

Continuous Tracking of Structures from an Image Sequence

Yann Lepoittevin^{1,2}, Dominique Béréziat³, Isabelle Herlin^{1,2} and Nicolas Mercier^{1,2}

¹Inria, B.P. 105, 78153 Le Chesnay, France

²CEREA, Joint Laboratory ENPC - EDF R&D, Université Paris-Est,
Cité Descartes Champs-sur-Marne, 77455 Marne la Vallée Cedex 2, France

³Université Pierre et Marie Curie, 4 place Jussieu, Paris 75005, France

Keywords: Tracking, Motion, Data Assimilation, Satellite image, Meteorology.

Abstract: The paper describes an innovative approach to estimate velocity on an image sequence and simultaneously segment and track a given structure. It relies on the underlying dynamics' equations of the studied physical system. A data assimilation method is applied to solve evolution equations of image brightness, those of motion's dynamics, and those of distance map modelling the tracked structures. Results are first quantified on synthetic data with comparison to ground-truth. Then, the method is applied on meteorological satellite acquisitions of a tropical cloud, in order to track this structure on the sequence. The outputs of the approach are the continuous estimation of both motion and structure's boundary. The main advantage is that the method only relies on image data and on a rough segmentation of the structure at initial date.

1 INTRODUCTION

The issue of detecting and tracking a structure covers a broad of major computer vision problems. Readers can refer to (Yilmaz et al., 2006), for instance, in order to get an extensive description on this issue. However, images may be noisy, as this is the case for satellite acquisitions, and assumptions on dynamics should then be involved. To our knowledge, no paper concerns a method that simultaneously estimates motion and segments/tracks a structure from only image data and a rough segmentation of the structure. However, methods exist that segment and track a structure, given motion field and initial segmentation (Peterfreund, 1999; Rathi et al., 2007; Avenel et al., 2009), or that track a structure and estimate its motion if this structure has been accurately segmented on the first image (Bertalmío et al., 2000).

The use of data assimilation recently emerged in the image processing community. In (Béréziat and Herlin, 2011), motion estimation is discussed, and solutions are described for processing noisy images. In (Papadakis and Mémin, 2008), an incremental 4D-Var is used, that also computes motion field and tracks a structure, but relies, as inputs, on both image data and accurate segmentation of the structure on the whole sequence. Our approach has the advantage to simultaneously solve the issues of motion estimation,

detection, segmentation and tracking of the structure, based, as only inputs, on image data and their gradient values.

Section 2 describes the main mathematical components of the approach. Sections 3 and 4 discuss results obtained on synthetic data and meteorological satellite acquisitions. Section 5 concludes with some remarks and perspectives on the research work.

2 MATHEMATICAL SETTING

Our approach is based on a 4D-Var data assimilation algorithm, used to estimate motion on the sequence and track a structure.

Ω denotes the bounded image domain, on which pixels $\mathbf{x} = (x \ y)^T$ are considered, $[0, T]$ the studied temporal interval, and $A = \Omega \times [0, T]$.

2.1 Model of Structure and Input Data

Let define the structure tracked along the image sequence by an implicit function ϕ (see Figure 1): each pixel \mathbf{x} at date t gets for value its signed distance to the current position of the structure boundary.

Observations, used during the assimilation, are images themselves and their contour points, obtained by thresholding the maxima of the gradient norm.

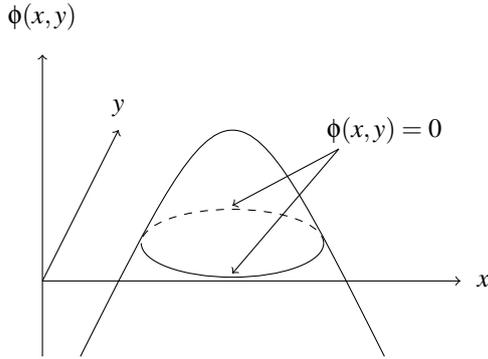


Figure 1: Implicit representation of structure's boundary.

