Exp Fluids
DOI 10.1007/s00348-010-0926-9

RESEARCH ARTICLE

Identification of velocity fields for geophysical fluids from a sequence of images

Didier Auroux · Jérôme Fehrenbach

Received: 7 April 2009 / Revised: 15 June 2010 / Accepted: 26 June 2010

Abstract We propose an algorithm to estimate the motion between two images. This algorithm is based on the nonlinear brightness constancy assumption. The number of unknowns is reduced by considering displacement fields that are piecewise linear with respect to each space variable, and the Jacobian matrix of the cost function to be minimized is assembled rapidly using a finite-element method. Different regularization terms are considered, and a multiscale approach provides fast and efficient convergence properties. Several numerical results of this algorithm on simulated and experimental geophysical flows are presented and discussed.

1 Introduction

Estimating the motion of a fluid is of great interest, particularly in geophysics where the fluid can be the ocean. Applications of the motion estimation in this domain include the assimilation of image data in oceanographic models, and a possible improvement in the forecasts. Indeed, the poor predictability of extreme geophysical events (e.g. El Niño for the ocean) has dramatic consequences. These events are usually visible on satellite images several days before they become extreme, but they are generally not used for forecast, and these data are not considered within the data assimilation process. Satellite images contain a huge amount of data that should be assimilated in order to improve the forecast quality.

Several ideas have been recently developed to assimilate image data. A first idea consists in identifying some characteristic structures of the image and then tracking them in time. This is currently developed in meteorology, using an adaptive thresholding technique for radiance temperatures in order to identify and track several cells (Michel and Bouttier 2006). Another idea is to consider a dual problem and to create some model images, coming from the numerical model itself, and to compare the satellite images with these model images, using for example a curvlet approach (Ma et al. 2006). The main difficulty comes from the definition of an image model, able to create a synthetic image from a numerical model solution (Huot et al. 2006; Isambert et al. 2007).

The main concept of this paper is to define a fast and efficient way to identify, or extract, pseudo-observations of velocity from several images (or a complete sequence of images). Assuming this point, we would then be able to obtain billions of pseudo-observations, mainly Lagrangian velocities, corresponding to the extracted velocity fields, that could be considered in the usual data assimilation processes. The advantage of such an approach is to provide an information on the velocity, which is a state variable of all geophysical models, as it is much more easy to assimilate data that are directly related to the state variables.

The hypothesis that is underlying this work is that the gray level of the points are preserved during the motion, this is known as the constant brightness hypothesis. The constant brightness hypothesis was introduced in Horn and Schunk (1981), and the linearized approximation derived

D. Auroux (✉)
Laboratoire J. A. Dieudonné, Université de Nice Sophia Antipolis, 06108 Nice cedex 2, France
e-mail: auroux@unice.fr

D. Auroux
INRIA, Grenoble, Rhône-Alpes, France

J. Fehrenbach
Institut de Mathématiques de Toulouse, Université Paul Sabatier Toulouse 3, 31062 Toulouse cedex 9, France

Published online: 10 July 2010

from this hypothesis is the cornerstone of optical flow methods (Lucas and Kanade 1981; Ananden 1989; Beauchemin and Barron 1995).

This hypothesis is justified here in the framework of oceanography, as the object of interest, allowing us to track the fluid and identify its velocity, is usually a passive tracer, at least on relative short time periods: chlorophyll, sea surface temperature, chemical pollutants (e.g. hydrocarbons)… All these tracers do not interact with the water on a short time period, and they are passively transported by the fluid.

Fluid motion estimation was considered by numerous authors for several years. An established technique for experimental measurements is particle imaging velocimetry (PIV), where tracer particles are included inside the flow, and spatio-temporal cross-correlation techniques are used to estimate the motion of sub-windows of the domain. The quite large size of the interrogation windows usually implies some poor local motion estimation, as several particles with different velocities are present (Adrian 1991; Coriolis 2010).

A variational approach is presented in Alvarez et al. (2007) where a cost function similar to ours is minimized in the context of incompressible fluids. But an initial guess provided by cross-correlation is required, as the authors consider a 3D problem, in which the quality of the initial condition is essential, due to low particle density. The constant brightness assumption was replaced by an integrated continuity equation (Fitzpatrick 1988) in order to take into account the spreading of intensity sources. This continuity equation was integrated with respect to time in Corpetti et al. (2002). It did not appear necessary in our application to use this approach, an integration with respect to time of the constant brightness assumption provided satisfactory results. In the study by Cuzol et al. (2007), the motion estimation is replaced by a parameter estimation in a small dimensional vector space: a small number of vortex and source particles that describe the motion are retrieved.

Several approaches have also been specifically proposed for the estimation of fluid flow velocity, based on models derived from the physical laws governing fluid mechanics (e.g. the continuity equation) (Fitzpatrick 1985; Larsen et al. 1998; Corpetti et al. 2002; Mémin and Pérez 2002; Bruhn et al. 2006). Several regularizations have also been studied, from standard first-order norms to second-order div-curl operators Yuan et al. (2007).

We propose here to use an integrated version of the constant brightness hypothesis. Instead of linearizing the constant brightness hypothesis like in standard optical flow techniques, we define a nonlinear cost function that takes into account the fact that time sampling occurs at a finite rate. Moreover, the combination of bilinear interpolation of gray levels with spatial regularization provides an accurate estimation of sub-pixel motion.

The vector fields that we consider are piecewise linear with respect to each space variable, on squares defined by a grid. A cost function similar to ours and more general cost functions are studied in Papenberg et al. (2006), where the minimization uses a gradient descent. A contribution of the present paper is that the Jacobian of the cost function is assembled rapidly using a finite-element method (only one reading of the data is required) and a Gauss-Newton descent is performed. To our knowledge, this was not reported before. This method is coupled with a hierarchical approach, and the cost function is minimized in nested subspaces of admissible displacement vector fields.

This nonlinear cost function and the finite elements-like computation of the Jacobian have been introduced in Fehrenbach and Masmoudi (2008). We refer to this short note for details about the discretization of the displacement fields and the way to rapidly assemble the Jacobian matrix. We introduce here several improvements of the algorithm introduced in Fehrenbach and Masmoudi (2008) (regularization, multiscale approach, quality estimator, ...), and we also present several applications to experimental data.

This paper is organized as follows. In Sect. 2, we present a description of our algorithm with different regularization terms, a way to consider a multiscale approach, and an estimator of the quality of the results. Then, in Sect. 3 we present the results of extensive numerical experiments on simulated data. Section 4 is devoted to numerical results on two different kinds of experimental data. Finally, some concluding remarks and perspectives are given in Sect. 5.

2 Description of the algorithm

This section is devoted to the description of the algorithm that we use. We refer to the short note (Fehrenbach and Masmoudi 2008) for details about some of the main ingredients, namely the cost function and the procedure of assembling the Jacobian matrix.

2.1 Nonlinear cost function

The velocity estimation relies on the following hypothesis: the brightness of the points does not change between successive frames (at least when the time step between successive images is small enough). Let Ω denote the rectangular domain where the images are defined. The motion between the instants t_0 and t_1 where the images are I_0 and I_1 is then the vector field (u, v) such that for every point $(x, y) \in \Omega$,

$$I_1(x + u(x, y), y + v(x, y)) = I_0(x, y). \quad (1)$$

A vector field satisfying Eq. 1 is not unique, this is known as the aperture problem in optical flow. Moreover, measurements error make the equality Eq. 1 unlikely to be strictly satisfied. We propose a least squares optimization to replace the exact equality Eq. 1.

Denote $\mathcal{L}$ the vector space of Lipschitz vector fields. Consider the following function:

$$F(I_0, I_1; u, v)(x, y) = I_1(x + u(x, y), y + v(x, y)) - I_0(x, y), \tag{2}$$

where the vector field $(u, v) \in \mathcal{L}$ and the images I_0 and I_1 are continuously differentiable.

The map F is differentiable with respect to (u, v), and for $(u, v) \in \mathcal{L}$ and $\mathbf{d} = (du, dv) \in \mathcal{L}$, the Jacobian DF of F is given by

$$DF(u, v).\mathbf{d}(x, y) = \nabla I_1(x + u(x, y), y + v(x, y)).\mathbf{d}(x, y). \tag{3}$$

The optimal displacement vector field between the images I_0 and I_1 minimizes the following cost function:

$$J(u, v) = \frac{1}{2}\int_\Omega [F(I_0, I_1; u, v)(x, y)]^2 \mathrm{d}x\mathrm{d}y + \frac{1}{2}\alpha R(u, v), \tag{4}$$

where $R(u, v)$ is a spatial regularization term and α is the regularization factor. Our numerical experiments used different spatial regularization terms, these are described in Sect. 2.2 below.

The minimum of J is estimated in a nested sequence of subspaces of $\mathcal{L}$. On a small dimensional subspace, the optimization is efficient and the algorithm is not trapped in local minima. The result is used as an initial guess for the minimization of J in a larger subspace. The subspaces and the minimization strategy are described in Sect. 2.3.

