RESEARCH ARTICLE

Variational second order flow estimation for PIV sequences

L. Alvarez · C. A. Castaño · M. García · K. Krissian ·
L. Mazorra · A. Salgado · J. Sánchez

Abstract   We present in this paper a variational approach to accurately estimate simultaneously the velocity field and its derivatives directly from PIV image sequences. Our method differs from other techniques that have been presented in the literature in the fact that the energy minimization used to estimate the particles motion depends on a second order Taylor development of the flow. In this way, we are not only able to compute the motion vector field, but we also obtain an accurate estimation of their derivatives. Hence, we avoid the use of numerical schemes to compute the derivatives from the estimated flow that usually yield to numerical amplification of the inherent uncertainty on the estimated flow. The performance of our approach is illustrated with the estimation of the motion vector field and the vorticity on both synthetic and real PIV datasets.

L. Alvarez (✉) · C. A. Castaño · M. García · K. Krissian ·
L. Mazorra · A. Salgado · J. Sánchez
Departamento de Informática y Sistemas, Universidad de Las Palmas de Gran Canaria, Campus de Tafira s/n, 35017 Las Palmas de Gran Canaria, Spain
e-mail: lalvarez@dis.ulpgc.es

C. A. Castaño
e-mail: ccastano@dis.ulpgc.es

M. García
e-mail: mgarcia@dis.ulpgc.es

K. Krissian
e-mail: krissian@dis.ulpgc.es

L. Mazorra
e-mail: lmazorra@dis.ulpgc.es

A. Salgado
e-mail: asalgado@dis.ulpgc.es

J. Sánchez
e-mail: jsanchez@dis.ulpgc.es

1 Introduction

Particle image velocimetry (PIV) is a relatively new technique widely used to study the temporal evolution of all kinds of flows. From the initial work by Adrian (1988), where the theoretical framework for PIV was set, this technique has been a continuous research topic (Thomas et al. 2005). The basic idea behind this technique is to make visible fluid motion by adding small tracer particles to the fluid and infer the flow velocity field from the images of the position of these particles at two instances of time (Westerweel 1997). The interested reader is referred to the work of Keane and Adrian (1990), where the experimental design rules are detailed.

In the context of PIV research, the standard procedure to infer the flow velocity field from the image pairs is to compute the cross-correlation function between local interrogation windows of consecutive frames (Keane and Adrian 1992). In this way, the local maximum of that function determines the displacement of the pixel values. Hierarchical and iterative schemes have been developed to overcome the limitations of correlation techniques. In this sense, a lot of work has been done since the pioneering work of Soria et al. (1996) and it is still an interesting research topic Westerweel et al. (1997), Scarano (2002), Ruhnau et al. (2005b), and Nogueira et al. (2005).

Recently, variational motion estimation approaches widely used in computer vision or image processing domains have been proposed for PIV (Corpetti et al. 2002; Ruhnau et al. 2005b; Corpetti et al. 2006). Encouraging

results have also been obtained with these approaches, although some limitations have also to be overcome. For instance, variational methods are usually based on the assumption that light conditions do not change from frame to frame. However, real fluid images usually suffer from temporal intensity distortions, yielding to wrong results. In this sense, more robust methods with respect to light conditions have been proposed by Corpetti et al. (2002) based on an integrated version of the continuity equation of fluid mechanics.

In any case, either with correlation-based methods or variational approaches, there is still a certain level of noise inherent to the estimation method (Foucaut and Stanislas 2002). This noise may introduce a significant error on the flow components, but it is even more important for the computation of the derivatives of the motion field due to numerical error amplification. Since flow derivatives are a great source of information in fluid dynamics (they appear in the Navier–Stokes equations and important parameters such as divergence, vorticity, the strain rate tensor and dissipation rate), it is particularly important to obtain an accurate estimation of flow derivatives.

Usually the computation of the flow derivatives is a two steps procedure: First we compute a flow estimation using any method we want and second we compute flow derivatives using some kind of finite difference scheme. In this paper we present a variational method to compute simultaneously the flow and its derivatives directly from the image sequence. The proposed method is based in an energy minimization and a second order Taylor development of the flow.

