

# Joint Multiregion Segmentation and Parametric Estimation of Image Motion by Basis Function Representation and Level Set Evolution

Carlos Vázquez, *Member, IEEE*, Amar Mitiche, *Member, IEEE*, and Robert Laganière, *Member, IEEE*

**Abstract**—The purpose of this study is to investigate a variational method for joint segmentation and parametric estimation of image motion by basis function representation of motion and level set evolution. The functional contains three terms. One term is of classic regularization to bias the solution toward a segmentation with smooth boundaries. A second term biases the solution toward a segmentation with boundaries which coincide with motion discontinuities, following a description of motion discontinuities by a function of the image spatio-temporal variations. The third term refers to region information and measures conformity of the parametric representation of the motion of each region of segmentation to the image spatio-temporal variations. The components of motion in each region of segmentation are represented as functions in a space generated by a set of basis functions. The coefficients of the motion components considered combinations of the basis functions are the parameters of representation. The necessary conditions for a minimum of the functional, which are derived taking into consideration the dependence of the motion parameters on segmentation, lead to an algorithm which condenses to concurrent curve evolution, implemented via level sets, and estimation of the parameters by least squares within each region of segmentation. The algorithm and its implementation are verified on synthetic and real images using a basis of cosine transforms.

**Index Terms**—Motion estimation, motion segmentation, basis function representation of motion, parametric motion model, curve evolution, level sets.

## 1 INTRODUCTION

PARAMETRIC models of image motion are important for several reasons: 1) They are an economical representation of image motion. They can also be an accurate representation because a few coefficients of a suitable model can describe finely graded motion (see our first experiment for an example). As such, they are of prime interest in video processing for interpolation and coding of image sequences [1]. 2) They are a basis for image segmentation, where an image is partitioned into regions of differing parameters (see our experiments for examples). Motion-based segmentation is of broad interest in vision because it can serve various processes such as classification, tracking, and 3D interpretation [2]. An image segmented into regions where motion is represented by model parameters can be efficiently coded in accordance with the most recent MPEG standards [3]. 3) Parametric modeling allows assignment of motion estimates, within each region of segmentation, to segments

which have no significant texture (see our last experiment for an example). Therefore, parametric modeling is a means of propagating motion estimation from textured to nontextured parts of an image without blurring motion boundaries. By contrast, smoothing, which is the basis for regularization in variational methods, will either propagate motion estimation to nontextured segments but blur motion boundaries (isotropic smoothing in the Horn and Schunck type methods), or preserve motion boundaries but prevent propagation of estimation to nontextured segments (anisotropic smoothing in methods such as [4], [5]).

Several studies have shown the relevance of parametric models for motion estimation and segmentation [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17]. Most regarded estimation and segmentation as successive processes when these are mutually dependent. Also, they were limited to constant and affine models. When segmentation, but not estimation, is the main purpose, the constant and affine models are sufficient to segment a variety of flow fields, such as those generated by most familiar man-made rigid objects. In such a case, more complex models, although they compute more accurate motion, can lead to fragmented segmentation. However, more complex models are required for both estimation and segmentation when the variations of motion are complex within the regions of segmentation (see our second and third experiments). More complex models are also needed when the quality of the estimated motion is important, e.g., for interpolation or 3D interpretation. Nonparametric variational motion methods such as [4], [5], which use a Mumford-Shah type functional [18], can yield very good, boundary preserving estimates of motion but do not address explicit segmentation.

• C. Vázquez is with Advanced Video Systems, Communication Research Centre (CRC), 3701 Carling Ave., PO Box 11490, Stn. H. Ottawa, Ontario, Canada, K2H 8S2. E-mail: carlos.vazquez@crc.ca.

• A. Mitiche is with INRS-EMT, Institut National de la Recherche Scientifique, Université du Québec, Place Bonaventure, Suite 6900, 800 de la Gauchetière West, Montréal, QC, H5A 1K6, Canada. E-mail: mitiche@emt.inrs.ca.

• R. Laganière is with the School of Information Technology and Engineering (SITE), Faculty of Engineering, University of Ottawa, 161 Louis Pasteur, PO Box 450, Stn A, Ottawa, ON, K1N 6N5, Canada. E-mail: laganiere@site.uottawa.ca.

Manuscript received 15 June 2004; revised 6 June 2005; accepted 19 Sept. 2005; published online 13 Mar. 2006.

Recommended for acceptance by M. Srinivasan.

For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0309-0604.

Joint parametric estimation and segmentation was the focus of the study in [17], which envisaged representing each component of motion by a general linear combination of a set of parameters. However, the coefficients of the combinations were not related to any specific model of motion which would define them as functions of image coordinates. The affine model was given as an example and the constant model implemented.

Current parametric methods of motion estimation use the constant and affine models. Although these models can provide an adequate representation of motion generated by movement of rigid objects in space with smooth surface variations, they are insufficient to describe motion in general. For complex motion, more descriptive models must be used.

The subject of this study is joint segmentation and parametric estimation of image motion. We address the problem of segmentation into an arbitrary but fixed, known, number of regions and estimation of the parameters describing the motion in each region. This study investigates joint estimation and multiregion segmentation by representation of the components of motion as functions in a space  $\Theta$  generated by a set of basis functions  $\phi_1, \dots, \phi_n$ .  $\Theta = \text{span}(\phi_1, \dots, \phi_n)$ . The coefficients of the motion components considered combinations of the basis functions are the parameters of representation. Models can be obtained from a wide variety of basis functions, such as polynomials (the constant and affine models use polynomials of degree 0 and 1, respectively), and cosines/sines (which we implemented).

The representation by basis functions is used in a functional, minimized via curve evolution and level sets, which contains three terms. One term is of classic regularization to bias the solution toward a segmentation with smooth boundaries. A second term biases the solution toward a partition with boundaries which coincide with motion discontinuities, following a description of motion discontinuities by a scalar function of the image spatio-temporal variations. The third term refers to region information and measures conformity of the parametric representation of the motion of each region of segmentation to the image spatio-temporal variations.

The necessary conditions for a minimum of the functional lead to an algorithm which condenses to concurrent curve evolution, implemented by level sets, and estimation of the parameters by least squares within each region of segmentation. We account for the dependence of the motion parameters on the segmentation in the derivation of the necessary conditions. The derivation shows that this dependence does not lead to extra terms in the curve evolution equations [19]. Therefore, the assumption that the parameters are independent of the segmentation [16], [17], [20] is not necessary. Also, it is not necessary to determine the parameters before segmentation [14], [15], or call for a greedy algorithm [21] to iterate two steps, one of optimization with respect to the model parameters with curves fixed, the other of curve evolution with the model parameters fixed.

To verify the algorithm and its implementation, we show several examples with cosine transforms basis functions on synthetic and real image sequences.

The remainder of this paper is organized as follows: Section 2 describes the basis function representation of motion. Section 3 states the problem as functional minimization. In Section 4, the two-region segmentation case is considered. The Euler-Lagrange descent equations and their

level sets expression are developed. In Section 5, the formulation is extended to multiple regions. Section 6 deals with related computational issues. Experimental results for both synthetic and real image sequences are presented in Section 7, and Section 8 contains a conclusion and a discussion on future work.

## 2 BASIS FUNCTION REPRESENTATION OF MOTION

The goal is to write a variational formulation of the problem of joint segmentation and parametric estimation of image motion, and to derive the corresponding Euler-Lagrange equations to be solved by curve evolution via level sets. For simplicity, we will first treat the two-region case. A generalization to multiple regions will be given in Section 5.

Let  $\mathbf{I} : \Omega \times ]0, T[ \rightarrow \mathbb{R}$  be a time-varying image function, where  $\Omega$  is an open subset of  $\mathbb{R}^2$  and  $T$  the duration of the sequence. Let  $\tilde{\gamma} : [0, 1] \rightarrow \Omega$  be a simple closed plane curve parameterized by arc parameter  $s \in [0, 1]$ . Let  $\mathbf{R}_{\tilde{\gamma}}$  be the region enclosed by  $\tilde{\gamma}$ , and  $\mathcal{R} = \{\mathbf{R}_1 = \mathbf{R}_{\tilde{\gamma}}, \mathbf{R}_2 = \mathbf{R}_{\tilde{\gamma}}^c\}$  the corresponding partition of the image domain.

Each component of image motion within a region of segmentation is described by a parametric model. The model is based on a basis function representation of motion. More specifically, let  $\mathbf{u}_i(\mathbf{x}) = (u_{i1}(\mathbf{x}), u_{i2}(\mathbf{x}))^T$ ;  $i = 1, 2$  be the velocity vector describing the motion at point  $\mathbf{x} \in \mathbf{R}_i$ . Let  $\Theta$  be the space generated by  $M$  basis functions  $\{\theta_j(\mathbf{x})\}_{j=1, \dots, M}$ .  $\Theta = \text{span}\{\theta_1(\mathbf{x}), \dots, \theta_M(\mathbf{x})\}$ . The horizontal and vertical components of the velocity vector field describing the motion in each region are considered functions in this space:

$$\mathbf{u}_i(\mathbf{x}) = \alpha_i^T \boldsymbol{\theta}(\mathbf{x}), \quad i = 1, 2. \quad (1)$$

The parameter matrix  $\alpha_i$  is of size  $M \times 2$  and contains the horizontal and vertical motion parameters for region  $\mathbf{R}_i$ :

$$\alpha_i = \begin{pmatrix} \alpha_{i11} & \alpha_{i21} & \cdots & \alpha_{iM1} \\ \alpha_{i12} & \alpha_{i22} & \cdots & \alpha_{iM2} \end{pmatrix}^T, \quad (2)$$

where  $\alpha_{ijl}$  is the coefficient corresponding to function  $\theta_j$  for the horizontal ( $l = 1$ ) or vertical ( $l = 2$ ) component of vector field  $\mathbf{u}_i$ , describing the motion of region  $\mathbf{R}_i$ . Vector  $\boldsymbol{\theta}(\mathbf{x})$  is of size  $M$ , containing the values of the basis functions at  $\mathbf{x}$ :

$$\boldsymbol{\theta}(\mathbf{x}) = (\theta_1(\mathbf{x}) \quad \theta_2(\mathbf{x}) \quad \cdots \quad \theta_M(\mathbf{x}))^T. \quad (3)$$

This model generalizes the constant and affine models of image motion generally in use. It affords a vast choice of models, such as polynomials (the constant and affine models are in this class), cosine/sine (which we implemented), among others, and includes almost every parametric model used previously for 2D motion description [22].

