A Unified Approach to Moving Object Detection in 2D and 3D Scenes[*]

Michal Irani   P. Anandan

David Sarnoff Research Center
CN5300, Princeton, NJ 08543-5300, U.S.A.
Email: {michal,anandan}@sarnoff.com

Abstract

The detection of moving objects is important in many tasks. Previous approaches to this problem can be broadly divided into two classes: 2D algorithms which apply when the scene can be approximated by a flat surface and/or when the camera is only undergoing rotations and zooms, and 3D algorithms which work well only when significant depth variations are present in the scene and the camera is translating. In this paper, we describe a unified approach to handling moving object detection in both 2D and 3D scenes, with a strategy to gracefully bridge the gap between those two extremes. Our approach is based on a stratification of the moving object detection problem into scenarios and corresponding techniques which gradually increase in their complexity. Moreover, the computations required for the solution to the problem at one complexity level become the initial processing step for the solution at the next complexity level.

1 Introduction

The 2D motion observed in an image sequence is caused by 3D camera motion (the *ego-motion*) and by 3D motions of independently moving objects. The key step in moving object detection is accounting for the camera-induced image motion. After compensation for camera-induced image motion, the remaining residual motions must be due to moving objects.

The camera induced image motion depends both on the ego-motion parameters and the depth of each point in the scene. Estimating all of these physical parameters to account for the camera-induced motion is, in general, an inherently ambiguous problem [1]. When the scene contains large depth variations, these parameters can be recovered [18, 2, 14, 19, 20, 6]. We refer to these scenes as *3D scenes*. However, in *2D scenes* (namely when the scene is roughly planar, or distant from the camera, or when the camera is not translating), the recovery of the 3D camera and scene geometry is usually not robust or reliable [1]. 3D techniques are therefore applicable in scenarios containing dense 3D information, and very little independent motion. An effective approach to accounting for camera induced motion in 2D scenes is to model the image motion in terms of a global 2D parametric transformation [10, 5, 7, 15, 21, 3]. These techniques proved to be robust even in the presence of significant amount of independent motions. However, the 2D approach cannot be applied to cluttered 3D scenes.

Therefore, 2D and 3D techniques for detecting moving objects address two extreme cases in a continuum of scenarios: flat 2D scenes vs. cluttered 3D scenes. Each of these two classes of techniques fails on the other extreme case, or even on the intermediate case (when 3D information is *sparse* relative to amount of independent motion).

In this paper, we present an approach to unifying 2D and 3D techniques for moving object detection, with a strategy to gracefully bridge the gap between them. We present a set of techniques that progressively increase in their complexity, ranging from simple 2D techniques, to multi-layer 2D techniques, to the more complex 3D techniques. Moreover, the computations required for the solution to the problem at one complexity level become the initial processing step for the solution at the next complexity level. In particular, the 2D parametric motion compensation forms the basis to the solution of the multiple layer situation, and the single- or multiple-2D layered motion compensation forms the basis to the solution of the more general 3D case. Careful treatment is given to the intermediate case, when 3D information is sparse relative to amount of independent motion.

The goal in taking this approach is to develop a strategy for moving object detection, so that the analysis performed is tuned to match the complexity of the problem and the availability of information at any time. This paper describes the *core elements* of such a strategy. The integration of these elements into a single algorithm remains a task for our future research. For more details, refer to [9].

[*] This work was supported by ARPA under contract DAAA15-93-C-0061

2 2D Scenes

When the scene viewed from a moving camera is planar, or distant enough, or when the camera is not translating (only rotating/zooming), then the camera induced motion can be modeled by a *single global* 2D parametric transformation between a pair of successive image frames:

$$\begin{bmatrix} u(x,y) \\ v(x,y) \end{bmatrix} = \begin{bmatrix} p_1x + p_2y + p_5 + p_7x^2 + p_8xy \\ p_3x + p_4y + p_6 + p_7xy + p_8y^2 \end{bmatrix} \qquad (1)$$

where $(u(x,y), v(x,y))$ denotes the image velocity at the point (x,y). We refer to such cases as *2D scenes.* We use a previously developed method [4, 10] in order to compute the $2D$ parametric motion. This technique "locks" onto a "dominant" parametric motion between an image pair, even in the presence of independently moving objects. It does not require prior knowledge of their regions of support in the image plane [10]. When the dominant motion is that of the camera, all regions corresponding to static portions of the scene are in completely aligned as a result of the 2D image warping. Detection of moving objects is therefore performed by determining local misalignments [10] after the global 2D parametric registration.