We illustrate the behavior of our approach by the estimation of the out-of-plane vorticity, which is an interesting measure to detect and to study the evolution of vortices in the flow (Hunt 1987). Several approaches have been proposed in the literature to achieve this task. On the one hand, discrete differential operators based on different numerical schemes are directly applied to the estimated velocity vector field. In general, their accuracy depends on the initial uncertainty inherent to the vector field, the sampling rate, neighborhood correlation, etc. (Etebari and Vlachos 2005; Luff et al. 1999; Raffel et al. 1998; Kim and Lee 1996). A second technique was proposed by Fouras and Soria (1998) which consists on analytical derivation of the estimated vector field, once it has been fitted to a base of second order polynomials by means of a χ^2 fitting procedure. A third technique is presented by Raffel et al. (1998) and Abrahamson and Lonnes (1995) based on the computation on the circulation around an enclosed area, which is related to the vorticity by the Stokes theorem. The latter two methods are detailed in Sect. 3.

The paper is organized as follows. In Sect. 2, we describe the details of our variational approach and the way we have adapted the energy function to be minimized in order to directly estimate significative fluid parameters such as vorticity and strain rate tensor components. Next, in Sect. 3 we present an overview of standard techniques from the literature to compute those quantities. In Sect. 4 we present numerical experiments on synthetic and real data and we analyze the accuracy of the proposed method in comparison with the standard techniques described in Sect. 3, as well as we characterize the sensitivity of our method with respect to the main parameters. Finally in Sect. 5 we present the main conclusions of the paper.

2 A variational approach for second order motion estimation

To estimate the optic flow of a given sequence we propose a variational approach based on the minimization of an energy function $E(\tilde{\mathbf{u}})$ defined for each point $\mathbf{x_0}$ with the special fact that it depends not only on the displacement vector components $\mathbf{u} = (u,v)^{\mathrm{T}}$, but also on their partial derivatives $\tilde{\mathbf{u}} = (u, v, u_x, u_y, v_x, v_y)^{\mathrm{T}}$, as shown in the following equation:

$$
\begin{aligned}
E(\tilde{\mathbf{u}}) &= E(u, v, u_x, u_y, v_x, v_y) \\
&= \int_{\Omega(\mathbf{x_0})} K_\sigma(\mathbf{x} - \mathbf{x_0}) \Big(I_1(\mathbf{x}) - I_2(\mathbf{x} + (u,v)^{\mathrm{T}} \\
&\quad + \begin{pmatrix} u_x & u_y \\ v_x & v_y \end{pmatrix} (\mathbf{x} - \mathbf{x_0})) \Big)^2 d\mathbf{x}
\end{aligned}
\tag{1}
$$

where $I_1(\mathbf{x})$ is the first image of the sequence and $I_2(\mathbf{x})$ is the following frame, where the time increment from frame to frame is assumed to be normalized ($\Delta t = 1$), $K_\sigma(\mathbf{x} - \mathbf{x_0})$ is a Gaussian kernel with standard deviation σ which weighs the pixels in the domain $\Omega(\mathbf{x_0})$ centered at point $\mathbf{x_0}$. Hence, the goal is to find the components of vector $\tilde{\mathbf{u}}$, that is, the displacement and its partial derivatives such that minimize the error between $I_1(\mathbf{x})$ and $I_2(\mathbf{x})$ displaced by the unknown flow.