## 3 VARIATIONAL FORMULATION

Joint motion estimation and segmentation is stated as the minimization of an energy functional containing three terms:

$$\mathcal{E} = \mathcal{E}_m + \lambda \mathcal{E}_s + \epsilon \mathcal{E}_b, \quad (4)$$

where  $\mathcal{E}_m$  and  $\mathcal{E}_s$  measure, respectively, the accuracy of motion estimation and the smoothness of segmentation boundaries, and  $\mathcal{E}_b$  is related to a description of motion boundaries. The positive constants  $\lambda$  and  $\epsilon$  weigh the relative contribution of the three terms.

### 3.1 Motion Accuracy Term

Several energy functionals have been used for motion segmentation. For instance, in [17], the authors use the angle between the spatio-temporal gradient and the velocity vector as a measure of motion estimation accuracy to separate motions with distinct directions. The study in [16] considered the case of two regions and used the norm of the difference between least squares estimates of motion within the regions, a method which is quite extricate to extend to an arbitrary number of regions [23]. Here, we will simply use the optical flow constraint of Horn and Schunck [24]:

$$\mathcal{E}_m = \frac{1}{2} \sum_{i=1}^2 \int_{\mathbf{R}_i} (\nabla \mathbf{I} \cdot \mathbf{u}_i(\mathbf{x}) + \mathbf{I}_t(\mathbf{x}))^2 d\mathbf{x}, \quad (5)$$

where  $\cdot$  denotes the dot product,  $\nabla \mathbf{I}$  the spatial gradient of  $\mathbf{I}(\mathbf{x}, t)$ , and  $\mathbf{I}_t = \frac{\partial \mathbf{I}}{\partial t}$ . The velocity vector function  $\mathbf{u}_i(\mathbf{x})$  is as in (1) and is completely described by the matrix of parameters  $\alpha_i$  and the function space  $\Theta$ .

Substituting (1) in (5), (5) is rewritten, in terms of curve  $\tilde{\gamma}$ , as:

$$\mathcal{E}_m(\tilde{\gamma}) = \frac{1}{2} \sum_{i=1}^2 \int_{\mathbf{R}_i} (e_i(\mathbf{x})) d\mathbf{x}, \quad (6)$$

where  $e_i$ , the motion estimation accuracy term for region  $\mathbf{R}_i$ , is defined by:

$$e_i(\mathbf{x}) = (\nabla \mathbf{I} \cdot \alpha_i^T \theta(\mathbf{x}) + \mathbf{I}_t(\mathbf{x}))^2. \quad (7)$$

It is important to note that the motion parameters are functions of  $\tilde{\gamma}$ , i.e.,  $\alpha_i = \alpha_i(\tilde{\gamma})$ . This dependency will be taken into account explicitly in subsequent analysis, in contrast with others [14], [17] where motion parameters are either estimated prior to segmentation or are considered variables independent of  $\tilde{\gamma}$ .

### 3.2 Smoothness of Region Boundaries

With respect to regularization, we need only be concerned with region boundaries since the motion model regularizes the velocity field. Assuming that region boundaries are smooth, we use the boundary length functional:

$$\mathcal{E}_s(\tilde{\gamma}) = \frac{1}{2} \oint_{\tilde{\gamma}} ds. \quad (8)$$

### 3.3 Motion Boundary Term

This term follows a description of motion boundaries which we used in [25] for tracking. Derivation of the Horn and Schunck gradient equation,

$$\nabla \mathbf{I} \cdot \mathbf{u}_i(\mathbf{x}) + \mathbf{I}_t(\mathbf{x}) = 0 \quad (9)$$

with respect to spatial coordinates, and assuming that image motion is locally approximately constant, i.e.,  $\nabla u_1 \approx 0$  and  $\nabla u_2 \approx 0$ , everywhere except at motion boundaries, gives

$$\nabla(\nabla \mathbf{I} \cdot \mathbf{u}_i + \mathbf{I}_t) = \mathbf{H} \mathbf{u}_i + \nabla(\mathbf{I}_t) \quad (10)$$

$$\approx 0, \quad (11)$$

where  $\mathbf{H}$  is the Hessian of  $\mathbf{I}$ . Combining (9) and (10), we obtain a scalar function  $g$ :

$$g(\mathbf{x}) = \|\mathbf{H} \mathbf{I}_t - \nabla \mathbf{I} \cdot (\mathbf{H}^* \nabla \mathbf{I}_t)\|, \quad (12)$$

where  $|\cdot|$  denotes the determinant function if the argument is a matrix and absolute value if the argument is a scalar. The matrix  $\mathbf{H}^*$  is the transpose of the matrix of cofactors of  $\mathbf{H}$  and has the property  $\mathbf{H}^* \mathbf{H} = |\mathbf{H}| \mathcal{I}$ , where  $\mathcal{I}$  is the identity matrix. Function  $g$  is an indicator of motion boundaries. Inside motion regions, where motion is smooth, this function takes small values. At motion boundaries, where the optical flow constraint equation is violated, it takes large values.

The motion boundary term is then defined as the negative of the line integral of  $g$  along the region boundary:

$$\mathcal{E}_b(\tilde{\gamma}) = -\frac{1}{2} \oint_{\tilde{\gamma}} g(\mathbf{x}) ds. \quad (13)$$

### 3.4 Global Functional

Using (6), (8), and (13), the functional to minimize is:

$$\begin{aligned} \mathcal{E}(\tilde{\gamma}, \alpha_1(\tilde{\gamma}), \alpha_2(\tilde{\gamma})) &= \frac{1}{2} \sum_{i=1}^2 \int_{\mathbf{R}_i} (e_i(\mathbf{x})) d\mathbf{x} \\ &+ \lambda \frac{1}{2} \oint_{\tilde{\gamma}} ds - \epsilon \frac{1}{2} \oint_{\tilde{\gamma}} g(\mathbf{x}) ds. \end{aligned} \quad (14)$$

The problem of joint segmentation and parametric estimation of motion consists of determining  $\tilde{\gamma}$  such that:

$$\tilde{\gamma} = \arg \min_{\tilde{\gamma}} \mathcal{E}(\tilde{\gamma}, \alpha_1(\tilde{\gamma}), \alpha_2(\tilde{\gamma})). \quad (15)$$

This formulation can be viewed as a motion-based level-set implementation of the Mumford-Shah functional [18]. The Mumford-Shah functional for image brightness segmentation contains a data term, a boundary length term, and a smoothness term. Although functional (14) deals with motion segmentation, rather than with image brightness segmentation, it contains a data term (the motion accuracy term), a boundary length term (common in variational segmentation methods), and the smoothness term is implicit in the use of the parametric motion model. The motion boundary term in (14), although not essential, contains information computed along boundaries to complement the information about the regions enclosed by these boundaries.

Functional (14) can also be viewed as a generalization of the level-set implementation of Mumford-Shah functional in [26]. It is a generalization in two respects. First, it is a generalization considering that motion is not an observed variable but a variable to be estimated along with the segmentation. Second, it uses a general parametric model rather than the piecewise constant model.

## 4 FUNCTIONAL MINIMIZATION

The purpose in this section is to derive the equations of minimization of the objective functional (14) with respect to the motion parameters and curve  $\tilde{\gamma}$ . These equations will show that the corresponding algorithm consists of concurrent curve evolution implemented via level sets and least squares estimations of the motion parameters with each region of segmentation.

### 4.1 Motion Parameters Computation

Motion parameters for each region are computed by minimizing the motion accuracy term for that region:

$$\alpha_i = \arg \min_{\tilde{\alpha}_i} \frac{1}{2} \int_{\mathbf{R}_i} (e_i(\mathbf{x}, \tilde{\alpha}_i)) d\mathbf{x} \quad i = 1, 2, \quad (16)$$

where we made explicit the dependency of  $e_i$  on  $\alpha_i$  to facilitate reading of the equation and, for simplicity, we dropped the dependency of the motion parameters on  $\tilde{\gamma}$  from the notation.

The necessary condition corresponding to (16) can be written as:

$$\frac{\partial}{\partial \alpha_i} \left( \frac{1}{2} \int_{\mathbf{R}_i} (e_i(\mathbf{x}, \alpha_i)) d\mathbf{x} \right) = 0 \quad i = 1, 2. \quad (17)$$

Expanding the derivative in (17), we can write this equation in matrix form as:

$$\frac{\partial}{\partial \alpha_i} \left( \frac{1}{2} \int_{\mathbf{R}_i} (e_i(\mathbf{x}, \alpha_i)) d\mathbf{x} \right) = \mathbf{B}(\mathbf{R}_i) \beta_i + \mathbf{d}(\mathbf{R}_i), \quad (18)$$

where the matrices and vectors involved in the equation are defined as follows:

- Matrix  $\mathbf{B}(\mathbf{R}_i)$  is a  $2M \times 2M$  matrix formed by the vertical and horizontal concatenation of  $4 M \times M$  submatrices  $\mathbf{B}_{rc}$ :

$$\mathbf{B}(\mathbf{R}_i) = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \mathbf{B}_{21} & \mathbf{B}_{22} \end{bmatrix}, \quad (19)$$

where each submatrix has elements of the form:

$$B_{rc}[m, n] = \int_{\mathbf{R}_i} \mathbf{I}_r(\mathbf{x}) \mathbf{I}_c(\mathbf{x}) \theta_m(\mathbf{x}) \theta_n(\mathbf{x}) d\mathbf{x} \quad (20)$$

for  $m = 1, \dots, M$ ,  $n = 1, \dots, M$ , and  $\mathbf{I}_l$  being the spatial derivative of  $\mathbf{I}$  on the horizontal ( $l = 1$ ) and vertical ( $l = 2$ ) directions.

