A variational approach for object contour tracking

Nicolas Papadakis, Etienne Mémin and Frédéric Cao

IRISA/INRIA
Campus de Beaulieu,
35 042 Rennes Cedex, France
{npapadak,memin,fcao}@irisa.fr

Abstract. In this paper we describe a new framework for the tracking of closed curves described through implicit surface modeling. The approach proposed here enables a continuous tracking along an image sequence of deformable object contours. Such an approach is formalized through the minimization of a global spatio-temporal continuous cost functional stemming from a Bayesian Maximum a posteriori estimation of a Gaussian probability distribution. The resulting minimization sequence consists in a forward integration of an evolution law followed by a backward integration of an adjoint evolution model. This latter pde include also a term related to the discrepancy between the curve evolution law and a noisy observation of the curve. The efficiency of the approach is demonstrated on image sequences showing deformable objects of different natures.

1 Introduction

Tracking curves and contours is an important and difficult problem in computer vision. As a matter of fact, the shapes of deformable or rigid objects may vary a lot along an image sequence. These changes are due to perspective effects, self occlusions or to complex deformations of the object itself. The intrinsic continuous nature of these features and their high dimensionality makes difficult the conception of efficient non linear Bayesian filters as sampling in large scale dimension is usually completely inefficient. Besides, the use of lower dimensional features such as explicit parametric curves is limited to the visual tracking of objects with well defined shapes and that do not exhibit any change of topology [2,15]. This kind of representation is for instance very difficult to settle when focusing on the tracking of temperature level curves in satellite atmospheric images, or simply when the aim is to track an unknown deformable object with no predefined shape.

In such a context, approaches based on level set representation have been proposed [5,6,8,11,12,14,16]. Nevertheless, apart from [11], all these solutions aim more at estimating successive instantaneous segmentation maps than at really tracking objects. Indeed, in a formal point of view, they cannot be really considered as tracking approaches for several reasons. First of all, these methods are very sensitive to noise [9]. Unless the introduction of some statistical knowledges on the shape [3,7,13] these approaches do not allow to handle partial occlusions of the target. Since these methods do not include any temporal evolution law on the tracked object shape, they are not able to cope with severe failures of the imaging sensor (for instance a complete loss of image

data, a severe motion blur or high saturation caused by over exposure). And finally, no error assessment on the estimation is in addition possible. For all these reasons, only approaches introducing basically a competition between a dynamical evolution model and a measurement process of the target of interest enable to handle a robust tracking in natural way. On the same basis, we propose here a variational method allowing to combine these two ingredients for the tracking of non parametric curves.

Unlike the technique proposed in [11], which explicitly also introduces a dynamic law in the curve evolution, our work includes in the same spirit as a Kalman smoothing a temporal smoothing along the whole image sequence. In the same way as a Kalman smoother the technique we propose here allows to estimate the conditional expectation of the state process given all the available measurements extracted in the whole image sequence. Nevertheless, unlike stochastic techniques our approach allows to handle features of very high dimension.

The variational tracking technique we introduce relies on data assimilation concepts used for instance in meteorology [1,4,17]. As will be demonstrated in the experimental section, such a technique enables to handle naturally partial occlusions and a complete loss of image data on long time period without resorting to complex mechanisms.

2 A system for contour tracking

As we wish to focus in this work on the tracking of non parametric closed curves that may exhibit topology changes during the time of the analyzed image sequence, we will rely on an implicit level set representation of the curve of interest $\Gamma(t)$ at time $t \in [t_0, \tau]$ of the image sequence [12,16]. Within that framework, the curve $\Gamma(t)$ enclosing the target to track is implicitly described as the zero level set of a function $\phi(\boldsymbol{x}, t) : \Omega \times \mathbb{R}+ \rightarrow \mathbb{R}$:

$$\Gamma(t) = \{\boldsymbol{x} \in \Omega \mid \phi(\boldsymbol{x}, t) = 0\},$$

where Ω stands for the image spatial domain. This representation enables an Eulerian representation of the evolving contours. As such, it allows to avoid the inescapable regriding ad-hoc processes of the different control points associated to any explicit spline based Lagrangian description of the evolving curve. The problem we want to face thus consists in estimating for a whole time range the state of an unknown curve, and hence of its associated implicit surface ϕ. To that end, we first define an *a priori* evolution law of the unknown surface. We will assume that the curve is transported at each frame instant by a velocity fields, $\boldsymbol{w}(t)$, and diffuses according to a mean curvature motion. In term of the implicit surface this evolution model reads:

$$\frac{d\phi}{dt} = \frac{\partial \phi}{\partial t} + \boldsymbol{\nabla}\phi(\boldsymbol{x}, t)^T \boldsymbol{w}(t) = \epsilon\kappa\|\boldsymbol{\nabla\phi}\|, \tag{1}$$

where κ denotes the curve curvature. Introducing the surface normal, equation (1) can be written as:

$$\frac{\partial \phi}{\partial t} = -\left(\boldsymbol{w} \cdot \boldsymbol{n} - \epsilon\kappa\right) \|\boldsymbol{\nabla\phi}\|, \tag{2}$$

where the normal and the curvature are given directly in term of surface gradient,

$$\text{with} \quad \kappa = div\left(\frac{\boldsymbol{\nabla}\phi}{\|\boldsymbol{\nabla}\phi\|}\right) \quad \text{and } \boldsymbol{n} = \frac{\boldsymbol{\nabla}\phi}{\|\boldsymbol{\nabla}\phi\|}.$$