2.2 Regularization

The following regularization terms were particularly used in our numerical experiments. In the following definitions, $\|.\|$ represents the L^2 norm of a scalar or vector field on the image.

$$R_0(u, v) = \|u\|^2 + \|v\|^2, \tag{5}$$

$$R_1(u, v) = \|\nabla u\|^2 + \|\nabla v\|^2 = \|\partial_x u\|^2 + \|\partial_y u\|^2 + \|\partial_x v\|^2 + \|\partial_x v\|^2, \tag{6}$$

$$R_{div}(u, v) = \|div(u, v)\|^2 = \|\partial_x u + \partial_y v\|^2, \tag{7}$$

$$R_{curl}(u, v) = \|curl(u, v)\|^2 = \|\partial_y u - \partial_x v\|^2, \tag{8}$$

$$R_{div/curl}(u, v) = \|div(u, v)\|^2 + \|curl(u, v)\|^2 = \|\partial_x u + \partial_y v\|^2 + \|\partial_y u - \partial_x v\|^2, \tag{9}$$

$$R_{\nabla div}(u, v) = \|\nabla div(u, v)\|^2 = \left\|\partial^2_{xx} u + \partial^2_{xy} v\right\|^2 + \left\|\partial^2_{xy} u + \partial^2_{yy} v\right\|^2, \tag{10}$$

$$R_{\nabla div/\nabla curl}(u, v) = \|\nabla div(u, v)\|^2 + \|\nabla curl(u, v)\|^2 = \left\|\partial^2_{xx} u + \partial^2_{xy} v\right\|^2 + \left\|\partial^2_{xy} u + \partial^2_{yy} v\right\|^2 + \left\|\partial^2_{xy} u - \partial^2_{xx} v\right\|^2 + \left\|\partial^2_{yy} u - \partial^2_{xy} v\right\|^2. \tag{11}$$

$$R_2(u, v) = \|\Delta u\|^2 + \|\Delta v\|^2. \tag{12}$$

The choice of the regularization norm largely depends on the physical problem, but also on the well-posedness of the optimization problem.

Without any regularization, the minimization problem is ill-posed, and there is indeed a need for a regularization term. We naturally try R_0, R_1 and R_2 regularizations as they are usually considered in optimization problems. They make the optimization problem well-posed (Tikhonov regularization, see e.g. Tikhonov and Arsenin (1977)). These norms ensure some regularity of the solution: either a small displacement field (R_0), or a regular (or smooth) field (R_1, R_2). These regularization terms also ensure the well-posedness of the optimization problem. For instance, R_1 and R_2 terms mathematically ensure a high order of regularity of the solution (respectively in H^1 and H^2 Sobolev spaces).

Also, R_{div} is probably the most physical regularization for geophysical applications, as one usually assumes that the fluid has a null divergence. The space H^{div} (defined as the set of H^1 functions of null divergence) is commonly used in fluid mechanics, as it allows one to intrinsically take into account the null divergence condition. But in practice, this choice is not so straightforward in our framework, as 2D images of a 3D phenomenon will not have null divergence fields, and synthetic data may also not be of null divergence.

We also tried several other regularizations, involving the curl of the field, or second-order regularizations based on the variations of the divergence field.

In all the cases, the regularization term is the square of a linear operator S. Below, the unifying following notation is used:

$$R(u, v) = ||S(u, v)||^2,$$

where S is a linear operator. Some scalar coefficients have also been considered in order to weight the different terms of a given regularization. The weighting coefficient α in Eq. 4 was chosen in order to get the best compromise between a small first term of the cost function at the end of the optimization and a reasonable regularization term, like in the L-curve approach (Hansen 1992). It has been found

by successive trials, where α was increased by a factor of 10.

2.3 Multiscale approach and optimization

The minimization of the cost function J is performed in nested subspaces:

$$\mathcal{C}_{16} \subset \mathcal{C}_8 \subset \mathcal{C}_4 \subset \mathcal{C}_2 \subset \mathcal{C}_1,$$

where the set $\mathcal{C}_q$ of admissible displacement fields at the scale q contains piecewise affine vector fields with respect to each space variable, on squares of size $q \times q$ pixels. A multiscale approach indeed improves the quality of the results (by removing local minima) and also reduces the computation time (see e.g. Heitz et al. (1994), Hellier et al. (1999)). We present here an adaptation of the method that was introduced in Fehrenbach and Masmoudi (2008). The difference with hierarchical techniques issued from the optical flow family (e.g. Mémin and Perez (1998); Ruhnau et al. (2005)) is that we do not linearize the cost function. This should help finding large displacements, where the domain of linearity of the luminance function is not valid, as illustrated in Sect. 2.4. Another innovation of the present approach is the efficient computation of the product DF^TDF of the Jacobian of the first term of the cost function Eq. 4 by its transpose. This efficient computation comes from the observation that this matrix is sparse and can be assembled like a finite-element matrix using one loop over the data.

The space $\mathcal{C}_{16}$ is typically of small dimension, hence the minimization of J on $\mathcal{C}_{16}$ is fast and robust when a zero vector field is used as initial guess. The optimal vector field obtained at a given scale in the space $\mathcal{C}_q$ is used as initial guess to find the minimum at the finer scale in the space $\mathcal{C}_{q/2}$.

We assume for simplicity that the dimensions (n, m) of the images satisfy $n = 16N + 1$ and $m = 16M + 1$. At the scale of $q \times q$ pixels, control points are defined with coordinates in $X_q \times Y_q$, where $X_q = (1, q + 1, 2q + 1, ..., N'q + 1)$, $Y_q = (1, q + 1, 2q + 1, ..., M' \quad q + 1)$, $M' = 16M/q$ and $N' = 16N/q$, see Fig. 1 (left). Let $\mathcal{V}_q = X_q \times Y_q$ be the set of all control points, and $\mathcal{R}_q$ the set of all squares of the coarse grid. The vector space $\mathcal{C}_q$ of vector fields that are bilinear on each square of $\mathcal{R}_q$ is of dimension $2(N' + 1)(M' + 1)$.

The optimization of the nonlinear cost function J on $\mathcal{C}_q$ is performed by Gauss-Newton's method. When an initial guess (u^0, v^0) is given, the k-th iteration reads

$$\left(u^k, v^k\right) := \left(u^{k-1}, v^{k-1}\right) + \left(du^k, dv^k\right),$$

where (du^k, dv^k) solves

$$\left(DF^TDF + \alpha S^TS\right)(du, dv) = -DF^TF - \alpha S^TS(u, v), \qquad (13)$$

where $F = F(I_0, I_1; u^{k-1}, v^{k-1})$ is the error, $DF = DF(I_0, I_1; u^{k-1}, v^{k-1})$ is the Jacobian matrix of the error, and S is the linear operator associated with the regularization term.

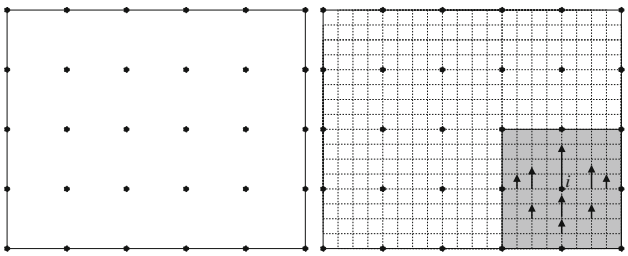

Fig. 1 *Left* the control points; here $N = 5$, $M = 4$. At this scale, the vector fields are piecewise affine in the squares bounded by the control points. *Right* one elementary displacement vector field $\mathbf{e}_i^y$: $\mathbf{e}_i^y$ is zero at the points of the *nonshaded area*, it is directed along the vertical with norm 1 at the vertex i, and is piecewise linear with respect to each space variable (for reason of clarity, the values at only some points are indicated)

Let $\mathcal{E}$ be the orthonormal basis of $\mathcal{C}_q$ defined as follows: $\mathcal{E} = \left(\mathbf{e}_i^x\right)_{i\in\mathcal{V}} \cup \left(\mathbf{e}_i^y\right)_{i\in\mathcal{V}}$, where $\mathbf{e}_i^x$ is the vector field that is zero at every control point of $\mathcal{V}_q$ except at the vertex i where it is directed along the horizontal axis, and $\mathbf{e}_i^y$ is the vector field that is zero at every control point of $\mathcal{V}_q$ except at the vertex i where it is directed along the vertical axis, see Fig. 1 (right) for an example.

Since the elementary displacements $\mathbf{e}_i^x$ and $\mathbf{e}_i^y$ are non zero in the rectangles adjacent to the vertex i, the coefficient of place (k, l) in the matrix DF^TDF is non zero only for displacements $\mathbf{e}_k$ and $\mathbf{e}_l$ associated with vertices that are corners of one common rectangle. The matrix DF^TDF has thus a sparse structure, and it can be assembled like a finite-element matrix, as we explain now.

$$\begin{aligned} DF^TDF &= \sum_{k,l} \left(DF^TDF\right)_{k,l} \mathbf{e}_k \otimes \mathbf{e}_l \\ &= \sum_{k,l} \int_\Omega (\nabla I_1(x') | \mathbf{e}_k(x))(\nabla I_1(x') | \mathbf{e}_l(x)) \mathbf{e}_k \otimes \mathbf{e}_l \, dx \\ &= \sum_{k,l} \sum_{R\in\mathcal{R}_q} \int_R (\nabla I_1(x') | \mathbf{e}_k(x))(\nabla I_1(x') | \mathbf{e}_l(x)) \mathbf{e}_k \otimes \mathbf{e}_l dx \\ &= \sum_{R\in\mathcal{R}_q} \sum_{k,l} \int_R (\nabla I_1(x') | \mathbf{e}_k(x))(\nabla I_1(x') | \mathbf{e}_l(x)) \mathbf{e}_k \otimes \mathbf{e}_l dx \end{aligned}$$

where we write $x' = x + \mathbf{V}(x)$ for the sake of concision. For a given rectangle $R \in \mathcal{R}_q$, the set of $\mathbf{e}_k$, $\mathbf{e}_l$ to be considered are the elementary vector fields attached to the 4 corners of the rectangle R. There are 8 such vector fields (one in each direction for each of the 4 corners), and 8 quantities of the form $(\nabla I_1(x + \mathbf{V}(x)) \mid \mathbf{e}_k(x))$ must be computed. A loop is performed over the rectangles, at each step 64 coefficients of the matrix DF^TDF are updated, but for symmetry reasons, only 36 different quantities are evaluated and they are straightforward to compute once $\nabla I_1(x + \mathbf{V}(x))$ is known.