At a first glance, the dependence on the partial derivatives (u_x, u_y, v_x, v_y) might seem a redundant operation since they can be computed from the obtained motion vector field. However, it can be shown that numerical differentiation of the estimated motion vector field yields to inaccurate results mainly due to undesired numerical error amplification (Raffel et al. 1998). The partial derivatives of the motion vector field are the components of the velocity gradient tensor, also called the rate of deformation tensor, $\mathbf{D} = d\mathbf{u}/d\mathbf{x}$ given by:

$$
\mathbf{D} = \frac{d\mathbf{u}}{d\mathbf{x}} = \begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix}
\tag{2}
$$

This tensor provides quantitative information of the unitary deformation rate of the flow. In a general context,

this tensor is not a symmetric matrix and it has four independent components. In the context of fluid flow analysis, this deformation rate tensor is usually decomposed into meaningful components related to physical phenomena. A common decomposition method is to divide the tensor into a symmetric and an antisymmetric matrix, as shown in Eq. 3.

$$\frac{d\mathbf{u}}{d\mathbf{x}} = \begin{bmatrix} u_x & \frac{1}{2}(u_y + v_x) \\ \frac{1}{2}(v_x + u_y) & v_y \end{bmatrix} + \begin{bmatrix} 0 & \frac{1}{2}(u_y - v_x) \\ \frac{1}{2}(v_x - u_y) & 0 \end{bmatrix} \tag{3}$$

This decomposition is uniquely determined for any matrix $\mathbf{A}$, where the symmetric part is given by $\frac{\mathbf{A}+\mathbf{A}^T}{2}$ and represents the strain rate tensor with the elongational strains on the diagonal and the shearing strains on the off-diagonal, whereas the antisymmetric part is given by $\frac{\mathbf{A}-\mathbf{A}^T}{2}$, which is the rate of rotation tensor, whose non-zero elements are the vorticity components.

Since vorticity and the strain rate tensor have a more relevant physical meaning in the context of fluid analysis than just the flow gradient it is possible to reformulate Eq. 1 as a function of the vorticity ω and the strain rate tensor components $\varepsilon_{1,1}, \varepsilon_{1,2}, \varepsilon_{2,2}$ performing the following change of variables:

$$\begin{cases} u_x = \varepsilon_{1,1} & u_y = \omega + \varepsilon_{1,2} \\ v_x = \varepsilon_{1,2} - \omega & v_y = \varepsilon_{2,2} \end{cases} \tag{4}$$

and now the expression of the energy we want to minimize is:

$$E(\tilde{\mathbf{u}}) = E(u,v,\omega,\varepsilon_{1,1},\varepsilon_{1,2},\varepsilon_{2,2}) = \int\limits_{\Omega(\mathbf{x_0})} K_\sigma(\mathbf{x}-\mathbf{x_0})$$
$$\left(I_1(\mathbf{x}) - I_2(\mathbf{x}+(u,v)^T + \begin{pmatrix} \varepsilon_{1,1} & \varepsilon_{1,2}+\omega \\ \varepsilon_{1,2}-\omega & \varepsilon_{2,2} \end{pmatrix}(\mathbf{x}-\mathbf{x_0})) \right)^2 d\mathbf{x} \tag{5}$$

2.1 Energy minimization

In order to be able to obtain the six components vector of unknowns $\tilde{\mathbf{u}} = (u,v,\omega,\varepsilon_{1,1},\varepsilon_{1,2},\varepsilon_{2,2})^T$ that minimize Eq. 5 we formulate the solution at step $n + 1$ as a function of the solution at step n and the six components vector of residuals $\tilde{\mathbf{h}} = (h_u, h_v, h_\omega, h_{\varepsilon_{1,1}}, h_{\varepsilon_{1,2}}, h_{\varepsilon_{2,2}})^T$ computed at each step, as shows the following expression:

$$\tilde{\mathbf{u}}^{n+1} = \tilde{\mathbf{u}}^n + \tilde{\mathbf{h}} \tag{6}$$

Then, introducing Eq. 6 in Eq 5 we are able to obtain an iterative operator towards the local minimum of the energy

function similar to a gradient descent algorithm Fletcher and Reeves (1964). But first, the following approximations are still necessary to simplify the term that depends on $I_2(\mathbf{x})$ in order to obtain our iterative operator:

$$I_2\left(\mathbf{x}+(u^{n+1},v^{n+1})^T\right.$$
$$\left.+\begin{pmatrix} \varepsilon_{1,1}^{n+1} & \varepsilon_{1,2}^{n+1}+\omega^{n+1} \\ \varepsilon_{1,2}^{n+1}-\omega^{n+1} & \varepsilon_{2,2}^{n+1} \end{pmatrix}(\mathbf{x}-\mathbf{x_0})\right)$$
$$= I_2\left(\mathbf{x}+(u^n,v^n)^T + \begin{pmatrix} \varepsilon_{1,1}^n & \varepsilon_{1,2}^n+\omega^n \\ \varepsilon_{1,2}^n-\omega^n & \varepsilon_{2,2}^n \end{pmatrix}(\mathbf{x}-\mathbf{x_0})\right.$$
$$\left.+(h_u,h_v)^T + \begin{pmatrix} h_{\varepsilon_{1,1}} & h_{\varepsilon_{1,2}}+h_\omega \\ h_{\varepsilon_{1,2}}-h_\omega & h_{\varepsilon_{2,2}} \end{pmatrix}(\mathbf{x}-\mathbf{x_0})\right) \tag{7}$$

To simplify notation, let us define I_2^n as:

$$I_2^n(\mathbf{x}) = I_2\left(\mathbf{x}+(u^n,v^n)^T \right.$$
$$\left.+ \begin{pmatrix} \varepsilon_{1,1}^n & \varepsilon_{1,2}^n+\omega^n \\ \varepsilon_{1,2}^n-\omega^n & \varepsilon_{2,2}^n \end{pmatrix} \begin{pmatrix} x-x_0 \\ y-y_0 \end{pmatrix}\right) \tag{8}$$

where the vector $\mathbf{x} - \mathbf{x}_0$ has been decomposed into their components $(x - x_0, y - y_0)^T$. The partial derivatives are denoted as:

$$I_{2,x}^n(\mathbf{x}) = \frac{\partial I_2}{\partial x}\left(\mathbf{x}+(u^n,v^n)^T \right.$$
$$\left.+ \begin{pmatrix} \varepsilon_{1,1}^n & \varepsilon_{1,2}^n+\omega^n \\ \varepsilon_{1,2}^n-\omega^n & \varepsilon_{2,2}^n \end{pmatrix} \begin{pmatrix} x-x_0 \\ y-y_0 \end{pmatrix}\right) \tag{9}$$

$$I_{2,y}^n(\mathbf{x}) = \frac{\partial I_2}{\partial y}\left(\mathbf{x}+(u^n,v^n)^T \right.$$
$$\left.+ \begin{pmatrix} \varepsilon_{1,1}^n & \varepsilon_{1,2}^n+\omega^n \\ \varepsilon_{1,2}^n-\omega^n & \varepsilon_{2,2}^n \end{pmatrix} \begin{pmatrix} x-x_0 \\ y-y_0 \end{pmatrix}\right) \tag{10}$$

Then, using the new notation, the solution at step $n + 1$ can be approximated by its Taylor expansion over $\tilde{\mathbf{u}}^n$ to provide:

$$I_2^{n+1}(\mathbf{x}) \simeq I_2^n(\mathbf{x}) + \left((h_u,h_v)^T \right.$$
$$\left.+ \begin{pmatrix} h_{\varepsilon_{1,1}} & h_{\varepsilon_{1,2}}+h_\omega \\ h_{\varepsilon_{1,2}}-h_\omega & h_{\varepsilon_{2,2}} \end{pmatrix} \begin{pmatrix} x-x_0 \\ y-y_0 \end{pmatrix}\right)^T \begin{pmatrix} I_{2,x}^n(\mathbf{x}) \\ I_{2,y}^n(\mathbf{x}) \end{pmatrix} \tag{11}$$