At the initial time, the implicit function is assigned to a signed distance function up to a white Gaussian noise. More precisely, the value of $\phi(\boldsymbol{x}, t_0)$ is set to the distance $g(\boldsymbol{x}, \Gamma(t_0))$ of the closest point of a given initial curve $\Gamma(t_0)$, with the convention that $g(\boldsymbol{x}, t_0)$ is negative inside the contour, and positive outside. An additive white noise process is added in order to model the uncertainty we have on the initial curve. Associated to this evolution model and to the initialization process we previously described, we will assume that an observation function $Y(t)$ which constitutes a noisy measurement of the target is available. This function will be assumed to be related to the unknown state function ϕ through a differential operator $\mathbb{H}$ and a white Gaussian noise $\varepsilon(\boldsymbol{x}, t)$:

$$Y(\boldsymbol{x}, t) = \mathbb{H}(\phi(\boldsymbol{x}, t)) + \varepsilon(\boldsymbol{x}, t). \tag{3}$$

Let us note that in our case, the continuous observation function, $Y(t)$, is obtained from discrete image frames, I_n, through multiplication by a family of localization functions. These functions can be defined from delta functions at the observed time and location, or from more advanced spatio-temporal averaging functions. Gathering all the elements we have described so far, we get the following system for our tracking problem:

$$\begin{cases} \frac{\partial \phi}{\partial t} + \mathbb{M}(\phi) = \eta(\boldsymbol{x}, t) \\ \phi(\boldsymbol{x}, t_0) = g(\boldsymbol{x}, \Gamma(t_0)) + \nu(\boldsymbol{x}, t) \\ Y(\boldsymbol{x}, t) = \mathbb{H}(\phi) + \varepsilon(\boldsymbol{x}, t) \end{cases} \tag{4}$$

In this system, $\mathbb{M}$, denotes the differential operator involved in equation (2) and η, ν and ε are time varying zero mean Gaussian noise functions defined on the whole image plane, with covariance functions $Q(\boldsymbol{x}, t, \boldsymbol{x}', t')$, $B(\boldsymbol{x}, \boldsymbol{x}')$, $R(\boldsymbol{x}, t, \boldsymbol{x}'t')$ respectively. The different noise functions represent the different errors involved in the different components of our system.

3 Variational tracking formulation

3.1 Penalty function

Considering a system such as the one we previously settled comes to fix the conditional probability distribution $p(\phi(t)|g)$, $p(g|\phi(t_0))$ and $p(Y(t)|\phi(t))$. From these pdf's, one get the *a posteriori* density function up to a normalization constant. As all the error distributions involved here are Gaussian, the *a posteriori* pdf is also Gaussian. The maximization of this distribution is thus equivalent to the minimization of the following quadratic penalty function:

$$\begin{aligned} J(\phi) = & \frac{1}{2}\int_{\Omega,t}\int_{\Omega,t}\left(\frac{\partial\phi}{\partial t} + \mathbb{M}(\phi)\right)^T(\boldsymbol{x}, t)Q^{-1}(\boldsymbol{x}, t, \boldsymbol{x}', t')\left(\frac{\partial\phi}{\partial t} + \mathbb{M}(\phi)\right)(\boldsymbol{x}', t')dt'd\boldsymbol{x}'dtd\boldsymbol{x} \\ & + \frac{1}{2}\int_{\Omega}\int_{\Omega}\left(\phi(\boldsymbol{x}, t_0) - g(\boldsymbol{x}, \Gamma(t_0))\right)^T B^{-1}(\boldsymbol{x}, \boldsymbol{x}')\left(\phi(\boldsymbol{x}', t_0) - g(\boldsymbol{x}', \Gamma(t_0))\right)d\boldsymbol{x}'d\boldsymbol{x} \\ & + \frac{1}{2}\int_{\Omega,t}\int_{\Omega,t}\left(Y - \mathbb{H}(\phi)\right)^T(\boldsymbol{x}, t)R^{-1}(\boldsymbol{x}, t, \boldsymbol{x}', t')\left(Y - \mathbb{H}(\phi)\right)(\boldsymbol{x}', t')dt'd\boldsymbol{x}'dtd\boldsymbol{x}. \end{aligned} \tag{5}$$

In this functional, $\boldsymbol{x}$, denotes the spatial domain coordinates defined on the image domain Ω and t is a continuous time index lying within the image sequence time interval $[t_0 = 0; \tau]$. The minimizer of this expression minimizes a sum of norms expressing all the possible correlations between errors at arbitrary two points of the image sequence. The double space and time integrations are here due to the fact that the covariance functions are first assumed to be non null for any two points $(\boldsymbol{x}, t)$ and $(\boldsymbol{x}', t')$ (correlated case) in order to stick to the most general case before considering a simplified uncorrelated case corresponding to diagonal correlation matrices in a discrete setting.

In order to devise a minimizing sequence for this functional, let us now derive the associated Euler-Lagrange equations.