The vector field $DF^TF \in \mathcal{C}_q$ is assembled rapidly in a similar way; we do not give the details here. The matrix DF^TDF is sparse, and it is assembled by reading once the data, using a finite-element loop over the squares between the control points. The right-hand side vector DF^TF is also assembled rapidly. The precise justifications and formulas are provided in Fehrenbach and Masmoudi (2008). The term S^TS associated with the spatial regularization parameter is easy to compute. Finally, we consider a conjugate-gradient method without preconditioning for updating the solution in the optimization process.

The algorithm that is used to obtain the optimal vector field in one space $\mathcal{C}_q$ is the following:

input: *images* I_0, I_1, *initial guess for the displacement* $\mathbf{V}_0$

1. set $k := 0$,
2. assemble the matrix $DF^T(\mathbf{V}_k)DF(\mathbf{V}_k)$ and the vector $DF^T(\mathbf{V}_k)F(\mathbf{V}_k)$,
3. solve the equation $DF^TDF\mathbf{d}_k = -DF^TF$ for $\mathbf{d}_k$, using conjugate gradient without preconditioning,
4. make a line search to find $\alpha_k \in [0, 10]$ that minimises the cost function $j(\mathbf{V}_k + \alpha_k\mathbf{d}_k)$,
4. set $\mathbf{V}_{k+1} := \mathbf{V}_k + \alpha_k\mathbf{d}_k$, set $k := k + 1$,
5. if the stopping criterion is not met, go back to step 2.

Note that it is usually better to use a constant (null) displacement field as initial guess $\mathbf{V}_0$ for the optimization problem.

2.4 A simple synthetic example

The nonlinear term defined by Eq. 2 allows to recover large displacements; this contrasts with optical flow methods where the Taylor series of Eq. 2 is truncated at the first term. An illustration of this fact is provided by the following synthetic example, where a multigrid estimation of the displacement allows to register two rectangles. A general purpose optical flow code (Black and Anandan 1993, 1996) was not able to register the images, when provided standard parameters.

We have applied our method to coarse versions of the images, that were successively refined. The spatial regularization term is of the form $\alpha R_0 + \alpha R_1$. In Fig. 2 we present the initial (I_0) and the final (I_1) images, the displacement field that was identified (for the sake of clarity, the length of the vectors was divided by a factor 5) and the difference of the images after registration (i.e. the residual error).

2.5 Quality estimate

An estimation of the quality of our results is highly motivated by the application we presented in introduction, namely data assimilation. A well-known issue and a crucial point in data assimilation are the knowledge of the statistics of observation errors. Hence, we propose here an estimation of the quality of the pseudo-observations identified by our algorithm. In order to assess the quality of the retrieved motion, a first idea could be to measure the difference of gray level after registration.

This simple idea can be improved by observing the type of images that we propose to process: some large zones have a constant gray level. In these zones, any displacement (that keeps the points inside the zone) would match the gray levels before and after the registration. In fact, the displacement estimates in these zones depend strongly on the regularization term. The fact that the gray levels are matched after registration does not provide relevant information.

For this reason, we propose a normalized quality estimate, where the quality of the motion depends on the ratio between the gray-level differences before and after registration:

$$e(I_0, I_1; u, v)(x, y) = 1 - \frac{|I_1(x + u(x, y); y + v(x, y)) - I_0(x, y)|}{|I_1(x, y) - I_0(x, y)|}, \tag{14}$$

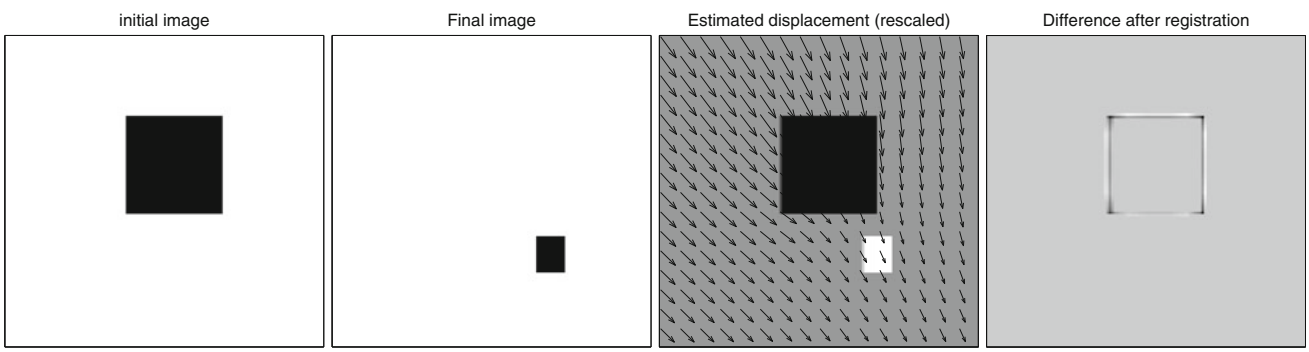

Fig. 2 Synthetic example: *from left to right*, initial image I_0, final image I_1, estimated displacement where the norm of the vectors was divided by a factor of 5, and difference of the images after registration

if the denominator is nonzero, otherwise we define $e(I_0, I_1;u, v) = 0$.

We can clearly see that if the two images were quite different on a pixel (x, y) before the process, and much less different after, then the estimate e is nearly equal to 1. We will further see that in some regions of the images, there is almost no signal, and then the two images are equal, both before and after the identification process. This leads to an estimate e equal to 0, not because the identified velocity is wrong, but because we cannot quantify whether it is good or not. This estimator is provided by our algorithm, so that it can be used along with the identified velocity fields in data assimilation experiments.

3 Numerical results on simulated data

3.1 Description of the shallow water model

The shallow water model (or Saint-Venant's equations) is a basic model, representing quite well the temporal evolution of geophysical flows. This model is usually considered for simple numerical experiments in oceanography, meteorology or hydrology. The shallow water equations are a set of three equations, describing the evolution of a two-dimensional horizontal flow. These equations are derived from a vertical integration of the three-dimensional fields, assuming the hydrostatic approximation, i.e. neglecting the vertical acceleration. There are several ways to write the shallow water equations, considering either the geopotential or height or pressure variables. We consider here the following configuration:

$$\begin{aligned} &\partial_t u - (f+\zeta)v + \partial_x B = -ru + v\Delta u, \\ &\partial_t v + (f+\zeta)u + \partial_y B = -rv + v\Delta v, \\ &\partial_t h + \partial_x(hu) + \partial_y(hv) = 0, \end{aligned} \tag{15}$$

where the unknowns are u and v the horizontal components of the velocity, and h the geopotential height (Blayo et al. 2003). The initial condition $(u(0), v(0), h(0))$ and no-slip lateral boundary conditions complete the system. The other parameters are the following: $\zeta = \partial_x v - \partial_y u$ is the relative vorticity, $B = g^* h + \frac{1}{2}(u^2 + v^2)$ is the Bernoulli potential, $g^* = 0.02$ m s^{-2} is the reduced gravity, $f = f_0 + \beta y$ is the Coriolis parameter (in the β-plane approximation), with $f_0 = 0.25$ s^{-1} and $\beta = 0.0406$ m^{-1} s^{-1}, $r = 9 \times 10^{-8}$ s^{-1} is the friction coefficient, and $v = 10^{-11}$ m^2 s^{-1} is the viscosity (or dissipation) coefficient. There is no forcing term in the model, e.g. corresponding to the wind stress, in our applications.

We consider a numerical configuration in which the domain is a square of 3×3 square meters, with no-slip boundary conditions, and the length of the time period is 500 seconds. The time step is 10^{-2} second, i.e. there are 50,000 time steps, and the spatial resolution is 6 mm, i.e. there are 500×500 grid points. From a physical point of view, such a configuration is equivalent to a much larger configuration (e.g. a $2{,}000 \times 2{,}000$ km square domain, and a time period of several days). This model has been developed by the MOISE research team of INRIA Rhône-Alpes (Blayo et al. 2003).

We briefly describe the numerical schemes used for the resolution of Eq. 15 and we refer to Durbiano (2001) for more details. We consider a leap-frog method for time discretization of Eq. 15, controlled by an Asselin time filter (Asselin 1972). The equations are then discretized on an Arakawa C grid (Arakawa and Lamb 1977), with $N \times N$ points ($N = 501$ in our experiments): the velocity components u and v are defined at the center of the edges, and the height is defined at the center of the grid cells. Then, the vorticity and Bernoulli potential are computed at the nodes and center of the cells, respectively.

This model is coupled with an advection-diffusion equation:

$$\partial_t c + u\partial_x c + v\partial_y c = 0, \tag{16}$$

where c is the concentration of a passive tracer (e.g. chlorophyll in oceans), and (u, v) is the fluid velocity. We also add to this equation an initial condition $c(t = 0)$. Equation 16 is solved by a conservative finite volume scheme.

We consider then a trajectory of this shallow water model coupled with a concentration equation, from which a concentration image is extracted every 100 time steps. Figure 3 shows four such concentration images, corresponding to the initial condition (first time step), and three intermediate states (10,001, 25,001 and 40,001*st* time steps, respectively, corresponding to images number 101, 251 and 401, respectively). The initialization of the concentration is shown on the first image of Fig. 3. The initial velocity field is a counterclockwise rotating vortex, centered at the location of the initial concentration.

3.2 Identification process

Two consecutive images are extracted from these simulated data, corresponding to images number 251 and 252 (or time steps 25,001 and 25,101). We recall that one image is obtained from this model every 100 time steps (i.e. every second). We applied our algorithm to these two images, with the aim of identifying the entire velocity field. In this experiment, the regularization norm is R_1, defined by Eq. 6. We will discuss the choice of the regularization below.