2.2 Evolution Model

The assumption on dynamics is the Lagrangian constancy of velocity $\mathbf{w} = (u \ v)^T$, rewritten as:

$$\frac{du}{dt} = 0 \Leftrightarrow \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = 0 \quad (1)$$

$$\frac{dv}{dt} = 0 \Leftrightarrow \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = 0 \quad (2)$$

A pseudo-image I_s is defined, that satisfies the optical flow constraint:

$$\frac{\partial I_s}{\partial t} + \nabla I_s \cdot \mathbf{w} = 0 \quad (3)$$

The pseudo-image is compared to satellite data during the optimization process: they have to be almost identical at acquisition dates. The implicit function ϕ is assumed to satisfy the same heuristics, as the structure moves accordingly to image evolution:

$$\frac{\partial \phi}{\partial t} + \nabla \phi \cdot \mathbf{w} = 0 \quad (4)$$

The state vector, defined as $\mathbf{X} = (u \ v \ I_s \ \phi)^T$, satisfies the evolution system (1, 2, 3, 4), summarized by:

$$\frac{\partial \mathbf{X}}{\partial t} + \mathbb{M}(\mathbf{X}(t)) = 0 \quad (5)$$

2.3 4D-Var Data Assimilation

In order to estimate \mathbf{X} , and obtain motion estimation and tracking of the structure, the 4D-Var algorithm considers the following three equations:

$$\frac{\partial \mathbf{X}}{\partial t}(\mathbf{x}, t) + \mathbb{M}(\mathbf{X})(\mathbf{x}, t) = 0 \quad (6)$$

$$\mathbf{X}(\mathbf{x}, 0) = \mathbf{X}_b(\mathbf{x}) + \boldsymbol{\varepsilon}_b(\mathbf{x}) \quad (7)$$

$$\mathbb{H}(\mathbf{X}, \mathbf{Y})(\mathbf{x}, t) = \boldsymbol{\varepsilon}_R(\mathbf{x}, t) \quad (8)$$

The first one is the evolution equation. One can notice that $\mathbf{X}(\mathbf{x}, t)$, for any t , is determined from $\mathbf{X}(\mathbf{x}, 0)$ and the integration of Equation (6).

Equation (7) corresponds to the knowledge, that is available on the state vector at initial date 0, and expressed as the background value $\mathbf{X}_b(\mathbf{x})$. The solution $\mathbf{X}(\mathbf{x}, 0)$, estimated by 4D-Var, should stay close to this background value. However, as it is uncertain, an error term, $\boldsymbol{\varepsilon}_b(\mathbf{x})$, is considered. No knowledge is available on the initial velocity field and its background value is null; the background on the pseudo-image I_s is the first image of the sequence; and the background of ϕ , denoted ϕ_b , roughly defines the structure to be tracked. Let \mathbb{P} be the projection of the state vector on components I_s and ϕ , Equation (7) is rewritten as:

$$\mathbb{P}(\mathbf{X}(0)) = \mathbb{P}(\mathbf{X}_b) + \boldsymbol{\varepsilon}_b \quad (9)$$

Equation (8), named observation equation, links the observations to the state vector \mathbf{X} . The observation vector \mathbf{Y} includes the image acquisitions and a distance map to the contour points, that have been computed on these acquisitions. This distance map is denoted by $D_c(\mathbf{x}, t)$. \mathbb{H} denotes the observation operator, split in two parts: $\mathbb{H} = (\mathbb{H}_I \ \mathbb{H}_\phi)^T$. \mathbb{H}_I compares pseudo-images I_s to image observations I :

$$\mathbb{H}_I(\mathbf{X}, \mathbf{Y}) = I_s - I = \boldsymbol{\varepsilon}_I \quad (10)$$