- Vector  $\beta_i$  is a  $2M \times 1$  vector constructed by vertically concatenating the motion parameters  $\alpha_{i1}$  and  $\alpha_{i2}$  corresponding to the horizontal and vertical dimensions, respectively:

$$\beta_i[m] = \alpha_{im1}, \quad (21)$$

$$\beta_i[M + m] = \alpha_{im2} \quad (22)$$

for  $m = 1, \dots, M$ .

- Vector  $\mathbf{d}(\mathbf{R}_i)$  is a  $2M \times 1$  vector with elements:

$$d_i[m] = \int_{\mathbf{R}_i} \mathbf{I}_t(\mathbf{x}) \mathbf{I}_1(\mathbf{x}) \theta_m(\mathbf{x}) d\mathbf{x}, \quad (23)$$

$$d_i[M + m] = \int_{\mathbf{R}_i} \mathbf{I}_t(\mathbf{x}) \mathbf{I}_2(\mathbf{x}) \theta_m(\mathbf{x}) d\mathbf{x} \quad (24)$$

for  $m = 1, \dots, M$ .

Therefore, the necessary conditions with respect to motion parameters  $\alpha_i$  are:

$$\mathbf{B}(\mathbf{R}_i) \beta_i = -\mathbf{d}(\mathbf{R}_i) \quad (25)$$

and the motion parameters are computed by:

$$\beta_i = -(\mathbf{B}(\mathbf{R}_i))^{-1} \mathbf{d}(\mathbf{R}_i). \quad (26)$$

*In summary*, the motion parameters are determined for each region by linear least-squares (26). For each region, the elements of matrix  $\mathbf{B}$  (19) and vector  $\mathbf{d}$  are given explicitly in terms of the image spatio-temporal variations and the basis functions (20), (23), and (24).

## 4.2 Curve Evolution Equation

We proceed now to derive the necessary condition for a minimum of (14) with respect to  $\tilde{\gamma}$ . Define a vector  $\tilde{\mathbf{h}} \in \mathbb{R}^{2M}$  having elements:

$$\tilde{h}_m(\mathbf{x}) = \mathbf{I}_1(\mathbf{x}) \theta_m(\mathbf{x}), \quad (27)$$

$$\tilde{h}_{M+m}(\mathbf{x}) = \mathbf{I}_2(\mathbf{x}) \theta_m(\mathbf{x}) \quad (28)$$

for  $m = 1, \dots, M$ .

We start with the integral inside the sum of the first term on the right-hand side of (14) for  $i = 1$ . Using the definition of  $\mathbf{B}(\mathbf{R}_{\tilde{\gamma}})$ ,  $\mathbf{d}(\mathbf{R}_{\tilde{\gamma}})$ ,  $\beta_1$ ,  $\tilde{\mathbf{h}}$ , and (25), this integral can be written as:

$$\int_{\mathbf{R}_{\tilde{\gamma}}} (e_1(\mathbf{x})) d\mathbf{x} \quad (29)$$

$$= \int_{\mathbf{R}_{\tilde{\gamma}}} \left( \beta_1^T \tilde{\mathbf{h}} + \mathbf{I}_t(\mathbf{x}) \right)^2 d\mathbf{x}$$

$$= \int_{\mathbf{R}_{\tilde{\gamma}}} \left( \beta_1^T \tilde{\mathbf{h}} \tilde{\mathbf{h}}^T \beta_1 + 2\mathbf{I}_t(\mathbf{x}) \beta_1^T \tilde{\mathbf{h}} + (\mathbf{I}_t(\mathbf{x}))^2 \right) d\mathbf{x}$$

$$= \int_{\mathbf{R}_{\tilde{\gamma}}} \left( \beta_1^T \left( \tilde{\mathbf{h}} \tilde{\mathbf{h}}^T \beta_1 + \mathbf{I}_t(\mathbf{x}) \tilde{\mathbf{h}} \right) + \mathbf{I}_t(\mathbf{x}) \beta_1^T \tilde{\mathbf{h}} \right.$$

$$\left. + (\mathbf{I}_t(\mathbf{x}))^2 \right) d\mathbf{x}$$

$$= \beta_1^T (\mathbf{B}(\mathbf{R}_{\tilde{\gamma}}) \beta_1 + \mathbf{d}(\mathbf{R}_{\tilde{\gamma}})) + \beta_1^T \mathbf{d}(\mathbf{R}_{\tilde{\gamma}})$$

$$+ \int_{\mathbf{R}_{\tilde{\gamma}}} (\mathbf{I}_t(\mathbf{x}))^2 d\mathbf{x}$$

$$= \beta_1^T \mathbf{d}(\mathbf{R}_{\tilde{\gamma}}) + \int_{\mathbf{R}_{\tilde{\gamma}}} (\mathbf{I}_t(\mathbf{x}))^2 d\mathbf{x}, \quad (30)$$

where we have used (25) and equalities:

$$\int_{\mathbf{R}_{\tilde{\gamma}}} \tilde{\mathbf{h}} \tilde{\mathbf{h}}^T d\mathbf{x} = \mathbf{B}(\mathbf{R}_{\tilde{\gamma}}), \quad \int_{\mathbf{R}_{\tilde{\gamma}}} \mathbf{I}_t(\mathbf{x}) \tilde{\mathbf{h}} d\mathbf{x} = \mathbf{d}(\mathbf{R}_{\tilde{\gamma}}).$$

Functional derivation with respect to  $\tilde{\gamma}$  of (30) gives:

$$\begin{aligned} \frac{\partial}{\partial \tilde{\gamma}} \left( \int_{\mathbf{R}_{\tilde{\gamma}}} (e_1(\mathbf{x})) d\mathbf{x} \right) &= \frac{\partial \beta_1^T}{\partial \tilde{\gamma}} \mathbf{d}(\mathbf{R}_{\tilde{\gamma}}) \\ &+ \beta_1^T \frac{\partial \mathbf{d}(\mathbf{R}_{\tilde{\gamma}})}{\partial \tilde{\gamma}} + \frac{\partial}{\partial \tilde{\gamma}} \left( \int_{\mathbf{R}_{\tilde{\gamma}}} (\mathbf{I}_t(\mathbf{x}))^2 d\mathbf{x} \right). \end{aligned} \quad (31)$$

Next, we determine the derivatives with respect to  $\tilde{\gamma}$  in (31). From (25), we have that:

$$\frac{\partial \beta_1^T}{\partial \tilde{\gamma}} = \left( -\frac{\partial \mathbf{d}^T(\mathbf{R}_{\tilde{\gamma}})}{\partial \tilde{\gamma}} - \beta_1^T \frac{\partial \mathbf{B}(\mathbf{R}_{\tilde{\gamma}})}{\partial \tilde{\gamma}} \right) (\mathbf{B}(\mathbf{R}_{\tilde{\gamma}}))^{-1}. \quad (32)$$

From the definition of  $\mathbf{d}$  and  $\mathbf{B}$  and using the result in [21] for the functional derivative of the integral of a scalar function independent of the region enclosed by the curve, we have:

$$\frac{\partial}{\partial \tilde{\gamma}} \left( \int_{\mathbf{R}_{\tilde{\gamma}}} (\mathbf{I}_t(\mathbf{x}))^2 d\mathbf{x} \right) = (\mathbf{I}_t(\tilde{\gamma}))^2 \tilde{\mathbf{n}}(\tilde{\gamma}), \quad (33)$$

$$\frac{\partial \mathbf{d}(\mathbf{R}_{\tilde{\gamma}})}{\partial \tilde{\gamma}} = \mathbf{I}_t(\tilde{\gamma}) \tilde{\mathbf{h}}(\tilde{\gamma}) \otimes \tilde{\mathbf{n}}(\tilde{\gamma}), \quad (34)$$

$$\frac{\partial \mathbf{B}(\mathbf{R}_{\tilde{\gamma}})}{\partial \tilde{\gamma}} = \tilde{\mathbf{h}}(\tilde{\gamma}) \tilde{\mathbf{h}}^T(\tilde{\gamma}) \otimes \tilde{\mathbf{n}}(\tilde{\gamma}), \quad (35)$$

where  $\tilde{\mathbf{n}}$  is the external unit normal function to  $\tilde{\gamma}$  and  $\otimes$  denotes tensor product.

Substitution of (34) and (35) in (32) results in an equation which, when substituted along with (33) and (34) in (31), gives:

$$\begin{aligned}
& \frac{\partial}{\partial \bar{\gamma}} \left( \int_{\mathbf{R}_{\bar{\gamma}}} (e_1(\mathbf{x})) d\mathbf{x} \right) \\
&= \left( (-\mathbf{I}_t \tilde{\mathbf{h}}^T - \beta_1^T \tilde{\mathbf{h}} \tilde{\mathbf{h}}^T) (\mathbf{B}(\mathbf{R}_{\bar{\gamma}}))^{-1} \mathbf{d}(\mathbf{R}_{\bar{\gamma}}) \right. \\
&\quad \left. + \mathbf{I}_t \beta_1^T \tilde{\mathbf{h}} + \mathbf{I}_t^2 \right) \bar{\mathbf{n}}(\bar{\gamma}) \\
&= \left( \beta_1^T \tilde{\mathbf{h}} \tilde{\mathbf{h}}^T \beta_1 + 2\mathbf{I}_t \beta_1^T \tilde{\mathbf{h}} + \mathbf{I}_t^2 \right) \bar{\mathbf{n}}(\bar{\gamma}) \\
&= \left( \beta_1^T \tilde{\mathbf{h}}(\bar{\gamma}) + \mathbf{I}_t(\bar{\gamma}) \right)^2 \bar{\mathbf{n}}(\bar{\gamma}) \\
&= (\nabla \mathbf{I} \cdot \boldsymbol{\alpha}_1^T \boldsymbol{\theta}(\bar{\gamma}) + \mathbf{I}_t(\bar{\gamma}))^2 \bar{\mathbf{n}}(\bar{\gamma}) \\
&= e_1(\bar{\gamma}) \bar{\mathbf{n}}(\bar{\gamma}).
\end{aligned} \tag{36}$$

We note that the tensor products which appears in (34) and (35) are no longer in the final equation because of the vector-vector multiplications that yield scalar values. For example, while introducing (34) into (31), the vector product  $\beta_1^T \tilde{\mathbf{h}}$  results in a scalar value, making the tensor product no longer present.