Figure 4 shows on the left the identified velocity field. This result has been obtained without any multiscale

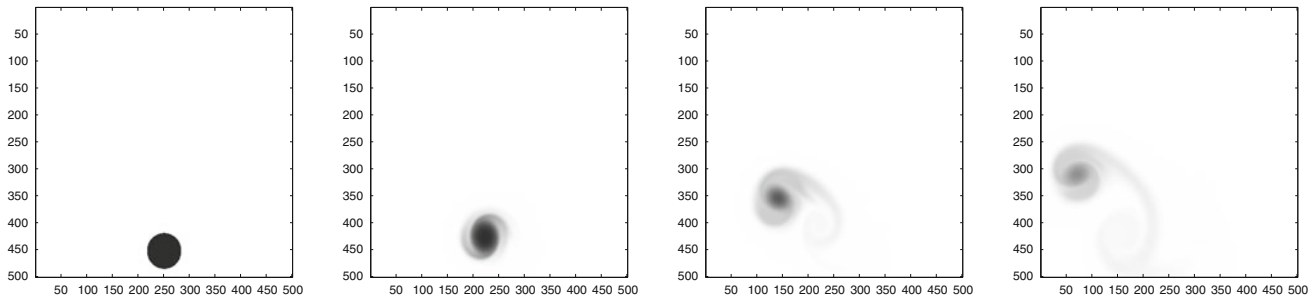

Fig. 3 Concentration images extracted from a simulation of a shallow water model coupled with a concentration equation, images number 1, 101, 251 and 401 (from left to right), respectively

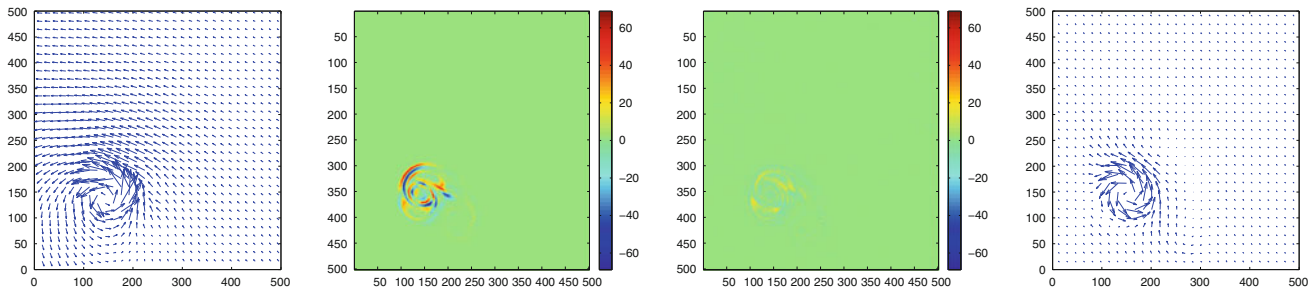

Fig. 4 Identified velocity between images I_0 and I_1 (*left*); Difference between image I_1 and image I_0 transported by: a constant null velocity field (initialization of the algorithm) (*center-left*), and the identified velocity field after convergence of the algorithm (*center-right*); Ground truth velocity field (*right*)

approach, and without any a priori estimation of the velocity (i.e. the algorithm was initialized with $u = 0$ and $v = 0$). A counterclockwise rotating vortex can clearly be identified on this figure. Moreover, the mean of the longitudinal component of the velocity is slightly negative and the mean of the transversal component is slightly positive. We have then identified a global displacement of the vortex to the top left of the domain.

We can first quantify the identification of the velocity field, by having a look at the difference between the second image I_1 and the first image I_0 transported by the velocity. Figure 4 shows on the two center pictures this difference at the beginning of the algorithm, when we initialize with a constant null velocity field, and at the end of the algorithm, with the identified field. It is clear that the first image has been well transported to the second one, and the reconstructed velocity field is reliable.

The algorithm converges in a few tens of iterations, and we do not need any a priori estimation of the velocity field. The computation time is nearly 1 h (one iteration costs 3 min), with Matlab(R) on a 2.0 GHz laptop.

Finally, these results are compared with the ground truth displacement field. As data are extracted from a known model trajectory, it is possible to integrate these true instantaneous velocities in order to obtain the corresponding Lagrangian velocity field between two time steps. Figure 4 shows on the right the ground truth displacement field at time step 25,001. As explained in the introduction, our algorithm extracts Lagrangian displacement fields, in order to register the images, and we indeed need to compare them with Lagrangian fields. Note that the scale is the same on both figures.

We can see that the shape of the vortex is pretty well identified. Also, the length of the vectors is almost the same in the vortex zone. The main differences between the two figures are located outside the vortex, where there is almost no information on the images. In order to better visualize what our algorithm does identify, we decided to show the identified velocity with colored vectors, where the color is defined by the quality estimate. In the framework of data assimilation, it is important to consider only reliable data.

In Fig. 5, we can compare the ground truth displacement field and the identified velocity field, filtered by the quality estimate, i.e. the color of the vectors is a function of the quality estimate, from blue to red for reliable to unreliable velocities. One should only consider the blue vectors, where the identified velocity is considered to be reliable (quality estimate greater than 66%, see Sect. 2.5 for the definition of the estimator). Note that most of the identified vectors are not reliable (in red, where the quality estimate is smaller than 33%), which is reasonable as there is absolutely no signal in the images. There is indeed no physical information about how the velocity field looks like outside the vortex (remind that we cannot use null

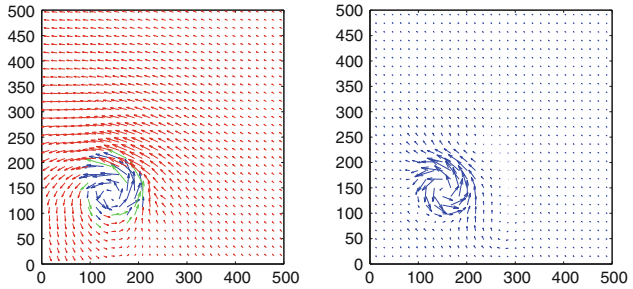

Fig. 5 Identified and ground truth velocities between images I_0 and I_1; the identified velocity is shown in *blue* (resp. *green*, resp. *red*) if the quality estimate is larger than 66% (resp. between 33 and 66%, resp. smaller than 33%)

divergence as we only look at a 2D projection of a 3D "null divergence" phenomenon). The reliable identified vectors are pretty well estimated, in comparison with the ground truth. The mean amplitude is almost the same (less than 10% of difference on the vortex zone).

We have seen that the apparent velocity is very well identified, as the first image is well transported toward the second one, and it is close to the ground truth displacement field where there is some information in the images.

In a more general framework, several issues may lead to a significative difference between the apparent velocity and the ground truth displacement field, for example, the tracer transport modeling. Equation 16 may not reproduce very well the transport phenomenon. Also, this phenomenon is well known in wave propagation problems, as one can have a null apparent velocity while the true velocity is not equal to zero, and we refer to this domain for a more detailed comprehension of this problem (see e.g. Lillie (1999)).

3.3 Choice of the regularization norm

We now study the impact of the regularization norm on the results. We will first consider the final value of the cost function as a criterion for the quality of the identified velocity field, as it measures exactly the difference between the second image and the first one transported to the second one. We are indeed interested in finding a velocity field that maps one image onto the second one, and the smaller the cost function is, the better the mapping is. But we will also look at the identified velocity fields and compare them with the ground truth displacement field. We initialize all minimization processes with $u = 0$ and $v = 0$, and we only show the 10 first iterations as all minimizations converge in at most 10 iterations.

Figure 7 shows the shape of the identified velocity around the vortex and Fig. 6 shows the evolution of the cost function versus the number of iterations of the minimization process, for several regularization terms. On Fig. 7, the color of the vectors is a function of the quality

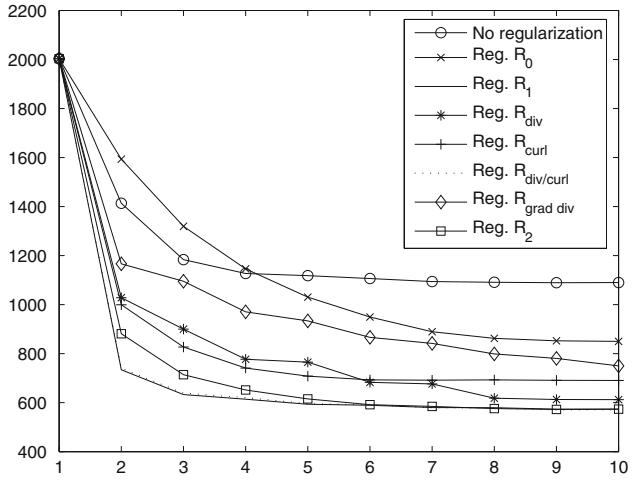

Fig. 6 Evolution of the cost function versus the number of iterations of the minimization process

estimate (see Sect. 2.5): blue if it is larger than 66% (reliable velocity), green between 33 and 66%, and red below 33%. The last picture shows the ground truth on the corresponding zone. All figures use the same scale. On Fig. 6, in order to make an unbiased comparison, only the difference between the two images (image I_1 and image I_0 transported by the velocity field) is shown, and not the regularization part. See Eqs. 5–12 for the definition of the various norms. Table 1 gives the corresponding values of the regularization coefficient α (see Eq. 4, and end of Sect. 2.2). Finally, Table 2 gives the root mean squared error of the identified velocity (in comparison with the ground truth), restricted to the *reliable* vectors, i.e. for which the computed quality estimate is larger than 66%. We do not take into account the identified velocity vectors for which the algorithm gives a low-quality estimate.