Their discrepancy is described by the error $\boldsymbol{\varepsilon}_I$. \mathbb{H}_ϕ compares ϕ to the distance map $D_c(\mathbf{x}, t)$. The absolute value of ϕ should be almost equal to D_c :

$$\mathbb{H}_\phi(\mathbf{X}, \mathbf{Y}) = \frac{4e^{-a\phi}}{(1+e^{-a\phi})^2} (|\phi| - D_c) = \boldsymbol{\varepsilon}_\phi \quad (11)$$

The function $\frac{4e^{-a\phi}}{(1+e^{-a\phi})^2}$ is introduced to decrease the impact of contours, that do not belong to the boundary of the tracked structure. Parameter a controls the slope of the function. The more a increases, the more the function looks like an indicator function: only pixels in a small neighborhood of structure's boundary are considered by \mathbb{H}_ϕ during the optimization process.

Errors $\boldsymbol{\varepsilon}_b$, $\boldsymbol{\varepsilon}_I$, $\boldsymbol{\varepsilon}_\phi$ are supposed Gaussian, zero-mean, not correlated, with respective variance B , R_I , R_ϕ . Solving System (6, 9, 10, 11) is then written as the minimization of the following cost function:

$$J(\mathbf{X}(0)) = \int_A \frac{\boldsymbol{\varepsilon}_I(\mathbf{x}, t)^2}{R_I(\mathbf{x}, t)} + \int_A \frac{\boldsymbol{\varepsilon}_\phi(\mathbf{x}, t)^2}{R_\phi(\mathbf{x}, t)} + \int_\Omega \frac{\boldsymbol{\varepsilon}_b(\mathbf{x})^2}{B(\mathbf{x})} \quad (12)$$

Let $\boldsymbol{\lambda}$ denote the adjoint variable, verifying:

$$\boldsymbol{\lambda}(T) = 0 \quad (13)$$

$$-\frac{\partial \boldsymbol{\lambda}(t)}{\partial t} + \left(\frac{\partial \mathbb{M}}{\partial \mathbf{X}} \right)^* \boldsymbol{\lambda}(t) = \mathbb{H}^* R^{-1} \mathbb{H}(\mathbf{X}, \mathbf{Y})(t) \quad (14)$$

with the adjoint operator $\left(\frac{\partial \mathbb{M}}{\partial \mathbf{X}} \right)^*$, which is defined by: $\langle Z\boldsymbol{\eta}, \boldsymbol{\lambda} \rangle = \langle \boldsymbol{\eta}, Z^* \boldsymbol{\lambda} \rangle$. The adjoint operator $\left(\frac{\partial \mathbb{M}}{\partial \mathbf{X}} \right)^*$

is automatically generated from the discrete operator \mathbb{M} by an efficient automatic differentiation software (Hascoët and Pascual, 2004). Then, gradient of J is:

$$\frac{\partial J}{\partial \mathbf{X}(0)} = \mathbb{P}^T B^{-1} [\mathbb{P}(\mathbf{X}(0)) - \mathbb{P}(\mathbf{X}_b)] + \lambda(0) \quad (15)$$

Minimization is achieved by a steepest method and the L-BFGS algorithm (Zhu et al., 1997).

3 TWIN EXPERIMENT

A sequence of five Image Observations (see Figure 3), $I_i = I(t_i)$ for $i = 1$ to 5, is generated by integrating model \mathbb{M} from initial conditions, displayed in Figure 2. Contours are first computed on images I_i . Then Distance Map Observations $D_c(\mathbf{x}, t_i)$ are derived. This is done in order to estimate motion on the whole image sequence and track the brightest square.

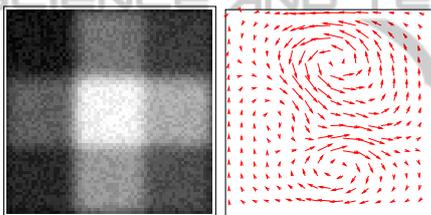


Figure 2: Left : initial image. Right : initial motion field.