3.2 Euler-Lagrange equations

Function ϕ is a minimum of functional J, if it is also a minimum of a cost function $J(\phi + \beta\theta(\boldsymbol{x}, t))$, where $\theta(\boldsymbol{x}, t))$ belongs to a space of admissible functions and β is a positive parameter. In other words, ϕ must cancel the directional derivative :

$$\delta J_\phi(\theta) = \lim_{\beta \to 0} \frac{dJ(\phi + \beta\theta(\boldsymbol{x}, t))}{d\beta} = 0.$$

The cost function $J(\phi + \beta\theta(\boldsymbol{x}, t))$ reads

$$\begin{aligned} J = & \frac{1}{2} \int_\Omega \int_0^\tau \left[\left(\frac{\partial \phi}{\partial t} + \beta \frac{\partial \theta}{\partial t} + \mathbb{M}(\phi + \beta\theta) \right)^T \int_\Omega \int_0^\tau Q^{-1} \left(\frac{\partial \phi}{\partial t} + \beta \frac{\partial \theta}{\partial t} + \mathbb{M}(\phi + \beta\theta) \right) dt' d\boldsymbol{x}' \right] dt d\boldsymbol{x} \\ & + \frac{1}{2} \int_\Omega \int_\Omega (\phi + \beta\theta - g)^T B^{-1} (\phi + \beta\theta - g) d\boldsymbol{x}' d\boldsymbol{x} \\ & + \frac{1}{2} \int_\Omega \int_0^\tau \int_\Omega \int_0^\tau (Y - \mathbb{H}(\phi + \beta\theta))^T R^{-1} (Y - \mathbb{H}(\phi + \beta\theta)) dt' d\boldsymbol{x}' dt d\boldsymbol{x}. \end{aligned} \tag{6}$$

Adjoint variable In order to perform an integration by part – to factorize this expression by θ– we introduce an "adjoint variable" λ defined by:

$$\lambda(\boldsymbol{x}, t) = \int_\Omega \int_0^\tau Q^{-1} \left(\frac{\partial \phi}{\partial t} + \mathbb{M}(\phi) \right) dt' d\boldsymbol{x}', \tag{7}$$

as well as *linear tangent operators* $(\frac{\partial \mathbb{M}}{\partial \phi})$ and $(\frac{\partial \mathbb{H}}{\partial \phi})$ defined by

$$\lim_{\beta \to 0} \frac{d\mathbb{M}(\phi + \beta\theta)}{d\beta} = \frac{\partial \mathbb{M}}{\partial \phi}(\theta). \tag{8}$$

By taking the limit $\beta \to 0$, the derivative of expression (6) then reads

$$\begin{aligned} \lim_{\beta \to 0} \frac{dJ}{d\beta} = & \int_\Omega \int_0^\tau \left(\frac{\partial \theta}{\partial t} + \frac{\partial \mathbb{M}}{\partial \phi} \theta \right)^T \lambda(\boldsymbol{x}, t) dt d\boldsymbol{x} \\ & + \int_\Omega \int_\Omega \theta^T(\boldsymbol{x}, 0) B^{-1} (\phi(\boldsymbol{x}', 0) - g(\boldsymbol{x}', 0)) d\boldsymbol{x}' d\boldsymbol{x} \\ & - \int_\Omega \int_0^\tau \int_\Omega \int_0^\tau \left(\frac{\partial \mathbb{H}}{\partial \phi} \theta \right)^T R^{-1} (Y - \mathbb{H}(\phi)) dt' d\boldsymbol{x}' dt d\boldsymbol{x} \\ & = 0. \end{aligned} \tag{9}$$

Considering the three following integrations by parts, we can get rid of the partial derivatives of the admissible function θ in expression (9), i.e.

$$\begin{aligned}\int_\Omega \int_0^\tau \frac{\partial \theta}{\partial t}\lambda(\boldsymbol{x},t)dtd\boldsymbol{x} = \int_\Omega \theta^T(\boldsymbol{x},\tau)\lambda(\boldsymbol{x},\tau)d\boldsymbol{x} - \int_\Omega \theta^T(\boldsymbol{x},0)\lambda(\boldsymbol{x},0)d\boldsymbol{x} \\ - \int_\Omega \int_0^\tau \theta^T(\boldsymbol{x},t)\frac{\partial \lambda}{\partial t}dtd\boldsymbol{x},\end{aligned} \tag{10}$$

$$\int_\Omega \int_0^\tau \left(\frac{\partial \mathbb{M}}{\partial \phi}\theta\right)^T \lambda(\boldsymbol{x},t)dtd\boldsymbol{x} = \int_\Omega \int_0^\tau \theta^T (\frac{\partial \mathbb{M}}{\partial \phi})^* \lambda(\boldsymbol{x},t)dtd\boldsymbol{x}, \tag{11}$$

$$\begin{aligned}\int_\Omega \int_0^\tau \int_\Omega \int_0^\tau \left(\frac{\partial \mathbb{H}}{\partial \phi}\theta\right)^T R^{-1}(Y-\mathbb{H}(\phi))dt'd\boldsymbol{x}'dtd\boldsymbol{x} = \\ \int_\Omega \int_0^\tau \int_\Omega \int_0^\tau \theta^T (\frac{\partial \mathbb{H}}{\partial \phi})^* R^{-1}\left(Y-\mathbb{H}(\phi)\right) dt'd\boldsymbol{x}'dtd\boldsymbol{x}.\end{aligned} \tag{12}$$