First, without any regularization term, the cost function does not decrease very much, and the identified velocity field is not satisfactory, as it is too noisy. However, the global shape of the vortex is pretty well identified. Similar results have been obtained with R_0, with slightly smaller identified velocities (as the regularization forces small velocities). The R_1 regularization provides the best results, from all points of view: fastest decrease of the cost function, shape of the identified velocity, global structure of the velocity (translation to the top left, and counterclockwise rotating vortex). The identified field is smooth, as imposed by the regularization. Then, regularization R_2 has comparable results: the decrease in the cost function is a little bit slower, but the final value is the same. The velocity field is slightly smoother. Note also that far away from the vortex, the velocity is small, as only the gradient of the field (and not the field itself) has to be small. However, from a mathematical point of view, shallow water solutions may not have enough regularity for considering second-order

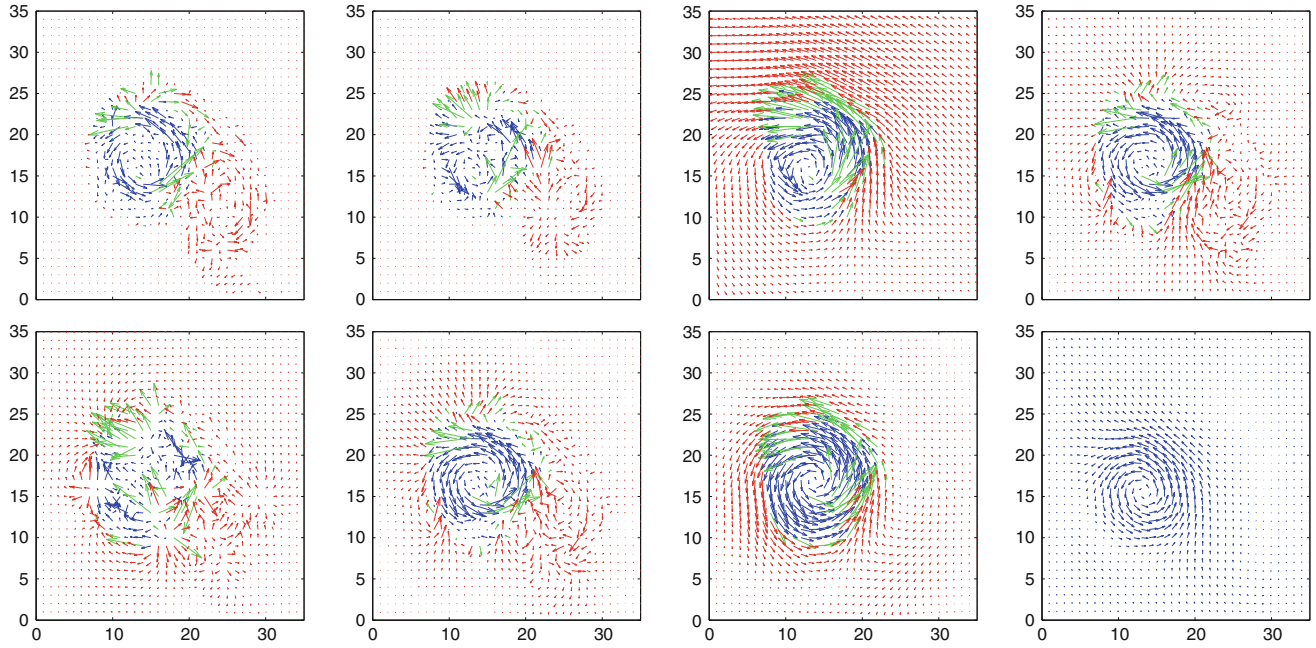

Fig. 7 Zoom of the identified velocity fields between images I_0 and I_1, using various regularizations, namely from *left to right* no regularization, R_0, R_1, R_{div}, R_{curl}, $R_{\nabla}div$, and R_2, where the color is a function of the quality estimate: *blue* if it is larger than 66%, *green* between 33 and 66%, and *red* below 33%; *Bottom right* ground truth displacement field on the same zone

Table 1 Values of the regularization coefficient α, see Eq. 4, for the different regularization norms

Norm	R_0	R_1	R_{div}	R_{curl}	$R_{div/curl}$	$R_{\nabla}div$	R_2
α	5×10^3	5×10^3	5×10^4	5×10^4	5×10^3	5×10^2	5×10^2

Table 2 Root mean squared error between the identified velocity and the ground truth displacement field, restricted to the vectors for which the quality estimate is larger than 66%

Norm	ϕ	R_0	R_1	R_{div}	R_{curl}	$R_{\nabla}div$	R_2
RMS error	1.25	0.67	0.36	0.71	1.63	1.01	0.38

regularizations. Moreover, the R_2 norm does not identify the global displacement of the vortex (translation to the top left). Even though second-order regularizations generally give accurate results (Corpetti et al. 2006; Yuan et al. 2007), its relative inability to identify the global displacement of the fluid flow in comparison with the first-order regularization is not satisfactory. We do not represent the solution corresponding to $R_{\nabla}div/\nabla curl$, as the optimization of the weight parameters between the two terms led to a norm equivalent to R_2.

Some regularizations produce clearly bad fields, from a physical point of view. For example, the R_{curl} regularization has absolutely no interest, as the velocity field cannot have a small curl in the experiments considered here. But it is interesting to have a look at the identified field, which is almost curl-free. The weak performance of this regularization is confirmed by the RMS error given on Table 2. We can also see that the solutions identified with the R_0 or $R_{\nabla}div$ regularizations, or without any regularization, are not very good, concerning both the decrease in the cost function and the identified field. Finally, the most physical regularization is probably R_{div}, as we expect a null divergence velocity field in geophysical flows. But the decrease in the cost function is not as good as some other regularizations, and the identified field has several false secondary recirculation vortices that are imposed by the transport constraint between the two images. Moreover, considering that the images are acquired every 100 time steps only, the velocity we want to identify between these two images is a time Lagrangian integration of many instantaneous velocities, and it cannot have a divergence equal to zero (e.g. in the case of a perfect rotating vortex, by integrating many curl fields, one can obtain a curl-free field).

Also, the R_1 norm leads to the smallest RMS difference with the ground truth displacement field. For these reasons, we decided to show only the R_1 regularization in all the following experiments.

3.4 Multiscale approach

We present here a multiscale approach of our algorithm. We still initialize with $u = 0$ and $v = 0$, and we now look for a piecewise affine velocity field. We first work on a coarse grid, every 16×16 pixels, and identify a velocity

field between the two images I_0 and I_1 which is piecewise affine every 16 × 16 pixels, i.e. the cost function J Eq. 4 is minimized on the set $\mathcal{C}_{16}$ (see Sect. 2.3). This field is then provided as an initial guess to the minimization of the same cost function on the set $\mathcal{C}_8$, in order to identify a 8 × 8 piecewise affine velocity field, and so on. For each refinement of the mesh, we use the previous identified field (on a coarse grid) as an initial guess for the optimization process on the finer grid. At the end of the process, we obtain a field on the finest mesh, i.e. an estimation of the velocity at each pixel of the image. We consider the gradient of the cost function as a stopping criterion for each optimization.

Figure 8 shows a zoom (on the vortex zone) of the identified velocity field, at every refinement step from a 16 × 16 pixels piecewise affine to a 4 × 4 pixel piecewise affine field. The multiscale approach works perfectly, as the identified solution on a coarse grid provides a very good estimation of the solution on a finer grid, and the 16 × 16 pixels piecewise affine field is already a good approximation of the final solution.

The computation time is now 6 min, which should be compared with the computation time of one iteration on the finest mesh (3 min). In the same time, without any multiscale approach, we would be able to perform only 2 minimization iterations, and the solution would not be good at all. Moreover, on a nonmultiscale approach, the final solution may not be realistic if there is no a priori information.

For the further experiments, we will only consider this multiscale approach, as it provided almost the best results, in a very short time.

3.5 Object tracking

We now present a simple extension of the identification process to the 3D case (2D in space + time). Assume that we have a particular feature or object in the first image. In our case, we can identify one specific vortex. We can then limit the identification process to a region around this object. This region is propagated from one pair of images to the next one by the mean of the identified velocity. This allows one to track this object in time. As mentioned in the introduction, this experiment is justified by a standard meteorological application, in which the goal is to track automatically one specific hurricane or storm (that may have been manually identified), with the aim of predicting its future trajectory.

Figure 9 shows the result of our tracking algorithm. We still consider the same starting image, in which we have defined a region of interest (represented by the red box) of size 130 × 130. We have considered 100 successive images, from time step 25,001 to 35,001 (i.e. images number 251 and 351), and for each pair of consecutive images, we have identified a velocity field in the region of interest only, simply by applying our previous identification algorithm to the selected region instead of the entire image. Then, the red box is propagated from one time step to the next one by the mean of the identified velocity. Figure 9 shows the tracking result after 50 and 100 images. The center of the vortex is still in the middle of the red box, even after 100 images, i.e. 10,000 time steps of the numerical model. This result confirms that between two images, the apparent velocity is very well retrieved, and hence the displacement of the object is also well identified.

The computation time of this entire simulation (100 images) is less than 4 min. This is due to the drastic reduction of the problem size, as we identify the velocity in a small region of the image, and each velocity identification nearly costs 2 s.