Equation (36) shows that the dependence of the motion parameters on the segmentation does not result in extra terms [19] in the functional derivative with respect to  $\bar{\gamma}$  of the (parameter dependent) motion accuracy term.

In a similar fashion, one obtains the derivative of the integral on  $\mathbf{R}_{\bar{\gamma}}^c$  ( $i = 2$ ):

$$\frac{\partial}{\partial \bar{\gamma}} \int_{\mathbf{R}_{\bar{\gamma}}^c} (e_2(\mathbf{x})) d\mathbf{x} = -e_2(\bar{\gamma}) \bar{\mathbf{n}}(\bar{\gamma}). \tag{37}$$

The minus sign in the right-hand side of (37) is due to the fact that the unit external normal of  $\mathbf{R}_{\bar{\gamma}}^c$  is  $-\bar{\mathbf{n}}$ .

Finally, inclusion of the functional derivative of the second and third terms in (14) (derivations can be found, for instance, in [27]) gives:

$$\begin{aligned}
\frac{\partial \mathcal{E}(\bar{\gamma})}{\partial \bar{\gamma}} &= \frac{1}{2} (e_1(\bar{\gamma}) - e_2(\bar{\gamma}) + \lambda \kappa(\bar{\gamma})) \\
&\quad - \epsilon \nabla g(\bar{\gamma}) \cdot \bar{\mathbf{n}}(\bar{\gamma}) - \epsilon g(\bar{\gamma}) \kappa(\bar{\gamma}) \bar{\mathbf{n}}(\bar{\gamma}),
\end{aligned} \tag{38}$$

where  $\kappa$  is the mean curvature function of  $\bar{\gamma}$ . The necessary condition for a minimum of (14) with respect to  $\bar{\gamma}$  follows setting to zero the right-hand side of (38).

Embedding  $\bar{\gamma}$  in a one-parameter family of curves  $\bar{\gamma}(s, t)$ , indexed by (algorithmic) time  $t$ , the gradient descent equation corresponding to (38) is given by:

$$\frac{d\bar{\gamma}}{dt} = -\frac{\partial \mathcal{E}(\bar{\gamma})}{\partial \bar{\gamma}} \tag{39}$$

or, explicitly,

$$\begin{aligned}
\frac{d\bar{\gamma}}{dt} &= -\frac{1}{2} \left( e_1(\bar{\gamma}) - e_2(\bar{\gamma}) + \lambda \kappa(\bar{\gamma}) \right. \\
&\quad \left. - \epsilon \nabla g(\bar{\gamma}) \bar{\mathbf{n}}(\bar{\gamma}) - \epsilon g(\bar{\gamma}) \kappa(\bar{\gamma}) \right) \bar{\mathbf{n}}(\bar{\gamma})
\end{aligned} \tag{40}$$

with the motion parameters computed simultaneously within each region of segmentation by least squares:

$$\begin{aligned}
\mathbf{B}(\mathbf{R}_{\bar{\gamma}}) \beta_1 &= -\mathbf{d}(\mathbf{R}_{\bar{\gamma}}), \\
\mathbf{B}(\mathbf{R}_{\bar{\gamma}}^c) \beta_2 &= -\mathbf{d}(\mathbf{R}_{\bar{\gamma}}^c).
\end{aligned} \tag{41}$$

### 4.3 Level Set Implementation

Let  $\bar{\gamma}(s, t)$  be represented at all times  $t$  as the zero level-set of a function  $\phi: \mathbb{R}^2 \times \mathbb{R} \rightarrow \mathbb{R}$ . One can show [28] that if the evolution of  $\bar{\gamma}$  is described by the equation:

$$\frac{d\bar{\gamma}(s, t)}{dt} = F(\bar{\gamma}(s, t)) \bar{\mathbf{n}}(s, t), \tag{42}$$

where  $F$  is a real-valued function defined on  $\mathbb{R}^2$ , then the evolution equation of function  $\phi$ , with the convention that  $\phi > 0$  inside the zero level-set, is:

$$\frac{\partial \phi(\mathbf{x}, t)}{\partial t} = F(\mathbf{x}, t) \|\nabla \phi(\mathbf{x}, t)\|. \tag{43}$$

In our case, the evolution equation for function  $\phi$  is obtained from (40) as:

$$\begin{aligned}
\frac{\partial \phi(\mathbf{x}, t)}{\partial t} &= -\frac{1}{2} (e_1(\mathbf{x}) - e_2(\mathbf{x}) + \lambda \kappa(\mathbf{x})) \\
&\quad - \epsilon \nabla g(\mathbf{x}) \cdot \bar{\mathbf{n}}(\mathbf{x}) - \epsilon g(\mathbf{x}) \kappa(\mathbf{x}) \|\nabla \phi(\mathbf{x}, t)\|.
\end{aligned} \tag{44}$$

The curvature function  $\kappa$  is given in terms of the level set function by:

$$\kappa(\mathbf{x}) = \operatorname{div} \left( \frac{\nabla \phi(\mathbf{x})}{\|\nabla \phi(\mathbf{x})\|} \right) \tag{45}$$

and the normal unit vector  $\bar{\mathbf{n}}$  is given by:

$$\bar{\mathbf{n}}(\mathbf{x}) = \frac{\nabla \phi(\mathbf{x})}{\|\nabla \phi(\mathbf{x})\|}. \tag{46}$$

To implement the level set equations, one must define extension velocities [28]. For instance, the extension velocity at a point is the velocity at the point closest to it on the evolving curve. Extension velocities can also be defined so that the level set function is, at all times, the distance function from the evolving curve. Both of these definitions, often implemented via narrow banding, require that the initial curves intersect the regions they segment. This is important when a region has unconnected components. An alternative, robust to initialization and which we use in our experiments, extends the expression of velocity on the evolving curve to the image domain [14], [25].

## 5 MULTIPLE REGION EXTENSION

The purpose in this section is to extend the two-region formulation to a fixed (and known) but otherwise arbitrary number of regions  $N > 2$ .

Let  $\bar{\gamma}_i|_{i=1, \dots, N-1}$  be a family of simple closed planar curves, their interior defining regions  $\mathbf{R}_i|_{i=1, \dots, N-1}$ . Region  $\mathbf{R}_N$  will be formed by the intersection of the exteriors of all curves:

$$\mathbf{R}_N = \bigcap_{i=1}^{N-1} \mathbf{R}_i^c. \tag{47}$$

We follow our view of segmentation as regularized clustering [29] and define  $N - 1$  energy functionals, each involving two regions, namely, the interior of a curve and its complement:

$$\begin{aligned}
\mathcal{E}_\Omega(\bar{\gamma}_i | \mathbf{I}) &= \frac{1}{2} \int_{\mathbf{R}_i} e_i(\mathbf{x}) d\mathbf{x} + \frac{1}{2} \int_{\mathbf{R}_i^c} \psi_i(\mathbf{x}) d\mathbf{x} \\
&\quad + \lambda \frac{1}{2} \oint_{\bar{\gamma}_i} ds - \epsilon \frac{1}{2} \oint_{\bar{\gamma}_i} g(\mathbf{x}) ds,
\end{aligned} \tag{48}$$

where  $e_i$  is defined as in (7), and  $\psi_i(\mathbf{x}) = \min_{j \neq i}(e_j(\mathbf{x}))$ . Segmentation of the image then results from the following set of simultaneous minimizations:

$$\tilde{\mathbf{R}}_i = \arg \min_{\mathbf{R}_i} (\mathcal{E}_\Omega(\tilde{\gamma}_i | \mathbf{I})), \quad i \in [1, N-1] \quad (49)$$

with the  $N$ th region defined by (47). The evolution equations of curves  $\tilde{\gamma}_i$ ,  $i = 1, \dots, N-1$ , for the minimizations (49) are given by:

$$\begin{aligned} \frac{d\tilde{\gamma}_i}{dt} = & -\frac{1}{2}(e_i(\tilde{\gamma}_i) - \psi_i(\tilde{\gamma}_i) + \lambda\kappa_i(\tilde{\gamma}_i)) \\ & - \epsilon \nabla g(\tilde{\gamma}_i) \cdot \tilde{\mathbf{n}}_i(\tilde{\gamma}_i) - \epsilon g(\tilde{\gamma}_i) \kappa_i(\tilde{\gamma}_i) \tilde{\mathbf{n}}_i(\tilde{\gamma}_i). \end{aligned} \quad (50)$$

Provided that the curves do not intersect initially, these evolution equations for the curves produce a partition of the image domain at convergence [29]. The motion within each region of the partition is described by motion parameters obtained from (25). The level set equations corresponding to (50) are given by:

$$\begin{aligned} \frac{\partial \phi_i(\mathbf{x}, t)}{\partial t} = & -\frac{1}{2}(e_i(\mathbf{x}) - \psi_i(\mathbf{x}) + \lambda\kappa_i(\mathbf{x})) \\ & - \epsilon \nabla g(\mathbf{x}) \cdot \tilde{\mathbf{n}}_i(\mathbf{x}) - \epsilon g(\mathbf{x}) \kappa_i(\mathbf{x}) \|\nabla \phi_i(\mathbf{x}, t)\|. \end{aligned} \quad (51)$$

## 6 IMPLEMENTATION ISSUES

There are several computations which we perform efficiently.

### 6.1 Incremental Least Squares Estimation

The least squares estimate of motion parameters is computed incrementally [7]. At each iteration, the new estimate is computed as the sum of the previous one and the one resulting from (25) by considering the image at time  $t = 0$  and its motion-compensated version, computed from the image at time  $t = 1$  and the currently available motion parameters:

$$\alpha_i^{(n+1)} = \alpha_i^{(n)} - \Delta \alpha_i^{(n)}, \quad i = 1, \dots, N, \quad (52)$$

where the superscript  $(n)$  denotes iterations and the deviation is computed by using (25):

$$\Delta \alpha_i^{(n)} = \left( \hat{\mathbf{B}}(\mathbf{R}_i) \right)^{-1} \hat{\mathbf{d}}(\mathbf{R}_i). \quad (53)$$

The matrix  $\hat{\mathbf{B}}$  and the vector  $\hat{\mathbf{d}}$  are obtained from (20), (23), and (24) where the temporal derivative,  $\mathbf{I}_t$ , is approximated by  $\tilde{\mathbf{I}}_t$ , a temporally compensated first difference using two frames (indexed by 0 and 1) and the currently available motion parameters:

$$\tilde{\mathbf{I}}_t^{(n)}(\mathbf{x}, 0) = \mathbf{I}\left(\mathbf{x} + \left(\alpha_i^{(n)}\right)^T \boldsymbol{\theta}(\mathbf{x}), 1\right) - \mathbf{I}(\mathbf{x}, 0). \quad (54)$$

This strategy improves the approximation of the temporal derivative, which becomes more reliable. At each iteration, we estimate a motion of lesser extent, so that the optical flow constraint is valid.