In the two last equations, we have introduced adjoint operators $(\frac{\partial \mathbb{M}}{\partial \phi})^*$ and $(\frac{\partial \mathbb{H}}{\partial \phi})^*$ as compact notations of the integration by parts of the associated linear tangent operators. Gathering all these elements, equation (9) can be rewritten as

$$\begin{aligned}&\lim_{\beta\to 0}\frac{dJ}{d\beta} = \\ &\int_\Omega \theta^T(\boldsymbol{x},\tau)\lambda(\boldsymbol{x},\tau)d\boldsymbol{x} + \int_\Omega \theta^T(\boldsymbol{x},0)\left[\int_\Omega \left(B^{-1}(\phi(\boldsymbol{x}',0)-g(\boldsymbol{x}',0))-\lambda(\boldsymbol{x},0)\right)d\boldsymbol{x}'\right]d\boldsymbol{x} \\ &+ \int_\Omega \int_0^\tau \theta^T\left[\left(-\frac{\partial \lambda}{\partial t}+(\frac{\partial \mathbb{M}}{\partial \phi})^*\lambda\right) - \int_\Omega \int_0^\tau (\frac{\partial \mathbb{H}}{\partial \phi})^* R^{-1}(Y-\mathbb{H}(\phi))dt'd\boldsymbol{x}'\right]dtd\boldsymbol{x} = 0.\end{aligned} \tag{13}$$

Forward-backward equations Since the functional derivative must be null for arbitrary independent admissible functions in the three integrals of expression (13), all the other members appearing in the three integral terms must be identically null. We finally obtain a coupled system of forward and backward PDE's with two initial and end conditions:

$$\lambda(\boldsymbol{x},\tau) = 0 \tag{14}$$

$$-\frac{\partial \lambda}{\partial t}+(\frac{\partial \mathbb{M}}{\partial \phi})^*\lambda = \int_\Omega \int_0^\tau (\frac{\partial \mathbb{H}}{\partial \phi})^* R^{-1}(Y-\mathbb{H}(\phi))dtd\boldsymbol{x} \tag{15}$$

$$\lambda(\boldsymbol{x},0) = \int_\Omega \left(B^{-1}(\phi(\boldsymbol{x}',0)-g(\boldsymbol{x}',0))\right)d\boldsymbol{x}' \tag{16}$$

$$\frac{\partial \phi(\boldsymbol{x},t)}{\partial t} + \mathbb{M}(\phi(\boldsymbol{x},t)) = \int_\Omega \int_0^\tau Q\lambda(\boldsymbol{x}',t')dt'd\boldsymbol{x}'. \tag{17}$$

The forward equation (17) corresponds to the definition of the adjoint variable (7) and has been obtained introducing Q, the pseudo-inverse of Q^{-1}, defined as [1]:

$$\int_\Omega \int_0^\tau Q(\boldsymbol{x},t,\boldsymbol{x}',t')Q^{-1}(\boldsymbol{x}',t',\boldsymbol{x}'',t'')dt'd\boldsymbol{x}' = \delta(\boldsymbol{x}-\boldsymbol{x}'')\delta(t-t'').$$

We will discuss the discretization of these equations in the next section. Before that, we can make several remarks. First of all, we can see that eq. (14) constitutes an explicit end condition for the adjoint evolution model eq.(15). This adjoint evolution model can be integrated backward from the end condition assuming the knowledge of an initial guess for ϕ to compute the discrepancy $Y - \mathbb{H}(\phi)$. To perform this integration, we also need to have an expression of the *adjoint evolution operator*. Let us recall, that this operator is defined from an integration by part of the *linear tangent operator* associated to the evolution law operator. The analytic expression of such an operator is obviously not accessible in general. Nevertheless, it can be noticed that a discrete expression of this operator can be easily obtained from the discretization of the linear tangent operator. As a matter of fact, the adjoint of the linear tangent operator discretized as a matrix consists simply of the transpose of that matrix. Knowing a first solution of the adjoint variable, an initial condition for the state variable can be obtained from (16) and a pseudo inverse expression of the covariance matrix B. From this initial condition, (17) can be finally integrated forward.

Incremental state function The previous system can be modified slightly to produce an adequate initial guess for the state function. Considering a function of state increments linking the state function and an initial condition function, $\delta\phi = \phi - \psi$, and linearizing the operator $\mathbb{M}$ around the initial condition function ψ:

$$\mathbb{M}(\phi) = \mathbb{M}(\psi) + \frac{\partial \mathbb{M}}{\partial \psi}(\delta\phi),$$

it is possible to split equation (17) into two pde's with an explicit initial condition:

$$\psi(\boldsymbol{x}, 0) = g(\boldsymbol{x}, \Gamma(t_0)) \tag{18}$$

$$\frac{\partial \psi}{\partial t} + \mathbb{M}(\psi) = 0 \tag{19}$$

$$\frac{\partial \delta\phi}{\partial t} + (\frac{\partial \mathbb{M}}{\partial \psi})\delta\phi = \int_\Omega \int_0^\tau Q(\boldsymbol{x}', t', \boldsymbol{x}, t)\lambda(\boldsymbol{x}, t) dt d\boldsymbol{x}\,. \tag{20}$$

The first equation initializes function ψ as a signed distance function corresponding to the initial contours. Integrating forward equation (19) provides an initial guess of the state function (assuming the increment is initially null). This initial guess can then be used for the backward integration of the adjoint variable (15). The increment state function is updated by a forward integration of equation (20). These two last integrations successively iterated until convergence constitute the overall process.

4 Curve tracking implementation

In this section, we describe further the implementation of the method we propose for object contour tracking. We present the discretization scheme we used and give the analytic expression of the tangent linear operator associated to our evolution model.

