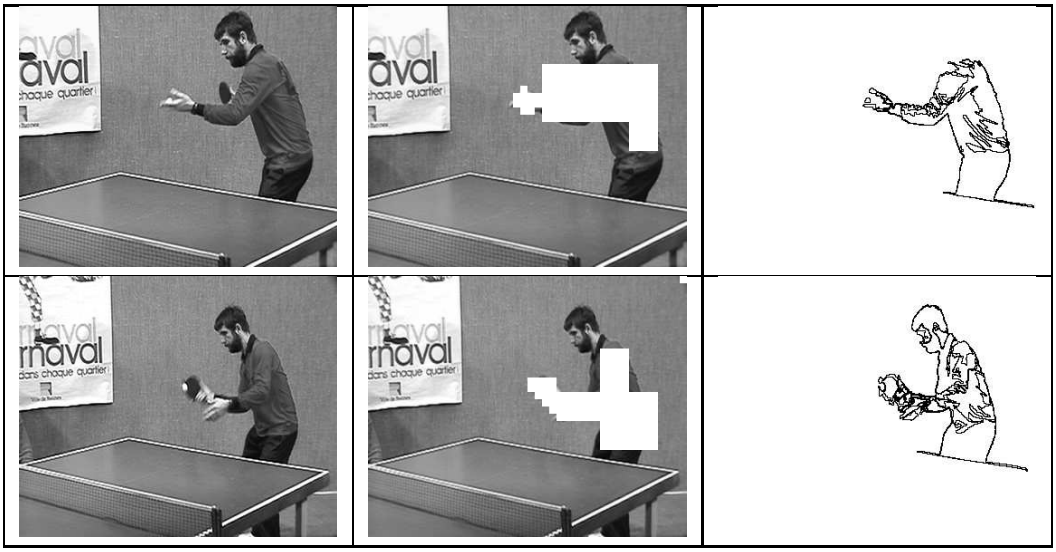


Figure 8: Table tennis sequence. This video sequence consists of 300 images. Presented results correspond to time instants $t = 66$ (first row) and $t = 75$ (second row). The first column shows original images, the second column contains motion detection results (white blocks) using the quadtree segmentation, and the third column presents motion detection results using meaningful level lines segmentation. At time instant $t = 66$, the detected blocks have NFAs ranging from 10^{-2} to 10^{-5}. At time instant $t = 75$, the NFAs of the detected blocks reach 10^{-7}. For the detected moving regions using the level line segmentation, NFAs reach 10^{-12} at time instant $t = 66$ and are about 10^{-20} for time instant $t = 75$.

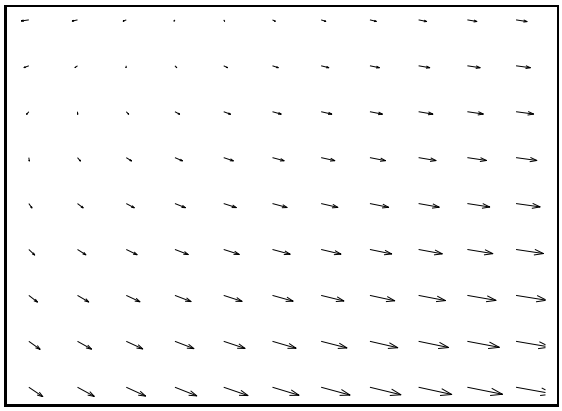


Figure 9: Estimated dominant motion for the “road” sequence between frames 290 and 291. The camera is placed on a helicopter. It is rotating and translating in order to track cars. Parameter values for the motion model are $a_1 = 7.1745$, $a_2 = 2.0394$, $a_3 = 0.02849$, $a_4 = 0.04075$, $a_5 = -0.0016$, $a_6 = 0.0122$, $a_7 = -0.00002$, and $a_8 = 0.00002$.