Finally, by using vectorial notation $I_2(\mathbf{x})^{n+1}$ can be expressed as:

$$I_2^{n+1}(\mathbf{x}) \simeq I_2^n(\mathbf{x}) + \hat{\mathbf{I}}_2^{\mathbf{n}}(\mathbf{x})^{\mathrm{T}} \cdot \tilde{\mathbf{h}} \qquad (12)$$

where

$$\begin{aligned}
\hat{\mathbf{I}}_2^{\mathbf{n}}(\mathbf{x}) = &(I_{2,x}^n(\mathbf{x}), I_{2,y}^n(\mathbf{x}), (y - y_0)I_{2,x}^n(\mathbf{x}) - (x - x_0)I_{2,y}^n(\mathbf{x}), \\
&\times (x - x_0)I_{2,x}^n(\mathbf{x}), (x - x_0)I_{2,y}^n(\mathbf{x}) + (y - y_0)I_{2,x}^n(\mathbf{x}), \\
&\times (y - y_0)I_{2,y}^n(\mathbf{x}))^{\mathrm{T}}
\end{aligned} \qquad (13)$$

Hence, introducing Eq. 12 in Eq. 5 we obtain the approximation for the energy function shown in Eq. 14, where, in addition, we introduce a second term in the energy function weighted by the parameter α. The role of this regularization term is to provide at every point a smooth vector field, by means of an additional constraint on the norm of the vector $\tilde{\mathbf{h}}$, which is forced to be small. In this sense, the parameter α determines the importance of this additional constraint on the vector field.

$$\begin{aligned}
\tilde{E}(\tilde{\mathbf{h}}) = &\int_{\Omega(\mathbf{x_0})} K_\sigma(\mathbf{x} - \mathbf{x_0})\Big(I_1(\mathbf{x}) - I_2^n(\mathbf{x}) - \hat{\mathbf{I}}_2^n(x)^{\mathrm{T}} \cdot \tilde{\mathbf{h}}\Big)^2 d\mathbf{x} \\
&+ \alpha \int_{\Omega(\mathbf{x_0})} \| \tilde{\mathbf{h}} \|^2 d\mathbf{x}
\end{aligned} \qquad (14)$$

This formulation allow us to easily obtain an analytical expression to compute the local minimum of the energy as a function of $\tilde{\mathbf{h}}$:

$$\begin{aligned}
\nabla\tilde{E}(\tilde{\mathbf{h}}) = &-2 \int_{\Omega(\mathbf{x_0})} K_\sigma(\mathbf{x} - \mathbf{x_0})\Big(I_1(\mathbf{x}) - I_2^n(\mathbf{x}) - \hat{\mathbf{I}}_2^n(\mathbf{x})^{\mathrm{T}} \cdot \tilde{\mathbf{h}}\Big) \\
&\times \hat{\mathbf{I}}_2^n(\mathbf{x})d\mathbf{x} + 2\alpha \int_{\Omega(\mathbf{x_0})} \tilde{\mathbf{h}}d\mathbf{x} = \mathbf{0}
\end{aligned} \qquad (15)$$

which is equivalent to:

$$\begin{aligned}
&\int_{\Omega(\mathbf{x_0})} K_\sigma(\mathbf{x} - \mathbf{x_0})\big(I_1(\mathbf{x}) - I_2^n(\mathbf{x})\big)\hat{\mathbf{I}}_2^n(\mathbf{x})d\mathbf{x} \\
&= \int_{\Omega(\mathbf{x_0})} \Big(K_\sigma(\mathbf{x} - \mathbf{x_0})(\hat{\mathbf{I}}_2^n(\mathbf{x}) \cdot \hat{\mathbf{I}}_2^n(\mathbf{x})^{\mathrm{T}}) + \alpha\mathbf{I}\Big)\tilde{\mathbf{h}}d\mathbf{x}
\end{aligned} \qquad (16)$$