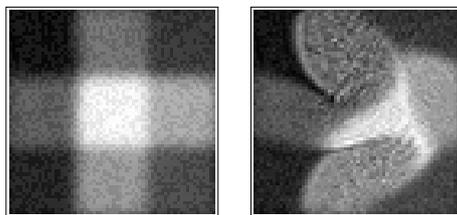


Figure 3: Image Observations at t_1 and t_5 .

After assimilation, pseudo-images are compared to Image Observations. They are almost identical and their correlation measure is over 0.999. At dates t_i , the region of positive values of ϕ , corresponding to the inside of the tracked structure, is compared to the contour points, see Figure 4.

The simulation, that provides Image Observations, also provides ground-truth of the velocity field. This allows to perform statistics on the discrepancy between estimated motion and ground-truth: average error is around 1% in norm and less than one degree in orientation. Motion estimated on the whole image is displayed on Figure 5 with the coloured representa-

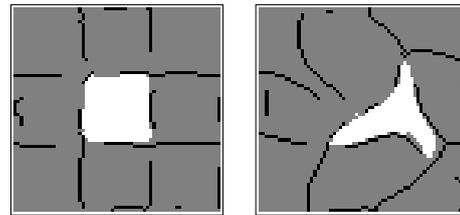


Figure 4: Comparison of ϕ and contour points on Image Observation. Left: t_1 , Right: t_5 .



Figure 5: Left: Ground-truth. Right: Assimilation result.

tion tool of the Middlebury database¹: there is no visible difference between estimation and ground-truth.

4 METEOSAT IMAGES

The assimilation method is applied on a Meteosat sequence and displayed on Figure 6.

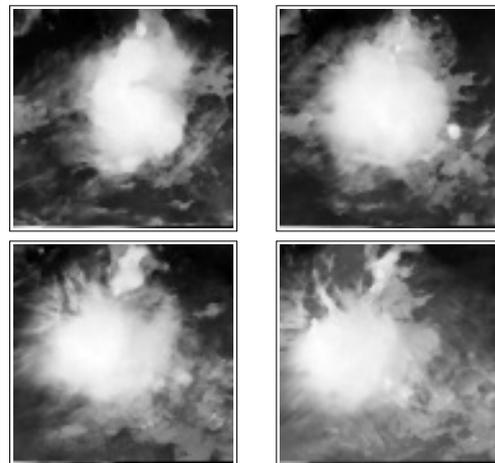


Figure 6: Images of a tropical cloud in the infrared domain. From left to right, up to down.

Figure 7, first column, displays the contour points, used to calculate the Distance Map Observations $D_c(\mathbf{x}, t_i)$. Pseudo-images, obtained as result of data assimilation, are displayed on the second column. On the third column, the blue curve corresponds to the

¹<http://vision.middlebury.edu/flow/>

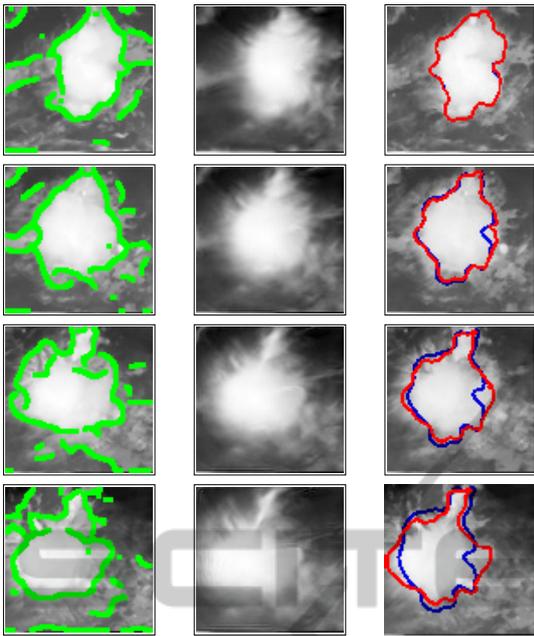


Figure 7: Left: Contours on observations. Middle: pseudo-images. Right: red is the result with \mathbb{H}_ϕ , blue is without.