Figure 1 shows an example of moving object detection in a "2D scene". The camera was both translating and rotating (camera jitter). The scene itself was not planar, but was distant enough (about 1 km away from the camera), for a 2D parametric transformation to account for the camera-induced motion between successive frames. Figure 1.e shows the detected moving object based on local misalignment analysis [10].

3 Multi-Planar Scenes

When the camera is translating, and the scene is not planar or is not sufficiently distant, then a *single* 2D parametric motion (Section 2) is insufficient for modeling the camera-induced motion. Aligning two images with respect to a *dominant* 2D parametric transformation may bring into alignment a large portion of the scene, which corresponds to a planar (or a remote) part of the scene. However, any other (e.g., near) portions of the scene that enter the field-of-view cannot be aligned by the dominant 2D parametric transformation. These out-of-plane scene points, although they have the same 3D motion as the planar points, have substantially different induced 2D motions. The *differences* in 2D motions are called *3D parallax motion.* Effects of parallax are only due to camera translation and 3D scene variations. Camera rotation or zoom do not cause parallax effects (see Section 4.1).

Figure 2 shows an example of a sequence where the effects of 3D parallax are evident. Figure 2.a and 2.b show two frames from a sequence with the same setting and scenario described in Figure 1, only in this case a frontal hill with bushes (which was much closer to the camera than the background scene) entered the field of view (FOV). Figure 2.c displays image regions which were found to be aligned after *dominant* 2D parametric registration (see Section 2). Clearly the global 2D alignment accounts for the camera-induced motion of the distant portion of the scene, but does *not* account for the camera-induced motion of the closer portion of the scene (the bushes).

Thus, simple 2D techniques, when applied to these types of scenarios, will not be able to distinguish between the independent car motion and the 3D parallax motion of the bush.

When the scene is piecewise planar, or is constructed of a few distinct portions at different depths, then the camera-induced motion can be accounted for by a few *layers* of 2D parametric transformations. This case is very typical of outdoor surveillance scenarios, especially when the camera FOV is narrow. The multi-layered approach is an extension of the simple 2D approach, and is implemented using a method similar to the sequential method presented in [10]: First, the dominant 2D parametric transformation between two frames is detected (Section 2). The two images are aligned accordingly, and the misaligned image regions are detected and segmented out (Figure 2.c). Next, the *same* 2D motion estimation technique is re-applied, but this time only to the segmented (misaligned) regions of the image, to detect the *next* dominant 2D transformation and its region of support within the image, and so on. For each additional layer, the two images are aligned according to the 2D parametric transformation of that layer, and the misaligned image regions are detected and segmented out (Figure 2.d). Each "2D layer" is continuously tracked in time by using the obtained segmentation masks. Moving objects are detected as image regions that are inconsistent with the image motion of any of the 2D layers. Such an example is shown in Figure 2.e.

A moving object is not detected as a layer by this algorithm if it is small. However, if the object is large, it may itself be detected as a 2D layer. To avoid this problem, additional cues are used to distinguish between moving objects and static scene layers [9]. The moving car shown in Figures 1 and 2 was successfully and continuously detected over the entire two-minute video sequence, which alternated between the single-layered case (i.e., no 3D parallax; frontal scene part was not visible in the FOV) and the two-layered case (i.e., existence of 3D parallax).