4.1 Tangent linear operator

Considering a (nonlinear) operator $\mathbb{G}$ mapping one element of an initial functional space to another functional space, the tangent linear operator to $\mathbb{G}$ at point m is a linear operator defined by the limit:

$$\lim_{\gamma \to 0} \frac{\mathbb{G}(m + \gamma h) - \mathbb{G}(m)}{\gamma} = \frac{\partial \mathbb{G}}{\partial m}(h) \tag{21}$$

where γh is a small perturbation in the initial space. The tangent linear operator $\frac{\partial \mathbb{G}}{\partial m}$ is also known as the Gâteaux derivative of $\mathbb{G}$ at point m. Let us note that the Gâteaux derivative of a linear operator is the operator itself.

In our case, the evolution operator reads:

$$\mathbb{M}(\phi) = \boldsymbol{\nabla}\phi^{T}\,\boldsymbol{w} - \epsilon||\boldsymbol{\nabla}\phi||div\left(\frac{\boldsymbol{\nabla}\phi}{||\boldsymbol{\nabla}\phi||}\right).$$

This operator can be turned into a more tractable expression:

$$\mathbb{M}(\phi) = \boldsymbol{\nabla}\phi \cdot \boldsymbol{w} - \epsilon\left(\Delta\phi - \frac{\boldsymbol{\nabla}^{T}\phi\nabla^{2}\phi\boldsymbol{\nabla}\phi}{||\boldsymbol{\nabla}\phi||^{2}}\right).$$

After some calculations, the tangent linear operator to $\mathbb{M}$ at point ψ finally reads:

$$(\frac{\partial \mathbb{M}}{\partial \psi})\delta\phi = \boldsymbol{\nabla}\delta\phi \cdot \boldsymbol{w} - \epsilon\left[\Delta\delta\phi - \frac{\boldsymbol{\nabla}\psi^{T}\nabla^{2}\delta\phi\boldsymbol{\nabla}\psi}{||\boldsymbol{\nabla}\psi||^{2}} + 2\frac{\boldsymbol{\nabla}\psi^{T}\nabla^{2}\psi}{||\boldsymbol{\nabla}\psi||^{2}}\left(\frac{\boldsymbol{\nabla}\psi\boldsymbol{\nabla}\psi^{T}}{||\boldsymbol{\nabla}\psi||^{2}} - Id\right)\boldsymbol{\nabla}\delta\phi\right].$$

4.2 Algorithm specification

Up to now, we did not specified yet the observation function, Y, associated to our tracking problem. In order to have the simplest interaction as possible, we defined it in the same space as ϕ, that is to say $\mathbb{H} = Id$. As for the observation function corresponding to a measurement of the evolving object contours we chose to define it as the signed distance map to an observed closed curve, $g(\boldsymbol{x}, \Gamma(t))$. These curves are assumed to be generated by a basic threshold segmentation method or provided by some moving object detection method. Such observations are generally of bad quality. As a matter of fact, in the first case, very noisy curves are observed whereas in the later case, when the object motion is too slow, there is no detection at all. Concerning the motion field, $\boldsymbol{w}$, we used in this work an efficient and robust version of the Horn and Schunck optical-flow estimator [10].

Combining equations (14-15-16) and (18-19-20), we finally get the following iterative tracking system:

$$\psi^{k=0}(\boldsymbol{x}, t_0) = g(\boldsymbol{x}, \Gamma(t_0)) \tag{22}$$

$$\frac{\partial \psi^0}{\partial t} + \mathbb{M}(\psi^0) = 0 \tag{23}$$

$$\lambda^k(\tau) = 0 \tag{24}$$

$$-\frac{\partial \lambda^k}{\partial t} + (\frac{\partial \mathbb{M}}{\partial \psi^k})^* \lambda^k = \int_0^\tau R^{-1}\left(Y - \psi^k\right) \tag{25}$$

$$\lambda^k(t_0) = B^{-1}\delta\phi^k(t_0) \tag{26}$$

$$\frac{\partial \delta\phi^k}{\partial t} + (\frac{\partial \mathbb{M}}{\partial \psi^k})\delta\phi^k = \int_0^\tau Q\lambda^k(t). \tag{27}$$

A forward integration of the initial condition function (23) is done at the first iteration. Index k, represents the current iteration which consists of a backward integration of the adjoint function and a forward integration of the increment function (24 - 27). At the end of the iteration , ψ is updated according to the relation $\psi^{k+1} = \phi^k = \psi^k + \delta\phi^k$. We have chosen to represent the covariance matrix B as the diagonal matrix $B(\boldsymbol{x}, \boldsymbol{x}) = Id - e^{-|\psi(\boldsymbol{x}, t_0)|}$. In a similar way, we define matrix R from the observations Y, as

$$R = R_{min} + (R_{max} - R_{min})(Id - e^{-|Y(\boldsymbol{x}, t)|}).$$

This observation covariance matrix has therefore lower values in the vicinity of the observed curves and higher values faraway from them. When there is no observed curve, all the value of this covariance matrix are set to infinity. Otherwise, covariance matrix Q, has been fixed to a constant diagonal matrix.