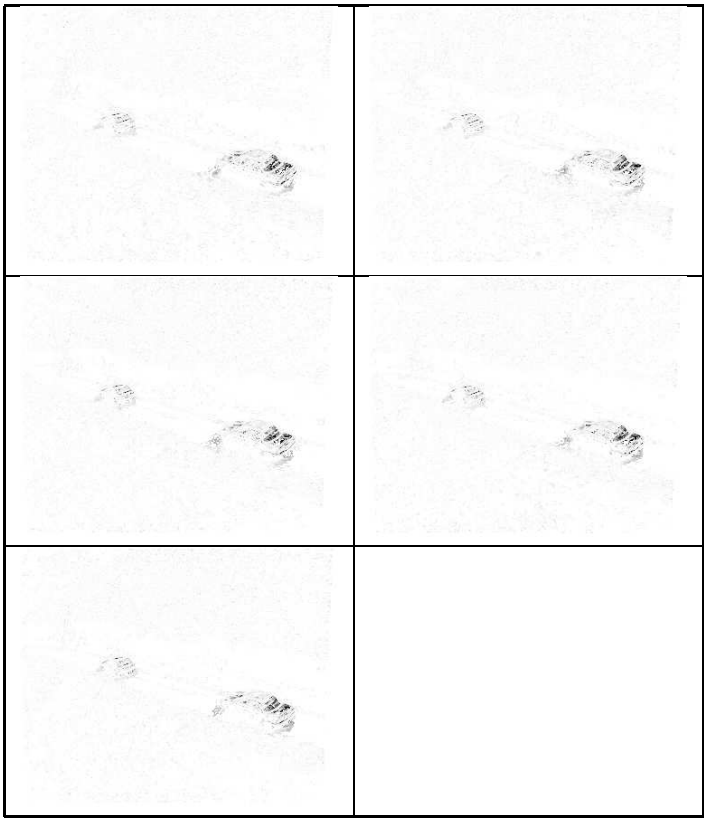


Figure 10: Road sequence. Maps of residual motion as defined by expression (8) for the five successive images displayed in Fig. 11. Motion information is concentrated at the back of the cars.

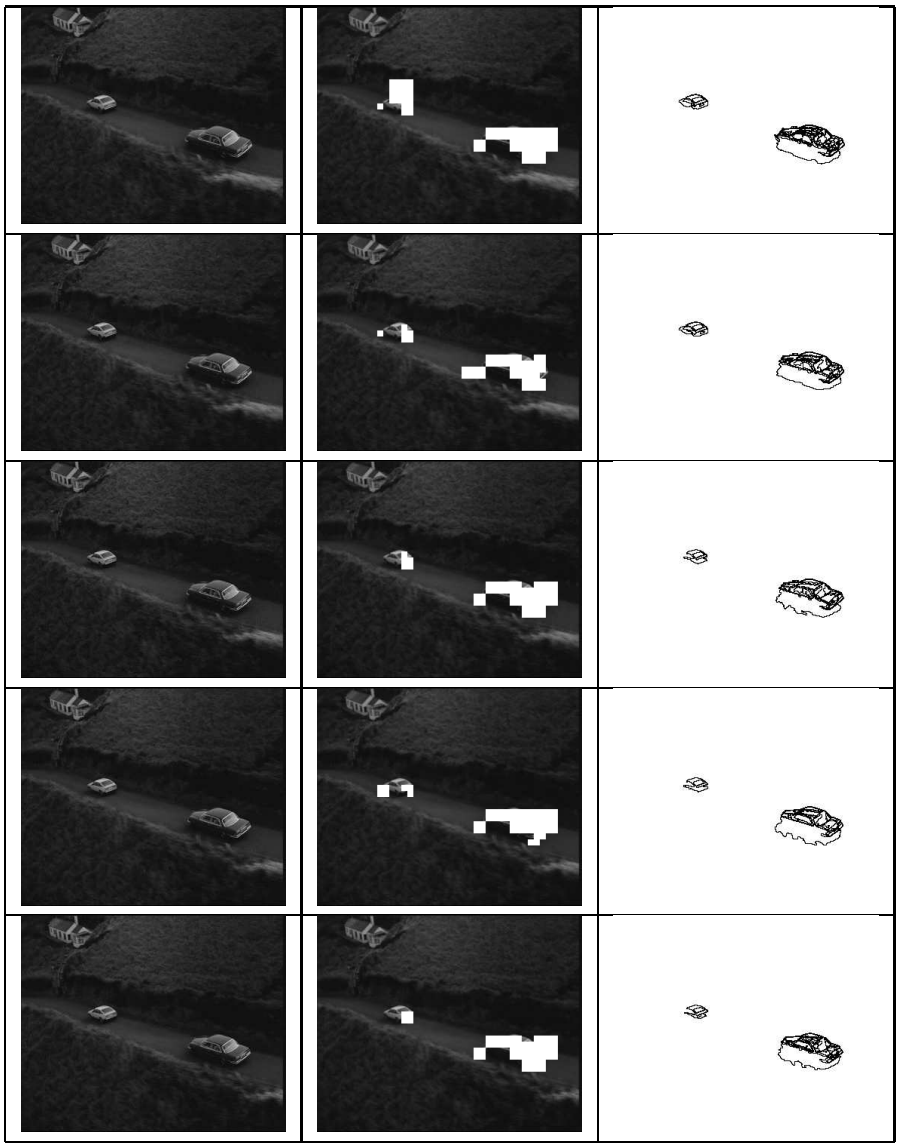


Figure 11: Road sequence. Left column: five consecutive images of the road sequence ($t = 289, \ldots, 293$). Middle column: detection results using quadtree segmentation. NFAs are about 10^{-5} for the left car and about 10^{-10} for the right car. In the first row, the large 32×32 block on the left car has an NFA of 0.95. Right column: detection results using maximal meaningful level lines segmentation. NFAs are about 10^{-3} for the left car and about 10^{-30} for the right car.