### 6.2 Computation of Derivatives

A recognized important aspect of any implementation of optic flow calculation is the computation of derivatives. In particular, the computation of the temporal derivative of the image sequence is a critical step to a successful implementation. There are two calculations that involve the computation of derivatives:

1. *Computation of matrix  $\mathbf{B}$  and vector  $\mathbf{d}$ .* We have computed the horizontal (vertical) derivatives as an average over three rows (columns) of a central first difference:

$$\tilde{\nabla}_x \mathbf{I}(x, y) = \frac{1}{6} \sum_{i=-1}^1 (\mathbf{I}(x+1, y+i) - \mathbf{I}(x-1, y+i)),$$

$$\tilde{\nabla}_y \mathbf{I}(x, y) = \frac{1}{6} \sum_{i=-1}^1 (\mathbf{I}(x+i, y+1) - \mathbf{I}(x+i, y-1)).$$

The temporal derivatives are approximated effectively by a spatial average of the temporally compensated first difference approximation  $\tilde{\mathbf{I}}_t$ :

$$\tilde{\mathbf{I}}_t^{(n)}(\mathbf{x}) = \sum_{\mathbf{y} \in \mathcal{N}(\mathbf{x})} w_{\mathbf{y}} \left( \tilde{\mathbf{I}}_t^{(n)}(\mathbf{x} + \mathbf{y}) \right), \quad \mathbf{x} \in \mathbf{R}_i,$$

where  $\mathcal{N}$  denotes the set of points consisting of  $\mathbf{x}$  and its four neighbors, and  $w_{\mathbf{y}}$  is the weight associated to  $\mathbf{y}$ .

2. *Computation of the motion accuracy term for the evolution of the curve  $\tilde{\gamma}$ .* We compute this term as the displaced frame difference (DFD) for the region:

$$\begin{aligned} \nabla \mathbf{I} \cdot \alpha_i^T \boldsymbol{\theta}(\mathbf{x}) &= \frac{d\mathbf{I}(\mathbf{x}(t), t)}{dt} \\ &\approx \mathbf{I}(\mathbf{x} + \alpha_i^T \boldsymbol{\theta}(\mathbf{x}), 1) - \mathbf{I}(\mathbf{x}, 0). \end{aligned}$$

### 6.3 Estimation by Gradual Increase of Model Complexity

One drawback of motion models of high degree of freedom is that they can explain motion from different objects with a unique set of parameters, making the convergence of the process slow and sometimes erroneous. To deal with this problem, we have adopted a gradually increasing model complexity. For the first stages of the process, we use a simple model, which will constraint the estimated motion to be very smooth. This allows the algorithm to well separate very differently moving objects. In a second stage, we use the segmentation obtained at the previous step as initial segmentation and introduce a more complex model with a higher number of degrees of freedom. This new model refines the initial segmentation and estimated motion to better reflect the motions in the scene. Once the segmentation has captured most of the image structure, an even more complex model can be used which, further, will better explain the motion of the segmented objects.

## 7 EXPERIMENTAL RESULTS

### 7.1 Motion Models Used in the Experiments

For testing purposes, we have used two different function spaces (function bases) to model motion fields.

#### 7.1.1 Polynomial Spaces

The first function space we tested is a polynomial space, which contains as special cases the constant, affine, and quadratic motion models. The basis function of a polynomial function space of degree  $P$  in 2D are described by:

$$\theta_j(\mathbf{x}) = x^{i-l} y^l, \quad (55)$$

TABLE 1  
Basis Functions for the Affine Model

$j$	0	1	2
$i$	0	1	1
$l$	0	0	1
$\theta_j$	$x^0 y^0$	$x^1 y^0$	$x^0 y^1$

where:

$$i = \left\lfloor \frac{\sqrt{8j+1} - 1}{2} \right\rfloor, \quad (56)$$

$$l = j - \frac{i(i+1)}{2}, \quad (57)$$

and  $j \in [0, (P+1)(P+2)/2]$ .

For the affine model, for instance, the degree is  $P = 1$ , and we have three basis functions given by:

$$\theta_0(\mathbf{x}) = 1, \quad (58)$$

$$\theta_1(\mathbf{x}) = x, \quad (59)$$

$$\theta_2(\mathbf{x}) = y, \quad (60)$$

and the values of  $j$ ,  $i$ , and  $l$  (of (55)) are as in Table 1. With digital images,  $x$  and  $y$  are discrete and  $x \in [0, W-1]$ ,  $y \in [0, H-1]$ , where  $W$  and  $H$  are, respectively, the width and height of the image in pixels.

### 7.1.2 Cosine Transforms

This second set of basis functions uses cosine transforms. In the discrete case of digital images, the basis functions are of the form (this representation is referred to as the discrete cosine transform, or DCT, in signal processing):

$$\theta_j(\mathbf{x}) = \rho_1 \rho_2 \cos\left(\frac{\pi(2x+1)q}{2W}\right) \cos\left(\frac{\pi(2y+1)p}{2H}\right), \quad (61)$$

where  $p = \lfloor j/P \rfloor$ ,  $q = j - pP$ ,  $j \in [0, P^2 - 1]$ , and

$$\rho_1 = \begin{cases} \frac{1}{\sqrt{P}} & p = 0 \\ \sqrt{\frac{2}{P}} & p \in [1, P-1], \end{cases} \quad (62)$$

$$\rho_2 = \begin{cases} \frac{1}{\sqrt{P}} & q = 0 \\ \sqrt{\frac{2}{P}} & q \in [1, P-1]. \end{cases} \quad (63)$$

We experimented with four different bases, using  $P = 1, 2, 3$ , and  $4$ . These bases have, respectively,  $M = 1, 4, 9$ , and  $16$  basis functions.

## 7.2 Results with Synthetic Motion

The purpose of this first example is to show that joint parametric motion segmentation and estimation can produce accurate motion. We use the synthetic sequence *Marmor*, for which ground truth motion and segmentation are available. This sequence consists of a 3D scene showing two textured blocks moving with different motions in an otherwise static environment. The experiments use frames 135 and 136 of the sequence. Application of the proposed method for this example uses discrete cosine transform bases, and the

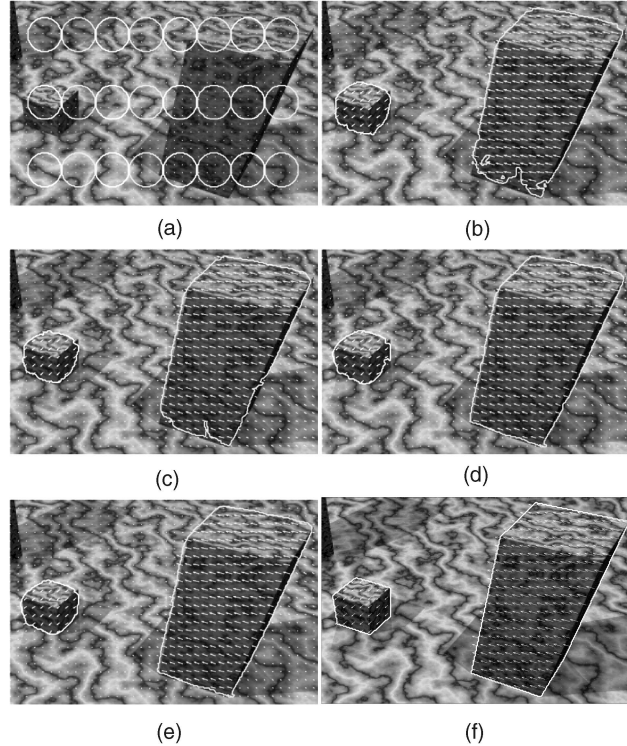


Fig. 1. Motion segmentation and parametric motion estimation for the sequence *Marmor* using the discrete cosines function base. The complexity of the motion was gradually increased from 1 to 16 basis functions. (a) Initialization. (b) Partial ( $M = 1$ ). (c) Partial ( $M = 4$ ). (d) Partial ( $M = 9$ ). (e) Final ( $M = 16$ ). (f) True motion.

strategy of gradual increase of the motion model complexity described in Section 6.

### 7.2.1 Evolution of Model Complexity

Fig. 1 shows the segmentation and corresponding estimated motion obtained by using different motion models. Fig. 1a shows the image with the initialization superimposed. Images in Figs. 1b, 1c, and 1d show partial results of applying motion models with increasing complexity. The first model applied is a piecewise constant model ( $M = 1$ ). The final result obtained by using this model (Fig. 1b) is used as input to the next stage, which uses a more complex model ( $M = 4$ ). The process is iterated by increasing the complexity of the motion model ( $M = 9$  and  $M = 16$ ). Fig. 1e displays the segmentation and motion estimation at the end of the process. The ground truth motion is shown in Fig. 1f.