This system of equations can be expressed with the standard matrix notation $\mathbf{Ax} = \mathbf{b}$, where the vector of unknowns in this case is the vector $\tilde{\mathbf{h}}$, while the system matrix and the independent term can be computed as:

$$\mathbf{A} = \int_{\Omega(\mathbf{x_0})} \Big(K_\sigma(\mathbf{x} - \mathbf{x_0})(\hat{\mathbf{I}}_2^n(\mathbf{x}) \cdot \hat{\mathbf{I}}_2^n(\mathbf{x})^{\mathrm{T}}) + \alpha\mathbf{I}\Big)dx \qquad (17)$$

$$\mathbf{b} = \int_{\Omega(\mathbf{x_0})} K_\sigma(\mathbf{x} - \mathbf{x_0})\big(I_1(\mathbf{x}) - I_2^n(\mathbf{x})\big)\hat{\mathbf{I}}_2^n(\mathbf{x})dx \qquad (18)$$

The solution of the system of equations is given by $\tilde{\mathbf{h}} = \mathbf{A}^{-1}\mathbf{b}$, so we only have to invert the 6×6 matrix $\mathbf{A}$ or use any other algorithm to solve the system of equations. We observe that if $\alpha > 0$, matrix $\mathbf{A}$ is positive definite and therefore the system of equations is well posed (it has a unique solution). It means that the parameter α avoids instabilities in the solution of the linear system of equations.

2.2 Numerical implementation

Next, we describe the main steps we followed to derive an efficient algorithm of the method proposed in Sect. 2, including the numerical considerations involved on the computation of integrals and spatial derivatives in the proposed method.

As it can be seen in Algorithm 1, our variational approach starts from an initial estimation of the motion vector field $\mathbf{u}^0$ which can be obtained using any other optic flow estimation method. Then, we start the iterative procedure towards the local minimum of the energy function given in Eq. 5. At each iteration we check the convergence of our algorithm in order to discard the solutions that do not provide the optimal response.

This algorithm is then applied for every point in the image (or a grid at a given scale) to obtain the desired first and second order flow parameters estimation, using information within a neighborhood of that point determined by the σ parameter of the Gaussian kernel.

The influence of the main parameters in this algorithm, σ and α, is detailed in Sect. 4. Concerning other parameters like N_{iter}, we remark that it should be as big as possible since it is automatically truncated when convergence is detected. Finally, the initialization of the energy at the first step requires the computation of the image derivatives. In our implementation, we have used central finite differences because it is a good compromise between an easy implementation, low time of computation and low error propagation Raffel et al. (1998).

3 Standard methods for vorticity estimation

In order to obtain an estimation of the vorticity field, alternatives to finite differencing have also been proposed to compute such magnitude. In this section we describe two of these methods that we will use later to compare the

performance of standard vorticity estimation algorithms with the performance of our variational approach.

where the unit vector $\hat{\mathbf{n}}$ is normal to the surface S. Stokes theorem can also be applied to the two dimensional vector field:

Algorithm 1 Implementation of the second order flow estimation method

Initialization of the parameters: σ (window size), α, $\tilde{\mathbf{u}}^0 = \mathbf{0}$, $E(\tilde{\mathbf{u}}^0)$, N_iter, $Failures = 0$, $MaxFailures$

Initial Estimation of the motion field $\mathbf{u}^0 = (u, v)$ with any estimation flow technique

for $n = 0$ to N_iter **do**

    Update $I_2^n(\mathbf{x})$ and $\hat{\mathbf{I}}_2^{\mathbf{n}}$ using $\tilde{\mathbf{u}}^n$

    Computation of the energy $E(\tilde{\mathbf{u}}^n)$, $\mathbf{A}$ and $\mathbf{b}$

    Solve system of equations $\tilde{\mathbf{h}} = \mathbf{A}^{-1}\mathbf{b}$

    if $E(\tilde{\mathbf{u}}^n) \leq E(\tilde{\mathbf{u}}^{n-1})$ **then**

        $\tilde{\mathbf{u}}_{optimized} = \tilde{\mathbf{u}}^n + \tilde{\mathbf{h}}$ {The method converges towards the solution}

        Update $\tilde{\mathbf{u}}^{n+1}$

        $Failures = 0$

    else if $E(\tilde{\mathbf{u}}^n) \geq E(\tilde{\mathbf{u}}^{n-1})$ and $Failures \leq MaxFailures$ **then**