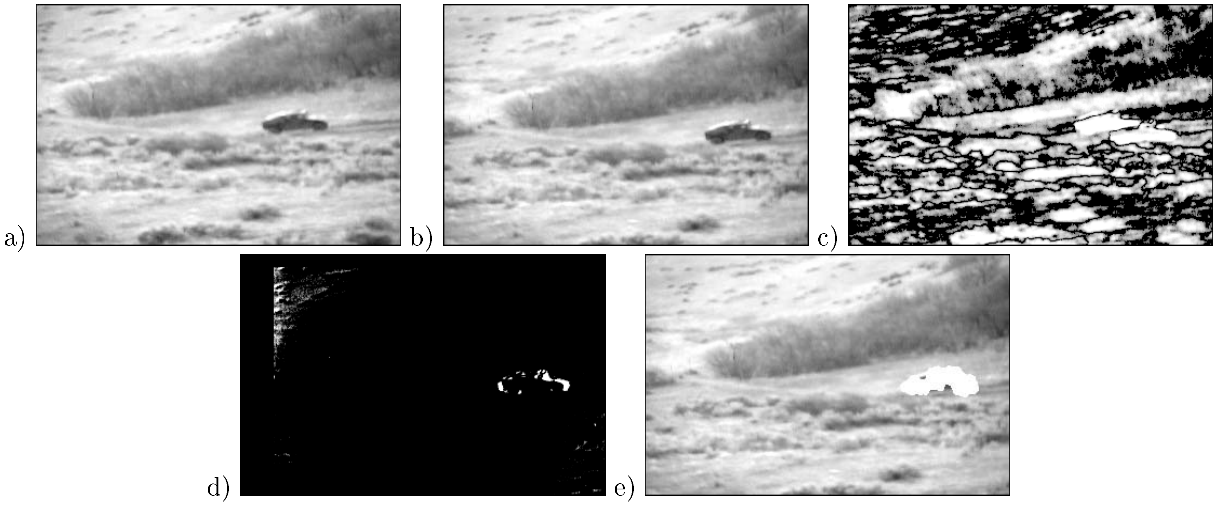

Figure 1: 2D moving object detection.
(a-b) Two frames in a sequence obtained by a translating and rotating camera. The scene itself was not planar, but was distant enough (about 1 km away from the camera) so that effects of 3D parallax were negligible. The scene contained a car driving on a road. (c) Intensity differences before dominant (background) 2D alignment. (d) Intensity differences after dominant (background) 2D alignment. Non-overlapping image boundaries were not processed. The 2D alignment compensates for the camera-induced motion, but not for the car's independent motion. (e) The detected moving object based on local misalignment *analysis. The white region signifies the detected moving object.*