4.3 Operator discretization

We will denote by $\phi^t_{i,j}$ the value of ϕ at image grid point (i, j) at time $t \in [0; \tau]$. Using (23) and a semi-implicit discretization scheme, the following discrete evolution model is obtained:

$$\frac{\phi^{t+\Delta t}_{i,j} - \phi^t_{i,j}}{\Delta t} + \mathbb{M}_{\phi^t_{i,j}} \phi^{t+\Delta t}_{i,j} = 0.$$

Considering ϕ_x and ϕ_y, the horizontal and vertical gradient matrices of ϕ, the discrete operator $\mathbb{M}$ is obtained as :

$$\mathbb{M}_{\phi^t_{i,j}} \phi^{t+\Delta t}_{i,j} = \begin{pmatrix} (\phi_x^{t+\Delta t})_{i,j} \\ (\phi_y^{t+\Delta t})_{i,j} \end{pmatrix}^T \boldsymbol{w} - \frac{\epsilon}{||\boldsymbol{\nabla}\phi^t_{i,j}||^2} \begin{pmatrix} -(\phi_y^t)_{i,j} \\ (\phi_x^t)_{i,j} \end{pmatrix}^T \nabla^2 \phi^{t+\Delta t}_{i,j} \begin{pmatrix} -(\phi_y^t)_{i,j} \\ (\phi_x^t)_{i,j} \end{pmatrix},$$

where we used usual finite differences for the advection term $\boldsymbol{\nabla}\phi^T \boldsymbol{w}$ and the Hessian matrix $\nabla^2\phi$. The discrete linear tangent operator (27) is similarly defined as:

$$\frac{\partial \mathbb{M}}{\partial \phi^t_{i,j}} \delta\phi^{t+\Delta t}_{i,j} = \mathbb{M}_{\phi^t_{i,j}} \delta\phi^{t+\Delta t}_{i,j} - \frac{2\epsilon \begin{pmatrix} A & B \end{pmatrix}}{||\boldsymbol{\nabla}\phi||^4} \begin{pmatrix} (\delta\phi_x^{t+\Delta t})_{i,j} \\ (\delta\phi_y^{t+\Delta t})_{i,j} \end{pmatrix},$$

where A and B are defined as:

$$A = \phi_x^t \phi_y^t (\phi_{xy}^t \phi_x^t - \phi_{xx}^t \phi_y^t) + (\phi_y^t)^2 (\phi_{yy}^t \phi_x^t - \phi_{xy}^t \phi_y^t),$$
$$B = \phi_x^t \phi_y^t (\phi_{xy}^t \phi_y^t - \phi_{yy}^t \phi_x^t) + (\phi_x^t)^2 (\phi_{xx}^t \phi_y^t - \phi_{xy}^t \phi_x^t).$$

As previously indicated, the discretization of the adjoint evolution model is obtained as the transposed matrix corresponding to the discretization of the derivative of the evolution law operator. Otherwise, we used a conjugated gradient optimization for the iterative solver involved in the implicit discretization.

5 Numerical results

In this section, we present results we obtained for three different kinds of image sequences. The first sequence is a 16 frames sequence presenting a moving skate fish on the sand (fig. 1). As this kind of fish possesses natural camouflage mechanisms, its luminance and texture are very similar to the surrounding sand. The contours of such an object are therefore really difficult to extract. For this sequence we used a simple segmentation algorithm based on selection of intensity level curves. To further demonstrate the robustness of our tracking approach, we only considered observations at every third frames (i.e for $k = 0, 3, 6, 9, 12, 15$). It can be noticed on the second row of figure 1 that the global shape and the successive locations of the skate have been well reconstructed at all time instants $\in [t_0, t_{16}]$. The noisy and instable observed contours have been smoothed in an appropriated way. For instance, it can be outlined that the technique has been able to cope with the partial occlusion generated by the lifting of the skate ventral fin (see images going from #5 to #15 in fig. 1).

The second sequence shows a person playing ping-pong. This is a 20 frames sequence where the camera is slightly moving backward. The observed curves are here provided by a motion detection method. For this sequence, no mask were detected between frames #5 and #14. Mask contours were thus only available for frame #0 to #4, and for frames #15 to #19. It can be noticed in addition that the observed curves are locally varying a lot between two consecutive frames. For example the racket is not always recovered by the motion detection technique. We show in figure 2 a sample of the observed curves and the corresponding results. We can observe that the recovered curves follow quite well the shape of the player even in the time interval for which no observation was available.

As a last example, we show on Figure 3 results obtained on a meteorological image sequence of the Meteosat infra red channel. The observed curve is a level line at a given value within a region of interest. We aim here therefore at tracking an isotemperature curve. The results demonstrate that the technique we propose keeps the adaptive topology property of level set methods, and in the same time, incorporates a consistent temporal prior for the curve evolution.

As for the computation time of the method, our code takes less than 1 minute for a forward-backward integration of a 20 frame sequence. It has to be noticed that our code, written in C, has not been particularly optimized. For instance, the different integration considered have been performed on the whole image plane. A significant reduction of the computational load could be probably obtained considering a narrow band technique [16].