### 7.2.2 Motion Estimation Evaluation

Table 2 shows the results of evaluation of the performance of the algorithm with respect to motion estimation. This table contains the mean and standard deviation of the error between the ground truth motion and the motion estimated by the proposed method and two well-known variational methods, the Horn-Schunk [24] method and the boundary-preserving method of Deriche et al. [4]. The proposed method was tested using DCT bases with  $P = 1, 2, 3, 4$ . Also shown in the table is the norm of the displaced frame difference (DFD), which is the error of reconstructing  $I(t+1)$  from  $I(t)$  using the estimated motion. The proposed method outperforms the reference methods and the results

TABLE 2  
Performance Comparison of the Objects Motion Estimation

Method	Angle ( $^{\circ}$ )		Magnitude		DFD (norm)
	mean	std	mean	std	
Horn-Schunk	5.1137	4.6126	0.4044	0.2214	0.0216
Deriche <i>et al.</i>	4.1988	3.7774	0.4795	0.2512	0.0211
Proposed (M=1)	2.5108	3.3217	0.1956	0.1790	0.0209
Proposed (M=4)	0.8981	1.8695	0.1294	0.1375	0.0212
Proposed (M=9)	0.7473	1.0626	0.0982	0.1048	0.0186
Proposed (M=16)	0.8838	1.2075	0.0689	0.0907	0.0176

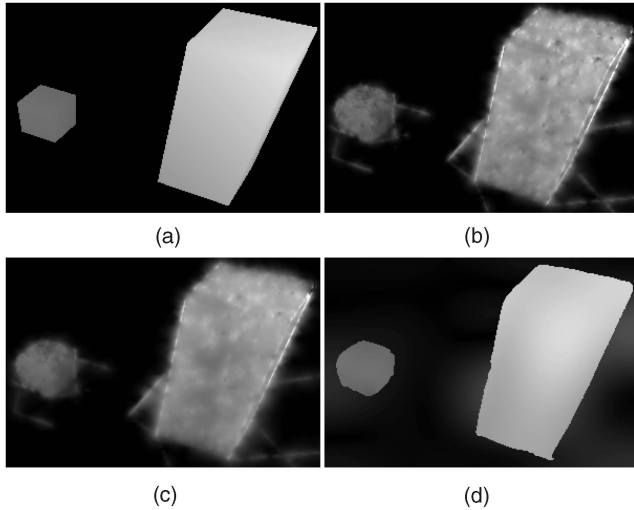


Fig. 2. Motion vector magnitude for the frame 135 of sequence *Marmor*. (a) Ground truth. (b) Horn-Schunk. (c) Deriche *et al.* (d) Proposed.

are comparable to others in the literature, in [5], for instance.

Fig. 2 shows the magnitude of the motion vectors for the reference methods, the ground truth, and the proposed method with a basis of 16 functions. The results from the nonparametric motion models show important variations inside the objects regions. The better results of the proposed method can be attributed to the fact that it segments the image and uses a sufficiently representative set of parameters within each region of segmentation.

### 7.2.3 Segmentation Evaluation

To evaluate the segmentation we used the ground truth regions to determine the percentage of points incorrectly classified. The results are shown in Table 3.

The segmentation errors are shown in Fig. 3. We note that the segmentation error decreases with the increase in motion model complexity. Therefore, more complex models have produced both a better segmentation and better motion estimates.

## 7.3 Results with Real Image Sequences

The purpose of these experiments is to show examples where parametric models more complex than the generally used constant and affine models are needed. In these experiments, DCT models are compared to the affine model.

The first experiment uses a sequence representing a curled snake in movement. The raised head of the snake is immobile, and the rest of the body undergoes contortions. Fig. 4 shows

TABLE 3  
Performance Evaluation of the Segmentation Results for Sequence *Marmor*

Method	Segmentation error (%)
Proposed (M=1)	1.90
Proposed (M=2)	1.85
Proposed (M=9)	1.27
Proposed (M=16)	1.16

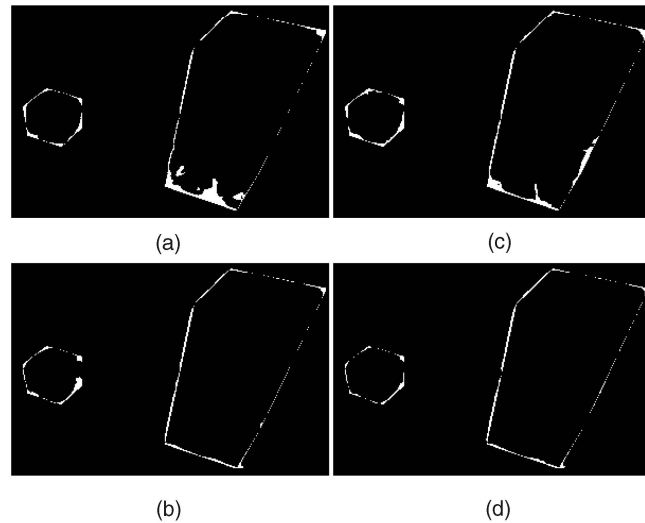


Fig. 3. Segmentation error for frame 135 of sequence *Marmor*. (a)  $M = 1$ . (b)  $M = 4$ . (c)  $M = 9$ . (d)  $M = 16$ .

the first of the two images used. The motion is in one direction in the upper part of the body, in the opposite direction in the middle part, and almost stationary in the lower part.

Fig. 4 shows the results for the constant, the affine, and DCT models. As can be seen in the figures, the constant and affine motion segmentations are quite poor. However, by allowing more complex variations of motion, the more complex DCT models yield good segmentations, and the estimates of motion have direction which is consistent with the observed contortion movement of the snake.

The second example uses images of a cheetah walking. The body of the animal has a forward, nonrigid movement. One leg of the animal is moving forward faster than the body. Another leg is moving with the body. The other two legs are at rest. The animal is moving left and forward. The camera is also in movement to follow the animal, inducing image motion of the background to the right and causing small motion for the animal.

Results are shown in Fig. 5. As in the preceding example, the constant and affine models fail. The constant model fails to capture the motion of the body, assigning a constant motion in the upper-left direction which is not consistent with the real motion. The affine motion is not able to segment the scene properly, giving a fragmented segmentation where the region assigned to the animal includes parts of the background and vice versa. In contrast, the more complex models yield a good segmentation and motion that is consistent with the motion of the animal.

Note that the piecewise constant model produces a reasonable segmentation but a poor estimate of motion. In contrast, the affine model, more complex, gives a good



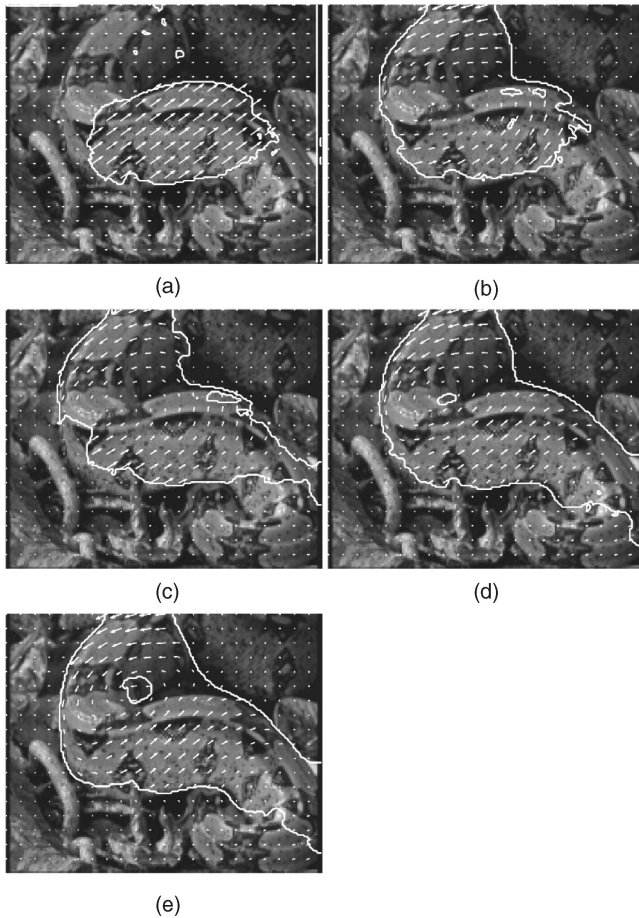


Fig. 4. Motion estimation and segmentation with different methods for the sequence "Snake." (a) Constant ( $M = 1$ ). (b) Affine ( $M = 3$ ). (c) Proposed ( $M = 4$ ). (d) Proposed ( $M = 9$ ). (e) Proposed ( $M = 16$ ).

estimate of motion but a poor segmentation. The affine model has simply fragmented the segmentation to better represent motion. The proposed method starts with a simple model to obtain a first approximation of the segmentation, and then refines the estimated motion and segmentation using more complex models.

#### 7.4 Motion Estimation in Nontextured Areas

The absence of texture on a scene surfaces makes the estimation of motion almost impossible and most algorithms fail to assign a correct motion to nontextured areas. Parametric motion estimation has the advantage of assigning a motion to a region based on the dominant motion available, propagating, therefore, motion estimation from textured to nontextured regions. Because segmentation is performed jointly with motion estimation within each region of segmentation via model parameters, nontextured areas of a region inherit the parameters estimated from the region textured areas. Algorithms which use smoothness regularization but no explicit, model-based segmentation of motion will propagate motion estimates from textured to untextured parts of the image by diffusion, leading to slow, inaccurate estimation. The next example is intended to demonstrate this advantage of parametric motion estimation. We use a real video sequence of a highway scene. In the images shown in Fig. 6, the truck moves towards the camera. There are several areas without texture where standard motion estimation

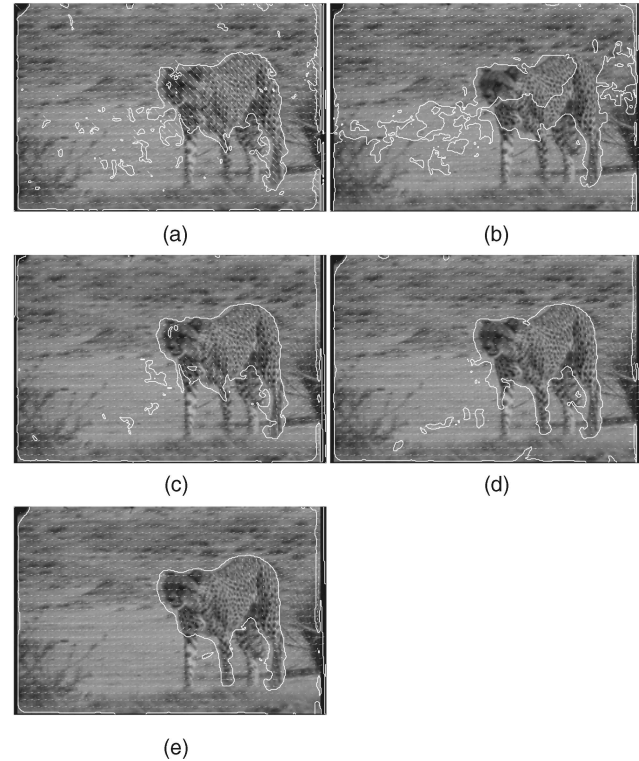


Fig. 5. Motion estimation and segmentation with different methods for the sequence "Cheetah." (a) Constant ( $M = 1$ ). (b) Affine ( $M = 3$ ). (c) Proposed ( $M = 4$ ). (d) Proposed ( $M = 9$ ). (e) Proposed ( $M = 16$ ).

algorithms fail. We present the results of motion estimation using the methods of Horn-and-Schuck and Deriche et al. and the proposed method using the affine model. The results show the accuracy of the segmentation, which includes the truck and its shadow.