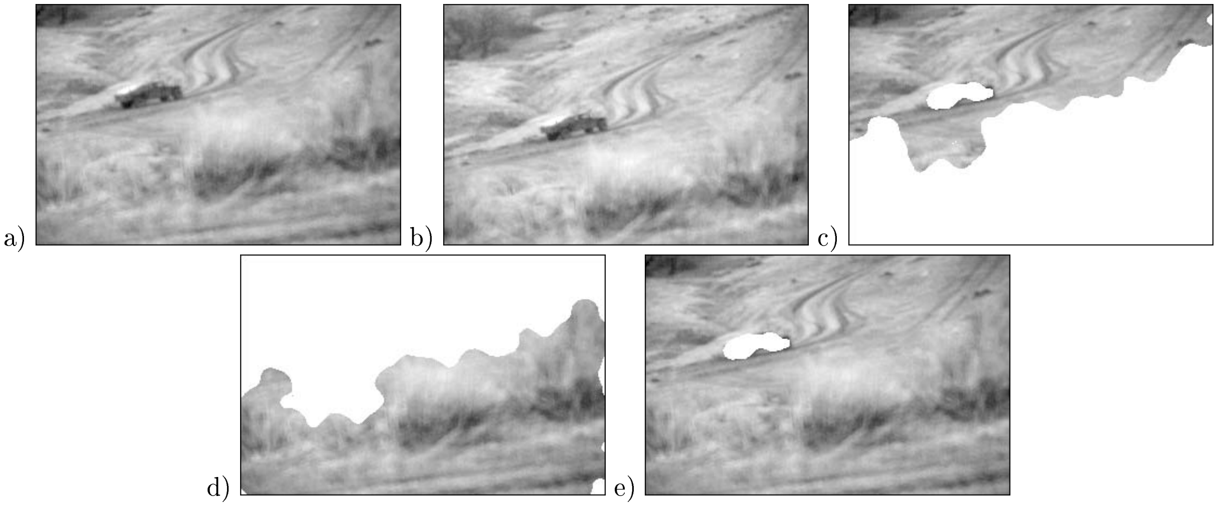

Figure 2: Layered moving object detection.
(a,b) Two frames in a sequence obtained by a translating and rotating camera. The FOV captures a distant portion of the scene (hills and road) as well as a frontal portion of the scene (bushes). The scene contains a car driving on a road. (c) The image region which corresponds to the dominant *2D parametric transformation. This region corresponds to the remote part of the scene. White regions signify image regions which were misaligned after performing global image registration according to the computed dominant 2D parametric transformation. These regions correspond to the car and the frontal part of the scene (the bushes). (d) The image region which corresponds to the* next *detected dominant 2D parametric transformation. This region corresponds to the frontal bushes. The 2D transformation was computed by applying the 2D estimation algorithm again, but this time only to the image regions highlighted in white in Fig. 2.c (i.e., only to image regions inconsistent in their image motion with the* first *dominant 2D parametric transformation). White regions in this figure signify regions inconsistent with the bushes' 2D transformation. These correspond to the car and to the remote parts of the scene. (e) The detected moving object (the car) highlighted in white.*

4 General 3D Scenes

While the single and multi-layered parametric registration methods are adequate to handle a large number of situations, there are cases when the 3D parallax *cannot* be modeled by 2D parametric transformations. An example is a *cluttered* scene (typically urban scenes or indoor scenes). In this section we present an approach to handling these more complex 3D scenes, which builds *on top* of the 2D analysis. For more details see [9].

4.1 The Plane+Parallax Decomposition

The key observation that enables us to extend the 2D parametric registration approach to general 3D scenes is the following: the plane registration process (using the dominant 2D parametric transformation) removes all effects of camera rotation, zoom, and calibration, *without explicitly computing them* [11, 13, 16, 17]. The residual image motion after the plane registration is due only to the *translational* motion of the camera and to the *deviations* of the scene structure from the planar surface. Hence, the residual motion is an *epipolar flow field.* This observation has led to the so-called "plane+parallax" approach to 3D scene analysis [12, 11, 13, 16, 17, 8].

Let $\vec{\mathbf{P}} = (X, Y, Z)^T$ and $\vec{\mathbf{P'}} = (X', Y', Z')^T$ denote the Cartesian coordinates of a scene point with respect to two different camera views, respectively. Let $\vec{\mathbf{p}} = (x, y)^T$ and $\vec{\mathbf{p'}} = (x', y')^T$ respectively denote the coordinates of the corresponding image points in the two image frames. Let $\vec{\mathbf{T}} = (T_x, T_y, T_z)$ denote the camera translation between the two views. Let Π denote a planar surface in the scene which is registered by the 2D parametric registration process mentioned in Section 2. It can be shown (see [8]) that the 2D image displacement of the point $\vec{\mathbf{P}}$ can be written as

$$\vec{\mathbf{u}} = (\vec{\mathbf{p'}} - \vec{\mathbf{p}}) = \vec{\mathbf{u}_\pi} + \vec{\mu},$$

where $\vec{\mathbf{u}_\pi}$ denotes the *planar* part of the $2D$ image motion (the homography due to Π), and $\vec{\mu}$ denotes the residual *planar parallax* $2D$ motion. The homography due to Π results in a 2D projective image motion that can be approximated by Equation (1). When $T_z \neq 0$:

$$\vec{\mathbf{u}_\pi} = (\vec{\mathbf{p'}} - \vec{\mathbf{p_w}}) \ \ ; \ \ \vec{\mu} = \gamma \frac{T_z}{d'_\pi}(\vec{\mathbf{e}} - \vec{\mathbf{p_w}}) \qquad (2)$$

where $\vec{\mathbf{p_w}}$ denotes the image point in the first frame which results from warping the corresponding point $\vec{\mathbf{p'}}$ in the second image, by the 2D parametric transformation of the plane Π. The $2D$ image coordinates of the epipole (or the *focus-of-expansion*, FOE) in the first frame are denoted by $\vec{\mathbf{e}}$, and d'_π is the perpendicular distance from the second camera center to the reference plane (see [8]). γ is a measure of the 3D projective structure of the point $\vec{\mathbf{P}}$. In the case when $T_z = 0$, the parallax motion μ has a slightly different form: $\mu = \frac{\gamma}{d'_\pi}\vec{\mathbf{t}}$, where $t = (T_x, T_y)^T$.

Since the residual parallax displacements after 2D planar alignment are due to the camera translational component alone, they form a radial field centered at the epipole/FOE. Violations in the epipolar constraint [18] can therefore be used to detect independently moving objects. Such a method, however, depends critically on the ability to accurately estimate the epipole. Epipole estimation can be very unreliable, in particular when the scene contains *sparse* parallax information and significant *independent motion* of objects in the scene. In the following section we develop an approach to moving object detection by directly comparing the parallax motion of pairs of points *without estimating the epipole.*

4.2 The parallax based rigidity constraint

Given the planar-parallax displacement vectors $\vec{\mu_1}$ and $\vec{\mu_2}$ of two image points $\vec{\mathbf{p_1}}$ and $\vec{\mathbf{p_2}}$ that belong to the static background scene, their *relative 3D projective structure* $\frac{\gamma_2}{\gamma_1}$ is given by (see [8]):

$$\frac{\gamma_2}{\gamma_1} = \frac{\vec{\mu_2}^T(\Delta\vec{\mathbf{p_w}})_\perp}{\vec{\mu_1}^T(\Delta\vec{\mathbf{p_w}})_\perp}. \qquad (3)$$

where, as shown in Figure 3, $\Delta\vec{\mathbf{p_w}} = \vec{\mathbf{p_{w_2}}} - \vec{\mathbf{p_{w_1}}}$ is the vector connecting the "warped" locations of the corresponding second frame points (as in Equation (2)), and $\vec{v}_\perp$ signifies a vector perpendicular to $\vec{v}$.

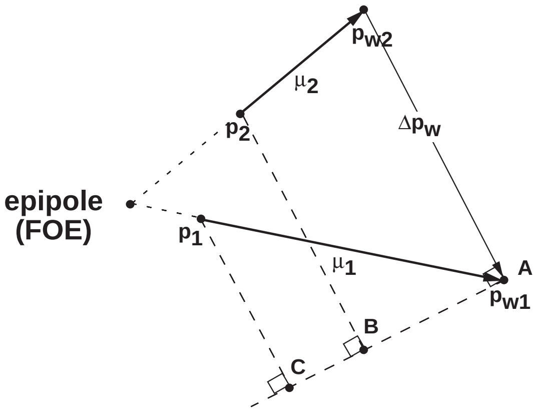

Figure 3: The pairwise parallax-based shape constraint. *This figure geometrically illustrates the relative structure constraint (Eq. 3):* $\frac{\gamma_2}{\gamma_1} = \frac{\vec{\mu_2}^T(\Delta\vec{\mathbf{p_w}})_\perp}{\vec{\mu_1}^T(\Delta\vec{\mathbf{p_w}})_\perp} = \frac{\mathbf{AB}}{\mathbf{AC}}$.

Note that this constraint directly relates the relative projective structure of two points to their parallax displacements alone: no camera parameters, in particular the *epipole* (FOE), are involved. Neither is any additional parallax information required at other image points. Application of this constraint to the recovery of 3D structure of the scene is described in [8]. Here we focus on its application to moving object detection.