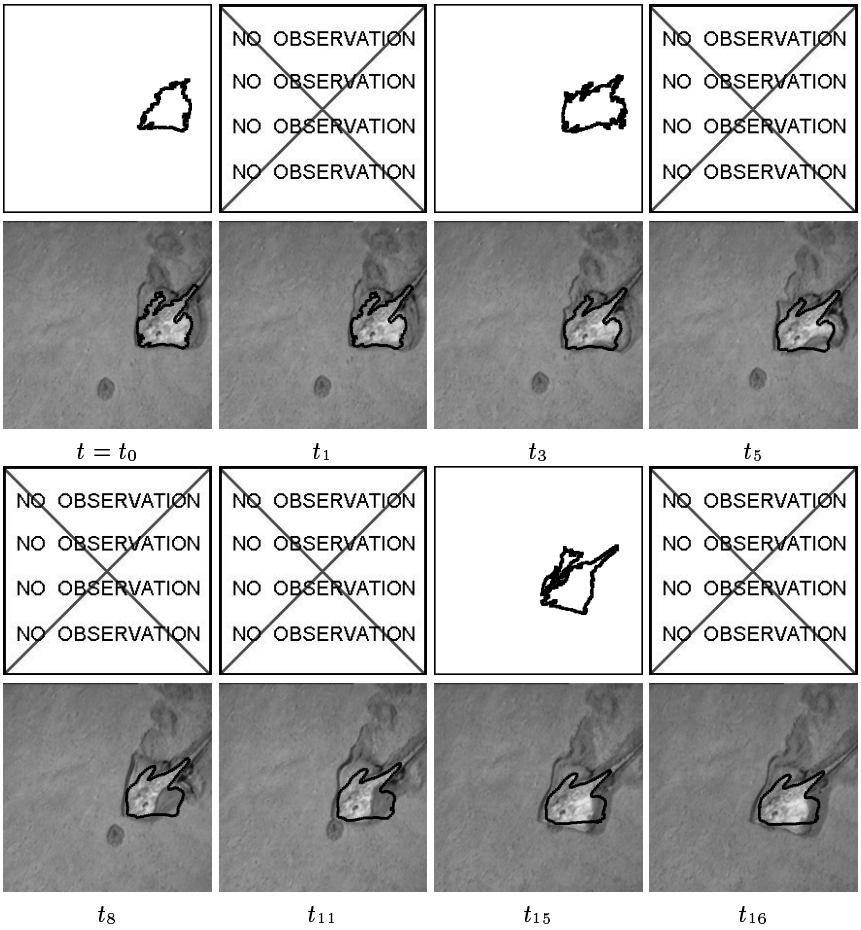

Fig. 1. Skate fish sequence. Top row: Sample of the observed curves. Bottom row: Recovered curve superimposed on the corresponding image

6 Conclusions

In this paper, we have presented a new technique for closed curves tracking. The proposed technique allows to estimate the contours location of a target object along an image sequence. In a similar way to a stochastic smoothing the estimation is led considering the whole set of the available measurements extracted from the image sequence. The technique is nevertheless totally different. It consists to integrate two coupled pde's representing the evolution of a state function and of an adjoint variable respectively. The method incorporates only few parameters. Similarly to Bayesian filtering techniques, these parameters mainly concern the definition of the different error models involved in the considered system. In our case, we have an additional parameter that weight the mean curvature motion appearing in our dynamic evolution model. The value of this parameter tunes the degree of smoothing of the curve (in our experiments it has been always fixed to the same value of 0.1).

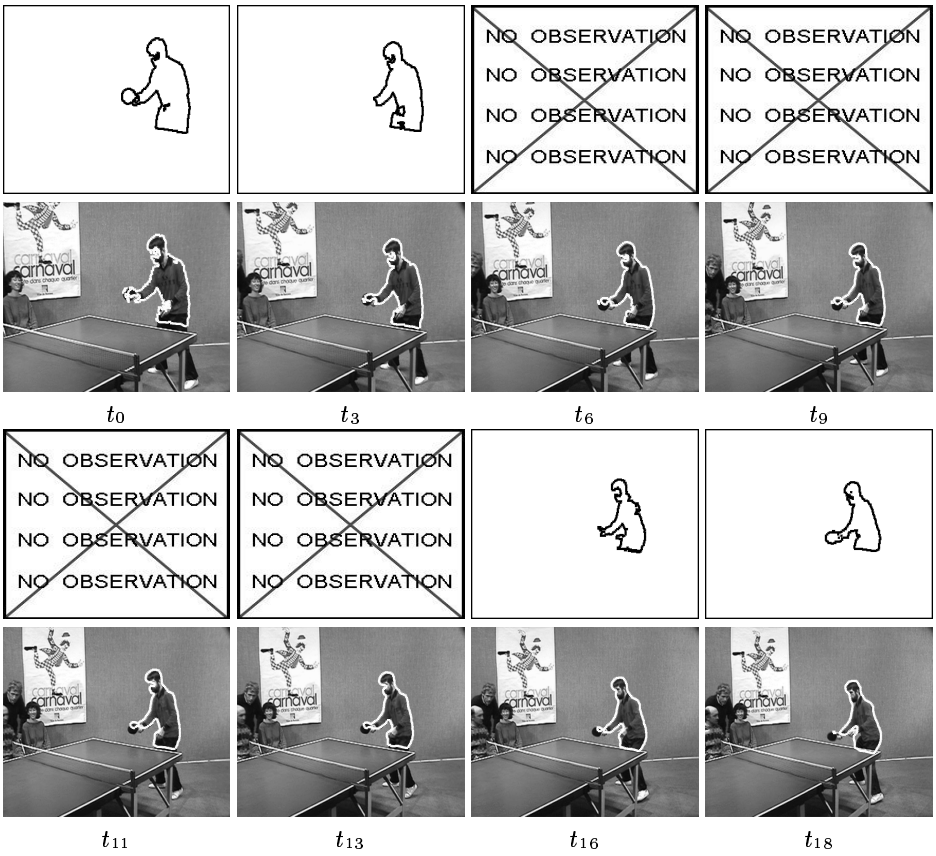

Fig. 2. Ping pong player sequence.Top row: Sample of the observed curves. Bottom row: Recovered curve superimposed on the corresponding image

Acknowledgments. This work was supported by the European Community through the IST FET Open FLUID project (`http://fluid.irisa.fr`).