However, in some experiments where the selected region is too small, the tracking is not so good, there can be a 10 to 20% drift between the tracked region and the object of interest after 100 images. This is mainly due to a too small region, and the quality of the identified field is degraded, and also the mean of the identified field in this

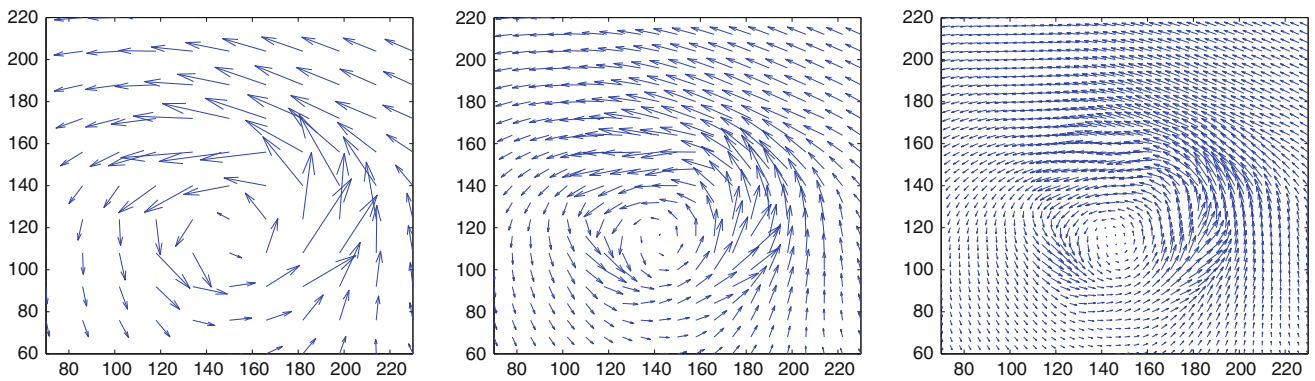

Fig. 8 Zoom of the identified velocity field between images I_0 and I_1 in a multiscale approach; 16 × 16 pixels, 8 × 8 pixels and 4 × 4 pixels piecewise affine fields, respectively

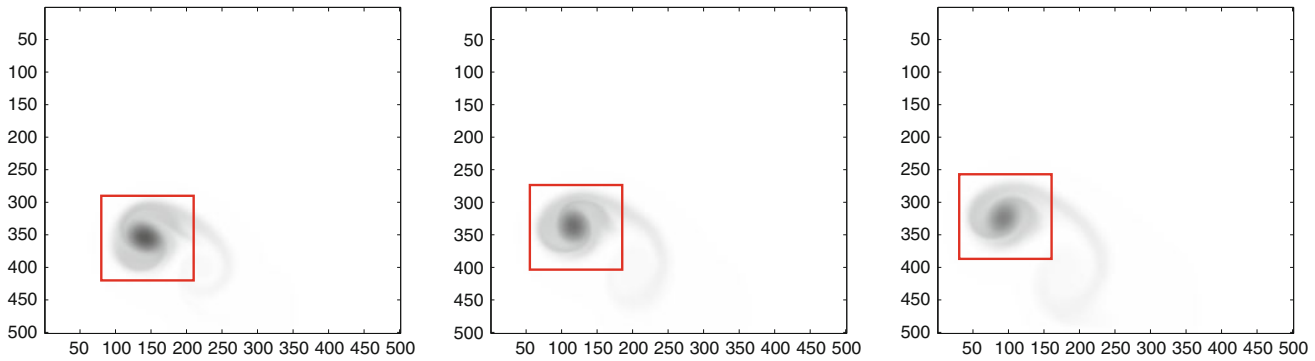

Fig. 9 Tracking of a specific region: initial image (number 251) and region of interest in *red* (*left*); intermediate image (number 301) and propagated region of interest, using the identified velocity at each time step (*center*); final image (number 351) and propagated region (*right*)

small region does not give a good estimation of the global displacement. The best way to track the object is to consider a reasonably large zone, e.g. not much smaller than in the example in Fig. 9. In such a situation, we did not observe any drift, even after 100 images.

4 Numerical results on experimental data

4.1 First experiment: colorant in water

We now consider data extracted from several experiments on the Coriolis rotating platform (Coriolis 2010). A large rotating turntable (diameter: 13 m) allows us to reproduce the oceanic or atmospheric flows. Some colorant is inserted in the water as the platform rotates, and among the various measurement devices, a camera takes pictures of the experiment (Flor and Eames 2002).

We consider two consecutive images extracted from these experimental data, see Fig. 10 (left and center). The multiscale approach of our algorithm is used (see previous sections), looking first for a piecewise affine velocity field every 16 × 16 pixels, and then refining 4 times the mesh. For each intermediate mesh, the minimization process is stopped when the decreasing rate of the cost function is lower than a threshold.

Figure 10 shows on the right the identified velocity field, represented every 12 pixels for visualization reasons. The total computation time is smaller than 5 min. Note that, unlike the previous simulated experiments, the transversal and longitudinal components of the velocity do not have a zero (or nearly) mean. The global structure of the displacement is clearly a rotating (counterclockwise) vortex in a translation field to the top (and a little bit left) of the domain. Note that using the same computation time as the multiscale approach, i.e. 5 min, we were able to perform only 3 iterations on the finest grid, and starting from a null initial velocity field, the final result is not satisfactory (the identified velocity does not show a rotating vortex).

In order to visualize the quality of the reconstruction, Fig. 11 (left) shows the difference between image I_1 and image I_0. Fig. 11 (center) shows the difference between I_0 and I_1 transported by the velocity field. The identified velocity transports very well the first image toward the second one. We can conclude that our identification algorithm of the apparent velocity works well on these experimental data. The quality estimate of the identified velocity is given on the right picture, from 0 (for unreliable data) to

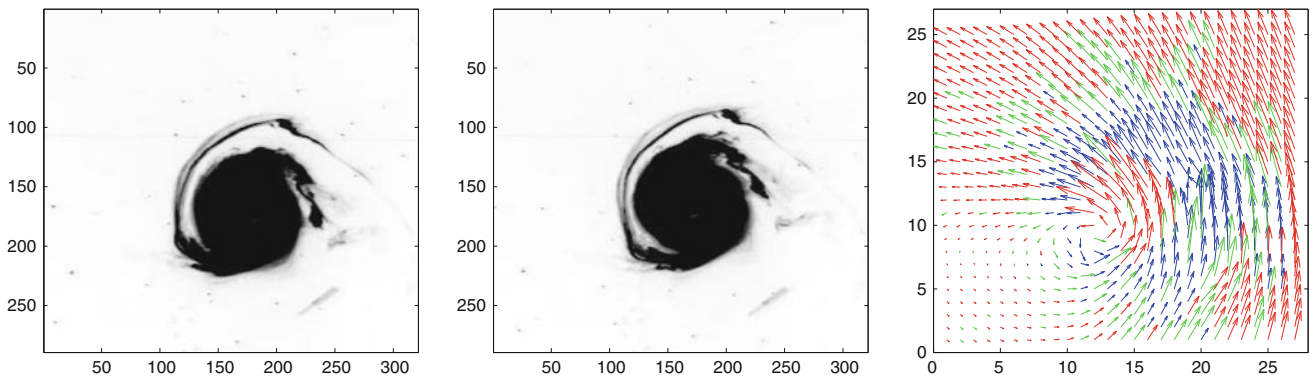

Fig. 10 Concentration images I_0 and I_1 at time steps 5 and 6 respectively (*left and center*); Identified velocity field between images I_0 and I_1 (*right*)

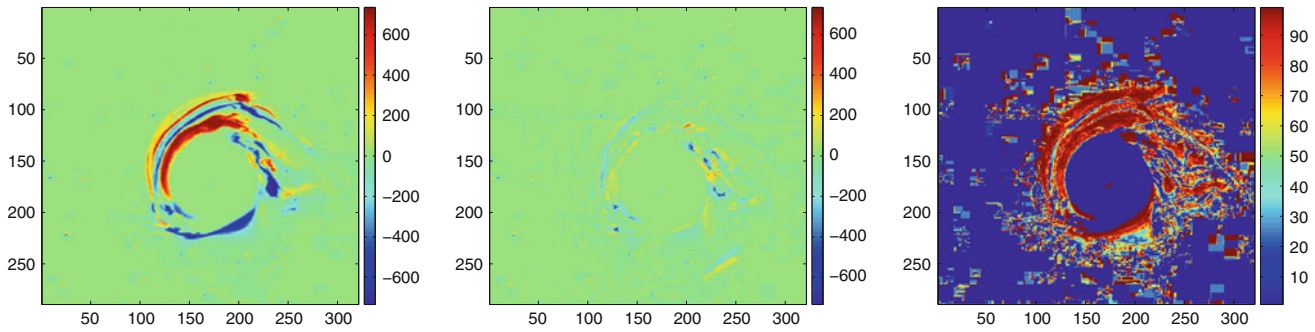

Fig. 11 Difference between image I_1 and image I_0 transported by: a constant null velocity field (initialization of the algorithm) on the *left*, the identified velocity field (after convergence of the algorithm) on the *center*; Quality estimate of the identified velocity, from 0 to 100% of reliability, on the fine 1×1 grid (*right*)

100% (for a perfect identified velocity). The quality estimate is equal to 0 in most of the image. This is due to the lack of signal in both images (see Fig. 10). In the interesting part of the image, i.e. the vortex, there is some difference between the two original images, and hence the estimate can be computed. We can see that it is very high, globally larger than 90%. This was predictable from Fig. 11 (center), in which we clearly see the good agreement between the transported image I_0 and image I_1.

4.2 Comparison with PIV

As we have no ground truth displacement field in this real experiment, we also compared our results with the PIV (particle imaging velocimetry) software used in the Coriolis platform (Coriolis 2010) on more realistic images. Figure 12 shows on the left such an image. Note that images are acquired on the Coriolis platform in such a way that they show both colorant and particles evolutions. Then, the same figure shows on the second image the identified velocity field by our approach. Finally, this figure shows on a zoom (in the vortex zone) a comparison between our approach and the PIV method on the right side of Fig. 12. On both images, the red (resp. green, resp. blue) vectors correspond to correlations (or quality estimate, for our approach) lower than 33% (resp. between 33 and 66%, resp. higher than 66%). The shape and global structure of the identified velocity fields are similar except in the middle of the vortex, where the PIV does not identify correct physical displacements. The velocities with good correlation (in blue) are very similar with ours, both in direction and in amplitude. Our algorithm has two advantages in comparison with the PIV method: first, it identifies some smaller structures that the PIV does not see (this is mainly due to a very low correlation, leading to very small identified velocities out of the vortex). Second, PIV methods can drastically reduce the resolution of the results, as in this experiment we could not consider smaller zones than 15×15 pixels for cross-correlation. Note also that the PIV results correspond to filtered velocities, obtained after several runs of the PIV operational software on the Coriolis platform (see Coriolis (2010) website).