Since the relative projective structure $\frac{\gamma_2}{\gamma_1}$ is invariant to camera motion, therefore, using Equation (3), for any two frames j and k (in addition to the reference frame) we get:

$$\frac{\vec{\mu_2}^{\mathbf{j}^\mathbf{T}} (\Delta \vec{\mathbf{p_w}})^{\mathbf{j}}_{\perp}}{\vec{\mu_1}^{\mathbf{j}^\mathbf{T}} (\Delta \vec{\mathbf{p_w}})^{\mathbf{j}}_{\perp}} - \frac{\vec{\mu_2}^{\mathbf{k}^\mathbf{T}} (\Delta \vec{\mathbf{p_w}})^{\mathbf{k}}_{\perp}}{\vec{\mu_1}^{\mathbf{k}^\mathbf{T}} (\Delta \vec{\mathbf{p_w}})^{\mathbf{k}}_{\perp}} = 0, \qquad (4)$$

where $\vec{\mu_1^{\mathbf{j}}}, \vec{\mu_2^{\mathbf{j}}}$ are the parallax displacement vectors of the two points between the reference frame and the jth frame, $\vec{\mu_1^{\mathbf{k}}}, \vec{\mu_2^{\mathbf{k}}}$ are the parallax vectors between the reference frame and the kth frame, and $(\Delta \vec{\mathbf{p_w}})^j, (\Delta \vec{\mathbf{p_w}})^k$ are the corresponding distances between the warped points as in Equation (3) and Figure 3.

Equation (4) provides a constraint on the parallax displacements of pairs of points over three frames, without referring to *camera geometry* (in particular the epipoles/FOE) or scene structure. The rigidity constraint (4) can therefore be applied to detect inconsistencies in the $3D$ motion of two image points (i.e., say whether the two image points are projections of $3D$ points belonging to a same or different $3D$ moving objects) based on their *parallax* motion among three (or more) frames alone, without relying on parallax information at other image points. An inconsistency measure is defined by the left-hand side of Equation (4), after multiplying by the denominators (to eliminate singularities). The larger this difference, the higher is the 3D-inconsistency of the two inspected image points.

Figures. 4 and 5 show examples of applying the parallax-based inconsistency measure to detect $3D$ inconsistencies.

The parallax-based rigidity constraint provides a useful mechanism for clustering (or segmenting) the "parallax" vectors (i.e., the residual motion after planar 2D registration) into consistent groups belonging to consistently $3D$ moving objects, even in cases where parallax information is minimal, and independent motion is not negligible. Note that this technique applies to uncalibrated cameras.

5 Conclusion

In this paper, we have described a unified approach to handling moving object detection in both 2D and 3D scenes, with a strategy to gracefully bridge the gap between those two extremes. The techniques that are described progressively increase in their complexity, while the computations required for the solution to the problem at one complexity level become the initial processing step for the solution at the next complexity level. Finally, we have presented the parallax-based rigidity constraint to detect $3D$-inconsistency when 3D parallax is sparse. This provides a natural way to *bridge* between $2D$ algorithms and $3D$ algorithms.

References

[1] G. Adiv. Inherent ambiguities in recovering 3D motion and structure from a noisy flow field. *PAMI*, 11:477–489, May 1989.

[2] Y. Aloimonos, editor. *Active Perception.* 1993.

[3] S. Ayer and H. Sawhney. Layered representation of motion video using robust maximum-likelihood estimation of mixture models and mdl encoding. In *ICCV*, pages 777–784, Cambridge, MA, June 1995.

[4] J. Bergen, P. Anandan, K. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *ECCV*, pages 237–252, May 1992.

[5] J. Bergen, P. Burt, R. Hingorani, and S. Peleg. A three-frame algorithm for estimating two-component image motion. *PAMI*, 14:886–895, September 1992.