References

1. A.F. Bennet. *Inverse Methods in Physical Oceanography*. Cambridge University Press, 1992.
2. A. Blake and M. Isard. *Active contours*. Springer-Verlag, London, England, 1998.
3. D. Cremers, F. Tischhäuser, J. Weickert, and C. Schnoerr. Diffusion snakes: introducing statistical shape knowledge into the Mumford-Shah functionnal. *Int. J. Computer Vision*, 50(3):295–313, 2002.
4. F.-X. Le Dimet and O. Talagrand. Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. *Tellus*, (38A):97–110, 1986.
5. R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky. Fast geodesic active contours. *IEEE Trans. on Image Processing*, 10(10):1467–1475, 2001.
6. R. Kimmel and A. M. Bruckstein. Tracking level sets by level sets: a method for solving the shape from shading problem. *Comput. Vis. Image Underst.*, 62(1):47–58, 1995.

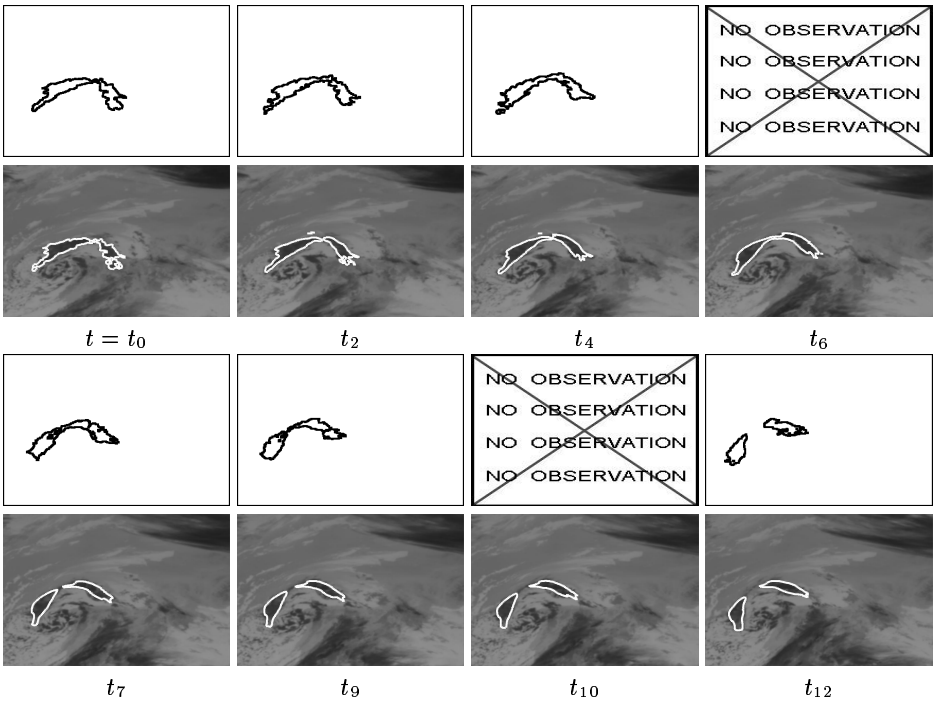

Fig. 3. Clouds sequence.Top row: Sample of observed curves. Bottom row: Recovered results superimposed on the corresponding image

7. M. Leventon, E. Grimson, and O. Faugeras. Statistical shape influence in geodesic active contours. In *Proc. Conf. Comp. Vision Pattern Rec.*, 2000.
8. A.R. Mansouri. Region tracking via level set PDEs without motion computation. *IEEE Trans. Pattern Anal. Machine Intell.*, 24(7):947–961, 2003.
9. P. Martin, P. Réfrégier, F. Goudail, and F. Guérault. Influence of the noise model on level set active contours segmentation. *IEEE Trans. Pattern Anal. Machine Intell.*, 26(6):799–803, 2004.
10. E. Mémin and P. Pérez. Dense estimation and object-based segmentation of the optical flow with robust techniques. *IEEE Trans. Image Processing*, 7(5):703–719, 1998.
11. M. Niethammer and A. Tannenbaum. Dynamic geodesic snakes for visual tracking. In *CVPR (1)*, pages 660–667, 2004.
12. S. Osher and J.A. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulation. *Journal of Computational Physics*, 79:12–49, 1988.
13. N. Paragios. A level set approach for shape-driven segmentation and tracking of the left ventricle. *IEEE trans. on Med. Imaging*, 22(6), 2003.
14. N. Paragios and R. Deriche. Geodesic active regions: a new framework to deal with frame partition problems in computer vision. *J. of Visual Communication and Image Representation*, 13:249–268, 2002.
15. N. Peterfreund. Robust tracking of position and velocity with Kalman snakes. *IEEE Trans. Pattern Anal. Machine Intell.*, 21(6):564–569, 1999.
16. J.A. Sethian. *Level set methods*. Cambridge University-Press, 1996.
17. O. Talagrand and P. Courtier. Variational assimilation of meteorological observations with the adjoint vorticity equation. I: Theory. *Journ. of Roy. Meteo. soc.*, 113:1311–1328, 1987.