Figure 13 shows another example of identified velocity field by our approach, on another image of a similar experiment performed on the Coriolis platform (still with both particles and colorant in water). We multiplied by three the length of the vectors for a better visualization. Then, for both identification methods (Coriolis' PIV and ours), we computed the relative vorticity from the velocity field. Figure 13 shows on the right side the identified

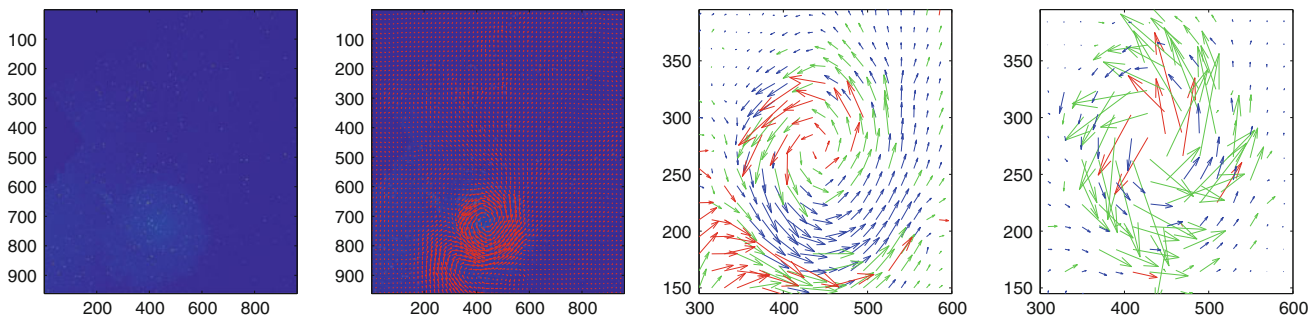

Fig. 12 From *left to right* one of the two images used for the identification process; identified velocity by our approach represented on the image; zoom on the vortex of the identified velocities by our approach (*left*) and PIV (*right*)

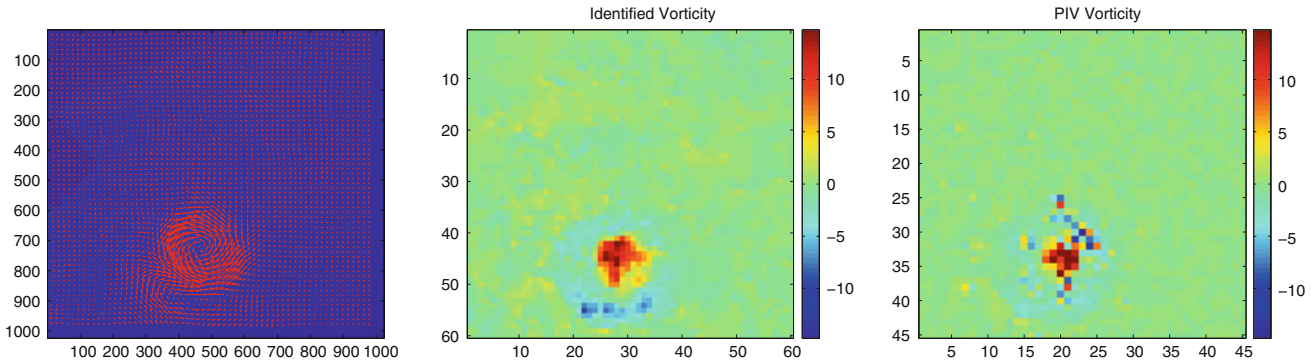

Fig. 13 From *left to right* identified velocity by our approach represented on another experimental image (*left*); identified vorticity field by our approach (*center*) and PIV (*right*)

vorticity fields by our approach and by PIV, respectively, using the same scale. We can see that the vorticity provided by our approach is much less noisy, the main vortex is quite nicely identified, and also, the secondary vortices are clearly visible, contrary to the PIV vorticity. This point is noticeable, as in a data assimilation framework, it is crucial to identify the main vortices and their location. In all comparisons that we did, our method was better on both qualitative and quantitative points of view. The improvements are mainly on the identification of secondary vortices, and on the identification of the velocity in the center of the vortices.

4.3 Second experiment: particles in water

We now consider a second experiment performed on the Coriolis rotating platform, in which some particles replace the colorant.

Figure 14 shows two images extracted from this other experimental movie. The structure of the velocity is the same as in the previous subsection: a vortex in a global translation displacement. We used the same multiscale approach as previously.

Figure 15 shows the identified velocity components and field, using this multiscale approach. The velocity field is represented every 12 pixels for visualization reasons. The

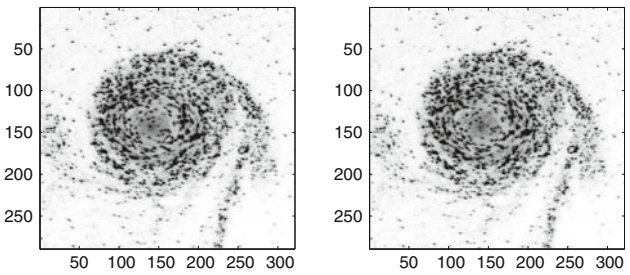

Fig. 14 Concentration images I_0 and I_1 at time steps 28 and 29, respectively

identified velocity is quite smooth, even though the regularization coefficient is the same as in the previous experimental results. Once again, it is very easy to see the global structure of the velocity field, with a characteristic counterclockwise rotating vortex, nearly located at the center of the image. The global displacement of the structure is a small translation to the top, as the mean longitudinal component is nearly zero whereas the mean transversal velocity is slightly positive. Note that almost all the vortex zone is considered as reliable.

Figure 16 shows the results of the identification process. On the left is represented the difference between the two original images, and on the center, the difference between the second image and the first one being transported by the identified velocity. Even if the original difference looks quite big, with a lot of small displacements everywhere, one can see that the smooth identified velocity provides a very good transport between these two images, as the difference has been drastically reduced.

Figure 16 shows on the right the quality estimate from Eq. 14 corresponding to the identified velocity, on the 4×4 pixels coarse grid. The scale goes from 0 for a totally untrustable velocity to 100% for a fully reliable velocity. One should notice that outside the vortex, the quality is poor. This is due to the fact that there is more or less nothing in both images, and then there is no way to assess that we have identified the right velocity. For example, in the bottom right corner, both images have a signal equal to 0, and then any reasonable velocity can transport a zone of 0 to another one. On the contrary, in the vortex zone, the quality is much better and is globally higher than 70%. This is due to the high signal in both images and also to the identification of an efficient transport field between the two images in this zone.

We also compared our algorithm with the operational PIV method of the Coriolis platform (see Sect. 4.2), on another set of images with particles in water. This experiment is similar to the one presented in Sect. 4.2, as there

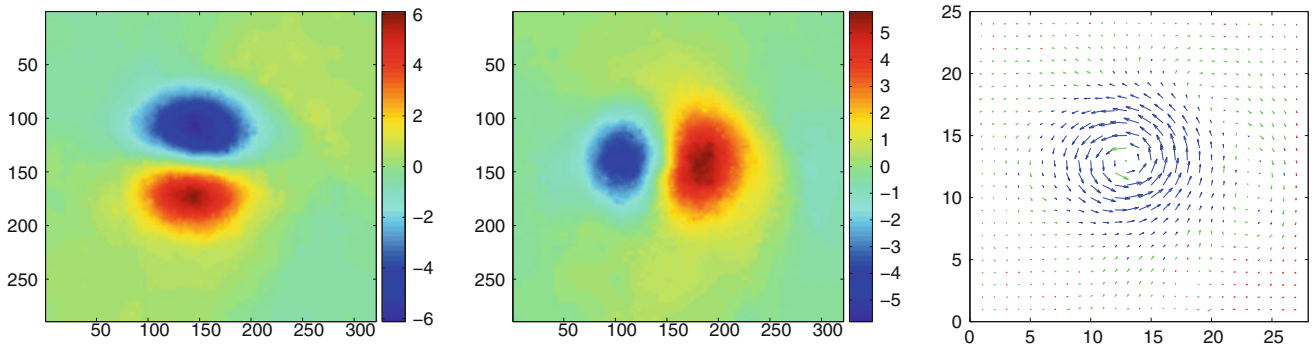

Fig. 15 Identified velocity between images I_0 and I_1: longitudinal and transversal components on the *left*, velocity field on the *right*

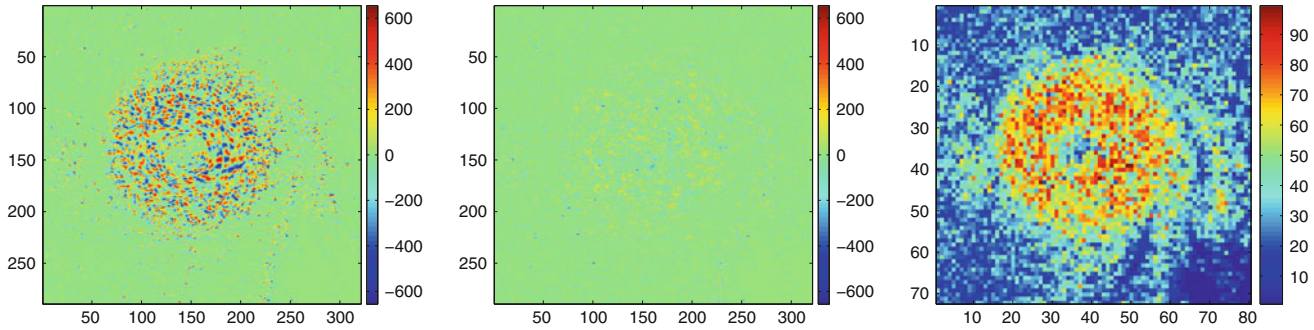

Fig. 16 Difference between image I_1 and image I_0 transported by: a constant null velocity field (initialization of the algorithm) (*left*), the identified velocity field (after convergence of the algorithm) (*center*); Quality estimate of the identified velocity, from 0 to 100% of reliability, on the 4 × 4 grid (*right*)