[6] J. Costeira and T. Kanade. A multi-body factorization method for motion analysis. In *ICCV*, 1995.

[7] T. Darrell and A. Pentland. Robust estimation of a multi-layered motion representation. In *IEEE Workshop on Visual Motion*, pages 173–178, October 1991.

[8] M. Irani and P. Anandan. Parallax geometry of pairs of points for 3d scene analysis. In *ECCV*, 1996.

[9] M. Irani and P. Anandan. A unified approach to moving object detection in 2d and 3d scenes. In *DARPA IU Workshop*, 1996.

[10] M. Irani, B. Rousso, and S. Peleg. Computing occluding and transparent motions. *IJCV*, 12:5–16, 1994.

[11] M. Irani, B. Rousso, and S. Peleg. Recovery of ego-motion using image stabilization. In *CVPR*, pages 454–460, Seattle, Wa., June 1994.

[12] J. Koenderink and A. van Doorn. Representation of local geometry in the visual system. *Biol. Cybern.*, 55:367 – 375, 1987.

[13] R. Kumar, P. Anandan, and K. Hanna. Direct recovery of shape from multiple views: a parallax based approach. In *Proc 12th ICPR*, 1994.

[14] J. Lawn and R. Cipolla. Robust egomotion estimation from affine motion parallax. In *ECCV*, May 1994.

[15] F. Meyer and P. Bouthemy. Region-based tracking in image sequences. In *ECCV*, May 1992.

[16] H. Sawhney. 3d geometry from planar parallax. In *CVPR*, June 1994.

[17] A. Shashua and N. Navab. Relative affine structure: Theory and application to 3d reconstruction from perspective views. In *CVPR*, pages 483–489, 1994.

[18] W. Thompson and T. Pong. Detecting moving objects. *IJCV*, 4:29–57, 1990.

[19] P. Torr and D. Murray. Stochastic motion clustering. In *ECCV*, 1994.

[20] P. Torr, A. Zisserman, and S. Maybank. Robust detection of degenerate configurations for the fundamental matrix. In *ICCV*, pages 1037–1042, 1995.

[21] J. Wang and E. Adelson. Layered representation for motion analysis. In *CVPR*, pages 361–366, June 1993.