In order to show the advantages of the flexibility in the motion model, we also estimated the motion for the truck with a quadratic motion model (polynomial of degree 2) by initializing the algorithm with the result obtained from the affine motion model previously shown.

The estimated motion is used to reconstruct a motion compensated version of the first frame from the second. The reconstruction error images ( $\times 5$ ) obtained by motion compensation are shown in Fig. 7. The PSNRs of the reconstructed frames are also shown.

#### 7.5 Image Sequences with Occlusions

This last experiment shows a limitation of the current method, namely, concerning handling of occlusions. Using real images and synthetic motion, we constructed a sequence containing two differently moving objects against a moving background. The image consists of a segment of the stereoscopic sequence "Aqua" as background with the images of two fishes from the same sequences superimposed on the background. The background is moving with a translational motion of  $t = (-3, -5)^T$ . The fishes are moving with affine motion. The motion is relatively of large extent, over 15 pixels at some points. These large motions produce large occluded/newly exposed regions. In the current implementation of the algorithm, these regions are assigned to one of the moving regions, producing an extended region which contains not only a moving object

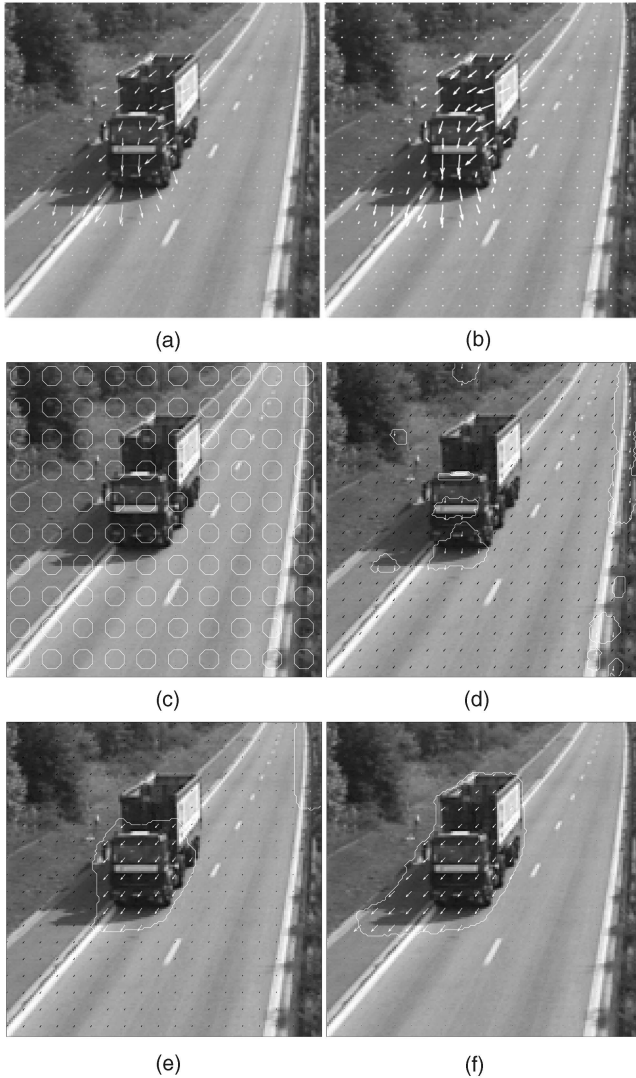


Fig. 6. Motion estimation for the sequence road (affine motion model). (a) Horn-Schunk. (b) Deriche et al. (c) Initialization. (d) Evolution. (e) Evolution. (f) Final ( $M = 3$ ).

but also the occluded/newly exposed regions. In Fig. 8, this effect is clearly observable, with occlusions being associated to the fish on the right of the image. We are currently working on an extension of the proposed algorithm to take into account occluded/newly exposed regions.

## 8 CONCLUSION AND FUTURE WORK

The purpose of this study was to investigate a variational method for joint segmentation and parametric estimation of image motion by basis function representation and level set evolution. The functional in the method contained three terms, one to bias the solution toward a segmentation with smooth boundaries, a second to bias the solution toward a segmentation with boundaries that coincide with motion discontinuities, and a third to measure the conformity of image motion in each region of segmentation to an expansion in a general linear space of functions. The resulting algorithm consisted of concurrent curve evolution and least-squares estimation of the motion parameters within each region of segmentation. The algorithm and its implementation have

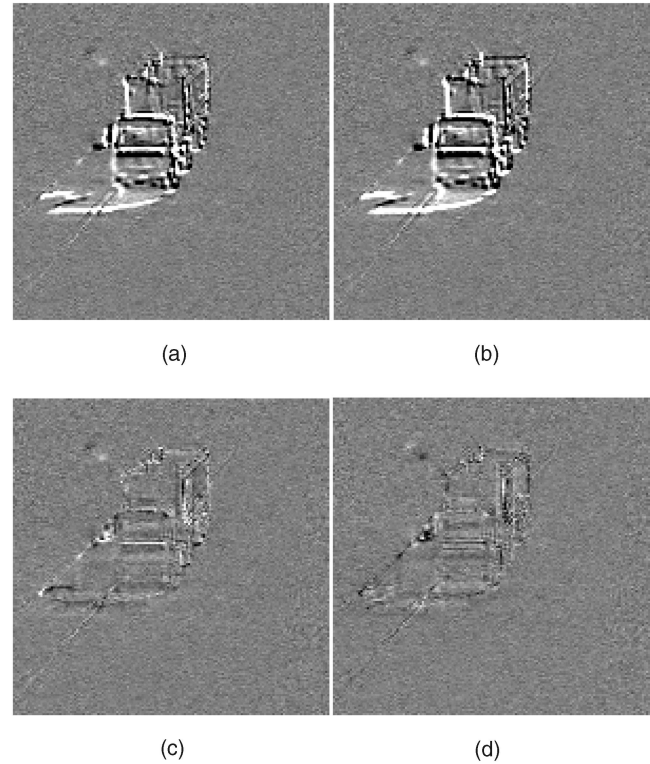


Fig. 7. Motion compensated reconstruction of first frame of road with the estimated motion. (a) Horn-Schunk (PSNR = 27.38dB). (b) Deriche et al. (PSNR = 26.35dB). (c) Affine (PSNR = 37.04dB). (d) Quadratic (PSNR = 37.59dB).

been verified on synthetic and real image sequences using a cosine expansion of image motion.

Further investigation can provide useful extensions of the method we presented. For instance, one can address the problem of relaxing the assumption that the number of regions of segmentation, although arbitrary, is known beforehand.

One can also address the important problem of overfitting mentioned in [17], and investigate methods to determine automatically (without user intervention) the proper parametric model complexity to use. A possible answer is the use of a wavelet representation of motion. The use of wavelets would be justified for two reasons. First, both theory and practice have shown that wavelet representation of images is quite effective in various image processing problems [30] such as coding, denoising, restoration, and reconstruction, among others. Second, the notion of multiresolution is intrinsic to wavelets. Therefore, multiresolution motion estimation and segmentation via wavelet representation at increasing resolutions would be an automatic, sound alternative to a user assisted incremental instantiation of parametric models of increasing complexity [31]. However, important antecedent issues to resolve are the evaluation of wavelet functions to compute the motion parameters. The use of a fast transform technique would be a desirable feature in this case, but the irregularity of the regions makes this a difficult problem to solve.

Finally, this method can be used for other types of images. We are currently following through a study with gray-scale images [32] and considering an application to segmentation and classification of satellite images, and texture via filters response statistics [33], [34].

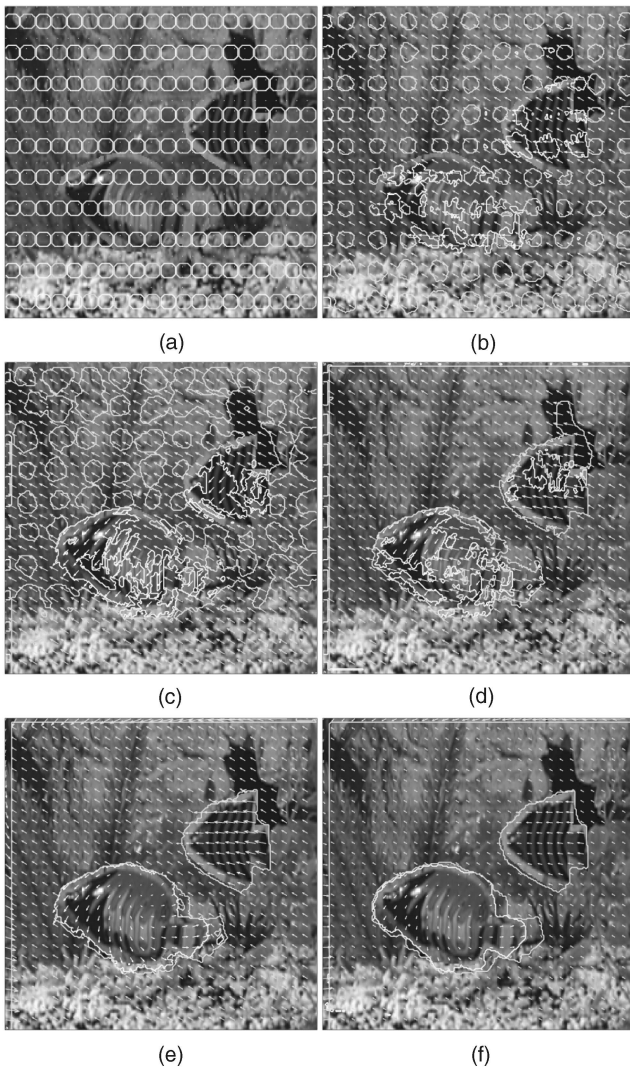
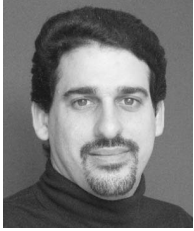


Fig. 8. Motion segmentation and parametric motion estimation for the sequence Aqua. (a) Initialization. (b) Evolution. (c) Evolution. (d) Evolution. (e) Evolution. (f) Final.