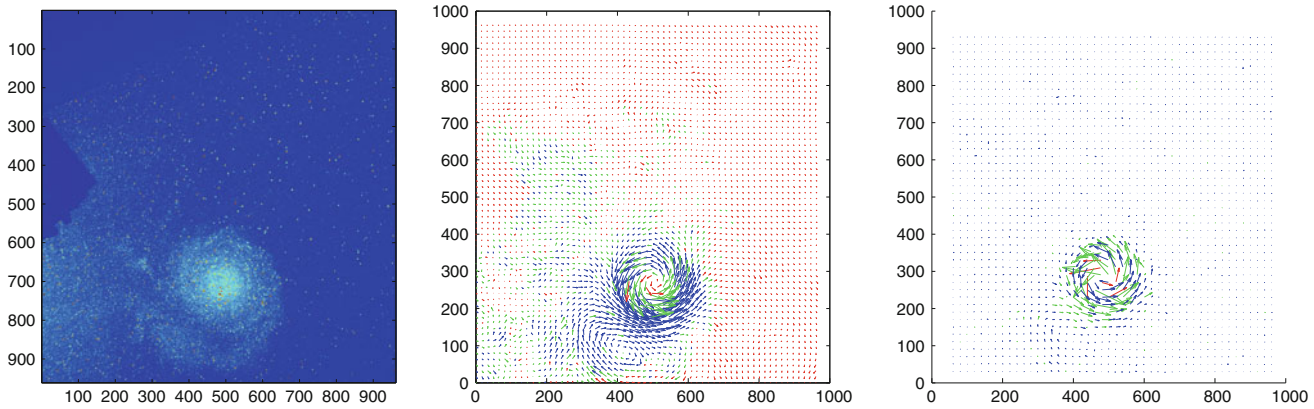

Fig. 17 From *left to right* one of the two particle images used for the identification process; identified velocity by our approach, and by the PIV method

are both colorant and particles inserted in the water flow in this experiment, and several images are acquired at the same time with different cameras in order to see mainly either the colorant or the particles.

Figure 17 shows one image of the sequence that we considered, and the identified displacement field by both our approach and the PIV method. We use the same color map as in Sect. 4.2: red/green/blue for correlations lower than 33/66/100%.

As for colorant images, our method identifies much more displacement vectors than the PIV, and moreover, we obtain much more accurate vectors in the main vortex. Also we can observe the presence of secondary vortices in the top left of the image, which are badly identified by the

PIV approach. Finally, note that the amplitude of the reliable vectors is almost the same on the two approaches.

5 Conclusion

In this paper we presented an algorithm to estimate the Lagrangian displacement between two geophysical images. This algorithm is based on the constant brightness assumption, but as shown, it also works when this assumption is not satisfied at all. A multiscale approach allows us to perform a minimization of the cost function in nested subspaces, the Jacobian matrix of the cost function being assembled rapidly at each scale using a finite-element method. The coarse estimation allows one to avoid local minima, while the fine scales give more precise details. Several regularization terms are discussed, and it appears that the L^2 norm of the gradient gives reliable results, both for the local (vortices) and for the more global (translations) displacements.

The results of this algorithm on simulated and real fluid flows are presented, and they are encouraging both from their computational efficiency and from the quality of the estimated motion, in comparison with PIV results on the Coriolis platform.

As explained in the introduction, the extracted velocity fields can be seen as pseudo-observations of the fluid velocity, and the next step will be to consider the assimilation of these Lagrangian data. The results should be compared with the assimilation of classical data (e.g. sea surface heights in the case of oceanographic systems).

Acknowledgments We greatly acknowledge the MOISE research project of INRIA Rhône-Alpes (France) and the Coriolis project of LEGI (Grenoble, France) for providing us the synthetic and real data, respectively. This work is partly done within the MOISE INRIA team and supported by the French National Research Agency (ANR ADDISA). The first author was member of Institut de Mathématiques de Toulouse (Université Paul Sabatier, France) when he contributed to this paper. The authors also thank the referees for their useful comments and perspectives.

References

Adrian R (1991) Particle imaging techniques for experimental fluid mechanics. Ann Rev Fluid Mech 23:261–304

Alvarez L, Castaño C, Garcia M, Krissian K, Mazorra L, Salgado A, Sanchez J (2007) A variational approach for 3D motion estimation of incompressible PIV flows, scale space and variational methods in computer vision, vol 15. Springer, Berlin, pp 837–847

Ananden P (1989) A computational framework and an algorithm for the measurement of visual motion. Int J Comput Vis 2:283–310

Arakawa A, Lamb V (1977) Computational design of the basic dynamical processes of the UCLA general circulation model, Methods in Computational Physics, vol 17. Academic Press, London, pp 174–267

Asselin R (1972) Frequency filter for time integrations. Mon Wea Rev 100:487–490

Beauchemin S, Barron J (1995) The computation of optical flow. ACM Comput Surv 27(3):433–467

Black MJ, Anandan P (1993) A framework for the robust estimation of optical flow. In: Proceedings of the Fourth International Conference on Computer Vision, ICCV-93. Berlin, Germany, pp 231–236

Black MJ, Anandan P (1996) The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. Comput Vis Image Underst 63(1):75–104

Blayo E, Durbiano S, Vidard PA, Le Dimet FX (2003) Reduced order strategies for variational data assimilation in oceanic models. Springer, Berlin

Bruhn A, Weickert J, Kohlberger T, Schnörr C (2006) A multigrid platform for real-time motion computation with discontinuity-preserving variational methods. Int J Comput Vis 70(3):257–277

Coriolis (2010) Coriolis rotating platform, website http://www.coriolis-legi.org/

Corpetti T, Mémin E, Pérez P (2002) Dense estimation of fluid flows. IEEE Trans Pattern Anal Mach Intel 24:365–380

Corpetti T, Heitz D, Arroyo G, Mémin E, Santa-Cruz A (2006) Fluid experimental flow estimation based on an optical-flow scheme. Exp Fluid 40(1):80–97

Cuzol A, Hellier P, Mémin E (2007) A low dimensional fluid motion estimator. Int J Comput Vis 75:329–349

Durbiano S (2001) Vecteurs caractéristiques de modèles océaniques pour la réduction d'ordre en assimilation de données. PhD thesis, University of Grenoble

Fehrenbach J, Masmoudi M (2008) A fast algorithm for image registration. C R Acad Sci Paris, Ser I 346:593–598

Fitzpatrick JM (1985) A method for calculating velocity in time dependent images based on the continuity equation. In: Proceedings of Conference Comparative Vision and Pattern Recognition, San Francisco, USA, pp 78–81

Fitzpatrick JM (1988) The existence of geometrical density-image transformations corresponding to object motion. Comput Vis Grap Imag Proc 44:155–174

Flor JB, Eames I (2002) Dynamics of monopolar vortices on the beta plane. J Fluid Mech 456:353–376

Hansen PC (1992) Analysis of discrete ill-posed problems by means of the l-curve. SIAM Rev 34:561–580

Heitz F, Pérez P, Bouthemy P (1994) Multiscale minimization of global energy functions in some visual recovery problems. CVGIP Imag Underst 59(1):125–134

Hellier P, Barillot C, Mémin E, Pérez P (1999) Medical image registration with robust multigrid techniques. In: Proceedings of Second International Conference on Medical Image Computing and Computer-Assisted Intervention, lecture notes in computer science, vol 1679, pp 680–687

Horn B, Schunk B (1981) Determining optical flow. Artif Intell 17:185–203

Huot E, Isambert T, Herlin I, Berroir JP, Korotaev G (2006) Data assimilation of satellite images within an oceanographic circulation model. In: Proceedings of International Conference on Acoustics, Speech, Signal Processing, Toulouse, France

Isambert T, Herlin I, Berroir JP (2007) Fast and stable vector spline method for fluid flow estimation. In: Proceedings of International Conference on Image Processing, San Antonio, USA, pp 505–508

Larsen R, Conradsen K, Ersboll BK (1998) Estimation of dense image flow fields in fluids. IEEE Trans Geosci Remote Sens 36(1):256–264

Lillie RL (1999) Whole earth geophysics: an introduction textbook for gelologists and geophysicists. Prentice Hall, NJ

Lucas B, Kanade T (1981) An iterative image registration technique with an application to stereo vision. In: Proceedings of Seventh International Joint Conference on Artificial Intelligence, Vancouver, Canada, pp 674–679

Ma J, Antoniadis A, Le Dimet FX (2006) Curvlets-based snake for multiscale detection and tracking of geophysical fluids. IEEE Trans Geosci Remote Sens 45(1):3626–3638

Mémin E, Perez P (1998) Optical flow estimation and object-based segmentation with robust techniques. IEEE Trans Imag Proc 7(5):703–719

Mémin E, Pérez P (2002) Hierarchical estimation and segmentation of dense motion fields. Int J Comput Vis 46(2):129–155

Michel Y, Bouttier F (2006) Automatic tracking of dry intrusions on satellite water vapour imagery and model output. Quart J Roy Meteor Soc 132:2257–2276

Papenberg N, Bruhn A, Brox T, Didas S, Weickert J (2006) Highly accurate optic flow computation with theoretically justified warping. Int J Comput Vis 67(2):141–158

Ruhnau P, Kohlberger T, Nobach H, Schnörr C (2005) Variational optical flow estimation for particle image velocimetry. Exp Fluids 38(1):21–32

Tikhonov AN, Arsenin VY (1977) Solution of Ill-posed Problems. Winston & Sons, Washington

Yuan J, Schnörr C, Mémin E (2007) Discrete orthogonal decomposition and variational fluid flow estimation. J Math Imag Vis 28(1):67–80