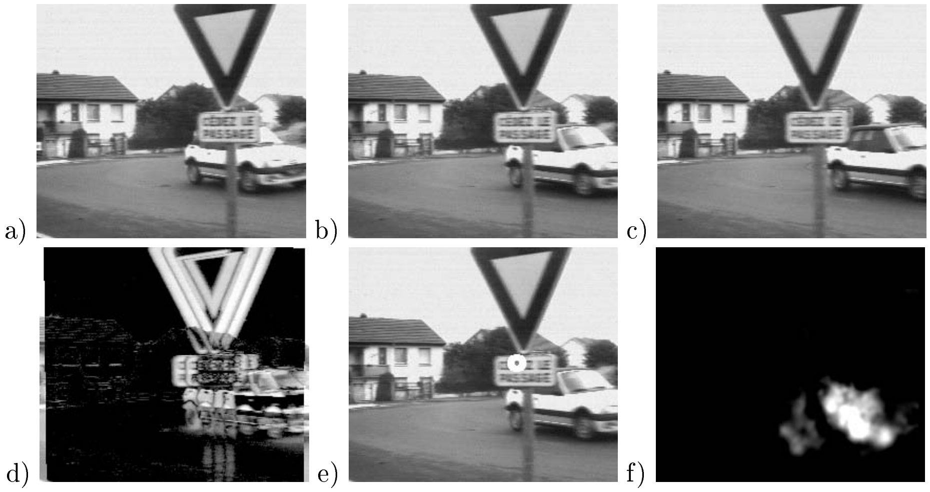

Figure 4: Moving object detection relying on a single parallax vector.
(a,b,c) Three image frames from a sequence obtained by a camera translating from left to right, inducing parallax motion of different magnitudes on the house, road, and road-sign. The car moves independently from left to right. The middle frame (Fig. 4.b) was chosen as the frame of reference. (d) Differences taken after $2D$ *image registration. The detected* $2D$ *planar motion was that of the house, and is canceled by the* $2D$ *registration. All other scene parts that have different* $2D$ *motions (i.e.,* parallax *motion or* independent *motion) are misregistered. (e) The selected point of reference (a point on the road-sign) highlighted by a white circle. (f) The measure of* $3D$-in*consistency of all points in the image with respect to the road-sign point. Bright regions indicate violations in* $3D$ *rigidity detected over three frames with respect to the selected road-sign point. These regions correspond to the car. Regions close to the image boundary were ignored. All other regions of the image appear to move* $3D$*-consistently with the road-sign point.*

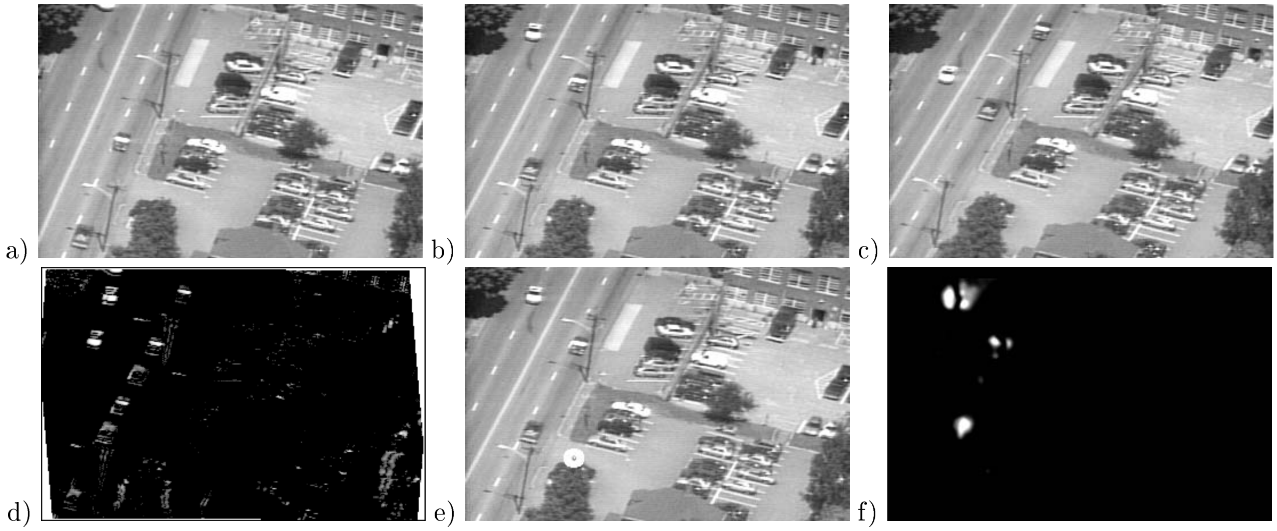

Figure 5: Moving object detection relying on a single parallax vector.
(a,b,c) Three image frames from a sequence obtained by a camera mounted on a helicopter (flying from left to right while turning), inducing some parallax motion (of different magnitudes) on the house-roof and trees (bottom of the image), and on the electricity poles (by the road). Three cars move independently on the road. The middle frame (Fig. 5.b) was chosen as the frame of reference. (d) Differences taken after $2D$ *image registration. The detected* $2D$ *planar motion was that of the ground surface, and is canceled by the* $2D$ *registration. All other scene parts that have different* $2D$ *motions (i.e.,* parallax *motion or* independent *motion) are misregistered. (e) The selected point of reference (a point on a tree at the bottom left of the image) highlighted by a white circle. (f) The measure of* $3D$-in*consistency of each point in the image with the tree point. Bright regions indicate violations in* $3D$ *rigidity detected over three frames with respect to the selected tree point. These regions correspond to the three cars (in the reference image). Regions close to the image boundary were ignored. All other regions of the image appear to move* $3D$*-consistently with the tree point.*