## REFERENCES

- [1] A. Kaup, "Object-Based Texture Coding of Moving Video in MPEG-4," *IEEE Trans. Circuits Systems Video Technology*, vol. 9, no. 1, pp. 5-15, Feb. 1999.
- [2] A. Mitiche, *Computational Analysis of Visual Motion*. Kluwer Academic, 1994.
- [3] T. Wiegand, G. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H. 264/AVC Video Coding Standard," *IEEE Trans. Circuits Systems Video Technology*, vol. 13, no. 7, pp. 560-576, July 2003.
- [4] R. Deriche, P. Kornprobst, and G. Aubert, "Optical-Flow Estimation while Preserving Its Discontinuities: A Variational Approach," *Proc. Second Asian Conf. Computer Vision*, vol. 2, pp. 290-295, Dec. 1995.
- [5] T. Brox, A. Bruhn, N. Papenberger, and J. Weickert, "High Accuracy Optical Flow Estimation Based on a Theory of Warping," *Proc. European Conf. Computer Vision*, vol. 4, pp. 25-36, May 2004.
- [6] C. Bergeron and E. Dubois, "Gradient Based Algorithms for Block Oriented MAP Estimation of Motion and Application to Motion-Compensated Temporal Interpolation," *IEEE Trans. Circuits Systems Video Technology*, vol. 1, pp. 72-85, Mar. 1991.
- [7] J. Odobez and P. Bouthemy, "Robust Multiresolution Estimation of Parametric Motion Models," *J. Visual Comm. and Image Representation*, vol. 6, no. 4, pp. 348-365, Dec. 1995.
- [8] Y. Altunbasak, R. Mersereau, and A. Patti, "A Fast Parametric Motion Estimation Algorithm with Illumination and Lens Distortion Correction," *IEEE Trans. Image Processing*, vol. 12, no. 4, pp. 395-408, Apr. 2003.
- [9] N. Diehl, "Object-Oriented Motion Estimation and Segmentation in Image Sequences," *Signal Processing: Image Comm.*, vol. 3, no. 1, pp. 23-56, Feb. 1991.
- [10] F. Dufaux, I. Moccagatta, F. Moscheni, and H. Nicolas, "Vector Quantization-Based Motion Field Segmentation under the Entropy Criterion," *J. Visual Comm. and Image Representation*, vol. 5, pp. 356-369, Dec. 1994.
- [11] J. Wang and E. Adelson, "Representing Moving Images with Layers," *IEEE Trans. Image Processing*, vol. 3, no. 5, pp. 625-638, Sept. 1994.
- [12] G. Borshukov, G. Bozdagi, Y. Altunbasak, and A. Tekalp, "Motion Segmentation by Multistage Affine Classification," *IEEE Trans. Image Processing*, vol. 6, no. 11, pp. 1591-1594, 1997.
- [13] M. Chang, A. Tekalp, and M. Sezan, "Simultaneous Motion Estimation and Segmentation," *IEEE Trans. Image Processing*, vol. 6, no. 9, pp. 1326-1333, Sept. 1997.
- [14] A.-R. Mansouri and J. Konrad, "Multiple Motion Segmentation with Level Sets," *IEEE Trans. Image Processing*, vol. 12, no. 2, pp. 201-220, Feb. 2003.
- [15] A. Mansouri, A. Mitiche, and C. Langevin, "Joint Space-Time Motion-Based Segmentation of Image Sequences with Level Set PDEs," *Proc. IEEE Workshop Motion and Video Computing*, pp. 50-55, Dec. 2002.
- [16] A.-R. Mansouri, A. Mitiche, and F. Dolla, "Motion-Based Figure-Ground Segmentation by Maximum Motion Separation," *Proc. IEEE Int'l Conf. Image Processing*, Sept. 2003.
- [17] D. Cremers and S. Soatto, "Variational Space-Time Motion Segmentation," *Proc. IEEE Int'l Conf. Computer Vision*, pp. 886-892, Oct. 2003.
- [18] D. Mumford and J. Shah, "Optimal Approximations by Piecewise Smooth Functions and Associated Variational-Problems," *Comm. Pure Applied Math.*, vol. 42, no. 5, pp. 577-685, July 1989.
- [19] G. Aubert, M. Barlaud, O. Faugeras, and S. Jehan-Besson, "Image Segmentation Using Active Contours: Calculus of Variations or Shape Gradients?" *SIAM J. Applied Math.*, vol. 63, no. 6, pp. 2128-2154, 2003.
- [20] D. Cremers, "A Multiphase Level Set Framework for Motion Segmentation," *Proc. Int'l Conf. Scale-Space Theories in Computer Vision*, pp. 599-614, June 2003.
- [21] S. Zhu and A. Yuille, "Region Competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multiband Image Segmentation," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 18, no. 9, pp. 884-900, Sept. 1996.
- [22] C. Stiller and J. Konrad, "Estimating Motion in Image Sequences: A Tutorial on Modeling and Computation of 2D Motion," *IEEE Signal Processing Magazine*, vol. 16, no. 4, pp. 70-91, July 1999.
- [23] A.-R. Mansouri, A. Mitiche, and C. Vázquez, "Image Portioning by Level Set Multiregion Competition," *Proc. IEEE Int'l Conf. Image Processing*, Oct. 2004.
- [24] B. Horn and B. Schunck, "Determining the Optical Flow," *Artificial Intelligence*, vol. 17, pp. 185-203, 1981.
- [25] A. Mitiche, R. Feghali, and A. Mansouri, "Motion Tracking as Spatio-Temporal Motion Boundary Detection," *J. Robotics and Autonomous Systems*, vol. 43, pp. 39-50, 2003.
- [26] T. Chan and L. Vese, "Active Contours without Edges," *IEEE Trans. Image Processing*, vol. 10, no. 2, pp. 266-277, Feb. 2001.
- [27] G. Aubert and P. Kornprobst, *Mathematical Problems in Image Processing*. Springer Verlag, 2002.
- [28] J. Sethian, *Level Set Methods and Fast Marching Methods*, second ed. Cambridge Univ. Press, 1999.
- [29] C. Vázquez, A. Mitiche, and I. BenAyed, "Segmentation of Vectorial Images by a Global Curve Evolution Method," *Proc. Reconnaissance des Formes et Intelligence Artificielle, RFLA-04*, Jan. 2004.
- [30] S. Mallat, *A Wavelet Tour of Signal Processing*, first ed. Academic Press, 1998.
- [31] J. Odobez and P. Bouthemy, "Direct Incremental Model-Based Image Motion Segmentation for Video Analysis," *Signal Processing*, vol. 6, no. 2, pp. 143-155, 1998.
- [32] C. Vázquez, A.-R. Mansouri, and A. Mitiche, "Approximation of Images by Basis Functions for Multiple Region Segmentation with Level Sets," *Proc. IEEE Int'l Conf. Image Processing*, pp. 549-552, Oct. 2004.

- [33] B. Sandberg, T. Chan, and L. Vese, "A Level-Set and Gabor-Based Active Contour Algorithm for Segmenting Textured Images," Technical Report CAM Report 02-39, Univ. of California, Los Angeles, July 2002.
- [34] S. Zhu and D. Mumford, "Minimax Entropy Principle and Its Application to Texture Modeling," *Neural Computation*, vol. 9, no. 8, 1997.



**Carlos Vázquez** received the BEng degree in electrical engineering with great distinction and the MSc degree in applied computer sciences from the Higher Polytechnic Institute "José Antonio Echeverría" (ISPJAE) in Havana, Cuba, in 1992 and 1997, respectively. He received the PhD degree in telecommunications from INRS-EMT, Montréal, Canada, in 2003. From September 1992 to August 1997, he was a staff member of the Telecommunications Department

of the Electrical Engineering Faculty at ISPJAE, first as a lecturer, becoming assistant professor in 1996. In September 1997, he joined INRS as a PhD student and recipient of an excellence scholarship awarded by the ministry of education of Quebec. From 2002 to 2004, he was a postdoctoral fellow at INRS-EMT, Montréal. He worked as research associate in the ECE Department of Concordia University, Montréal, Canada, in 2004-2005. He is currently a research scientist at the Advanced Video System Group of the Communication Research Centre (CRC) in Ottawa, Canada. His research interests are in the areas of stereo and multiview vision, image/video sampling and interpolation, splines and wavelets, image and video coding, motion/disparity estimation, and compensation. He has been a member of the IEEE since 2003.



**Amar Mitiche** received the Licence Ès Sciences in mathematics from the University of Algiers and the PhD degree in computer science from the University of Texas at Austin. He is currently a professor at the Institut National de Recherche Scientifique (INRS), Department of Telecommunications, in Montréal, Québec, Canada. His research is in computer vision. His current interests include image segmentation, motion analysis in monocular and stereoscopic image sequences (detection, estimation, segmentation, tracking, and 3D interpretation) with a focus on level set methods, and written text recognition with a focus on neural networks methods. He has been a member of the IEEE since 2003.



**Robert Laganière** received the PhD degree from INRS-Telecommunications in Montreal in 1996. He is an associate professor in the School of Information Technology and Engineering at the University of Ottawa. His research interests are in computer vision and image processing with applications to augmented reality, visual surveillance, 3D reconstruction, and image-based rendering. He is also coauthor of a book on object-oriented software development, published by McGraw Hill. He has been a member of the IEEE since 2000.

► **For more information on this or any other computing topic, please visit our Digital Library at [www.computer.org/publications/dlib](http://www.computer.org/publications/dlib).**