result of assimilation without the term \mathbb{H}_ϕ in Equation 11, while the red one is obtained with \mathbb{H}_ϕ . As it can be seen, including constraints on ϕ allows to improve the accuracy of segmentation. Motion field is estimated on the whole image, but Figure 8 focuses on the boundary of the structure. It shows that the resulting velocity vectors correctly assess displacement of the structure along the sequence. The displacement estimated at the boundary of the tracked structure, superposed on satellite images, is shown on Figure 8.

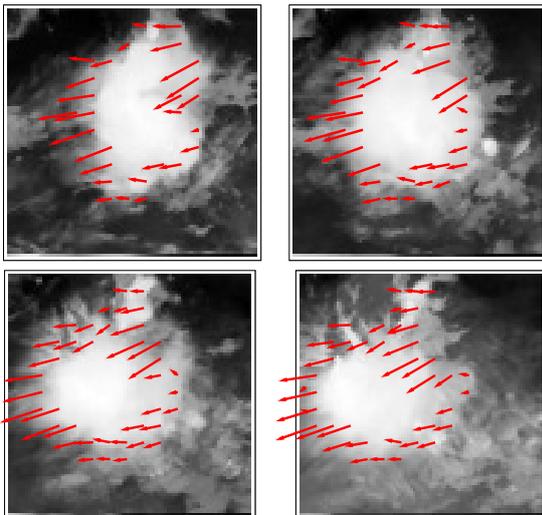


Figure 8: Motion result superposed to images on the boundary of the tracked structure. From left to right, up to down.

5 CONCLUSIONS

The paper describes an innovative approach enabling to estimate motion, segment and track a structure on images, such as, for instance, a cloud on a satellite sequence. The approach is based on 4D-Var data assimilation, and the state vector includes an implicit function ϕ modelling the boundary of the tracked structure. Results are given on synthetic and meteorological data. Additional experiments have been conducted, not described in the paper, that confirm the robustness of the approach.

The main perspective of this research is to extend the method to multi-structures tracking and to a space-time segmentation process. An additional interesting perspective is to allow uncertainty on the dynamic equations and take into account a model error in Equation (6).

REFERENCES

- Avenel, C., Mémin, E., and Pérez, P. (2009). Tracking closed curves with non-linear stochastic filters. In *Conference on Space-Scale and Variational Methods*.
- Béréziat, D. and Herlin, I. (2011). Solving ill-posed image processing problems using data assimilation. *Numerical Algorithms*, 56(2):219–252.
- Bertalmio, M., Sapiro, G., and Randall, G. (2000). Morphing active contours. *Pat. Anal. and Mach. Int.*, 22(7):733–737.
- Hascoët, L. and Pascual, V. (2004). Tapenade 2.1 user's guide. Technical Report 0300, INRIA.
- Papadakis, N. and Mémin, E. (2008). Variational assimilation of fluid motion from image sequence. *SIAM Journal on Imaging Sciences*, 1(4):343–363.
- Peterfreund, N. (1999). Robust tracking of position and velocity with kalman snakes. *Pat. Anal. and Mach. Int.*, 21(6):564–569.
- Rathi, Y., Vaswani, N., Tannenbaum, A., and Yezzi, A. (2007). Tracking deforming objects using particle filtering for geometric active contours. *Pat. Anal. and Mach. Int.*, 29(8):1470–1475.
- Yilmaz, A., Javed, O., and Shah, M. (2006). Object tracking: A survey. *ACM Computing Surveys*, 38(4 2006):13.
- Zhu, C., Byrd, R., Lu, P., and Nocedal, J. (1997). L-bfgs-B: Algorithm 778: L-bfgs-B, FORTRAN routines for large scale bound constrained optimization. *ACM Transactions on Mathematical Software*, 23(4):550–560.