        Update $\tilde{\mathbf{u}}^{n+1}$

        Reject this solution as $\tilde{\mathbf{u}}_{optimized}$ {The method diverges from the solution}

        $Failures + +$

    else

        Exit

    end if

end for

3.1 Circulation method

By definition the vorticity is related to the circulation by Stokes theorem:

$$\Gamma = \oint \mathbf{u} \cdot d\mathbf{l} = \int (\nabla \times \mathbf{u}) \cdot d\mathbf{S} = \int \omega \cdot d\mathbf{S} \qquad (19)$$

where $\mathbf{l}$ describes the path of integration around a surface $\mathbf{S}$. The vorticity for a fluid element is found by reducing the surface $\mathbf{S}$ and with it the path $\mathbf{l}$, to zero:

$$\hat{\mathbf{n}} \cdot \omega = \hat{\mathbf{n}} \cdot \nabla \times \mathbf{u} = \lim_{\mathbf{S} \to 0} \frac{1}{S} \oint \mathbf{u} \cdot d\mathbf{l} \qquad (20)$$

$$(\overline{\omega}_Z)_{i,j} = \frac{1}{A} \Gamma_{i,j} = \frac{1}{A} \oint_{l(x,y)} (u, v) \cdot d\mathbf{l} \qquad (21)$$

where $(\overline{\omega}_Z)_{i,j}$ reflects the average vorticity within the enclosed area.

In practice, Eq. 21 is implemented by choosing a small rectangular contour around which the circulation is calculated using a standard integration scheme such as the trapezoidal rule. The local circulation is then divided by the enclosed area to arrive at an average vorticity, as shows the following equation which uses a neighborhood of eight points:

$$(\omega_z)_{i,j} \cong \frac{\Gamma_{i,j}}{4\Delta x \Delta y} \qquad (22)$$

where

$$\begin{aligned}
\Gamma_{i,j} = &\frac{1}{2}\Delta x(u_{i-1,j-1} + 2u_{i,j-1} + u_{i+1,j-1}) \\
&+ \frac{1}{2}\Delta y(v_{i+1,j-1} + 2v_{i+1,j} + v_{i+1,j+1}) \\
&- \frac{1}{2}\Delta x(u_{i+1,j+1} + 2u_{i,j+1} + u_{i-1,j+1}) \\
&- \frac{1}{2}\Delta y(v_{i-1,j+1} + 2v_{i-1,j} + v_{i-1,j-1})
\end{aligned} \qquad (23)$$

3.2 Second order polynomial fit and analytic differentiation

The second standard method we use to compare our method with was initially proposed in Fouras and Soria (1998). For this method, it is necessary to provide an initial estimation of the motion vector components $(u_{i,j}, v_{i,j})$ within a certain neighborhood centered at the point (i_0, j_0) where we want to compute the vorticity. Then, the components $(u_{i,j}, v_{i,j})$ are separately fitted to a basis of polynomials $\{P_k(x,y)\}$ of K-th power using a χ^2 procedure, as proposed in Press et al. (1992). The linear combination of the polynomials requires $M = (K + 1)^2$ coefficients. Thus, the number of data points N in the neighborhood required for the χ^2 fitting process is $N \geq M$.

In other words, the χ^2 fitting procedure provides the coefficients u_k and v_k we need to express our initial motion estimation vector as a linear combination of the elements of the basis of polynomials:

$$u_{i,j} = \sum_{k=0}^{M-1} u_k P_k(x, y) \qquad (24)$$

$$v_{i,j} = \sum_{k=0}^{M-1} v_k P_k(x, y) \qquad (25)$$

Then, it is straightforward to analytically differentiate the previous expressions. The numerical value of the vorticity at the current position (i,j) is given by the following expression evaluated in $(x = 0, y = 0)$:

$$(\omega_z)_{i,j} = \sum_{k=0}^{M-1}\left(v_k \frac{\partial P_k(x=0, y=0)}{\partial x} - u_k \frac{\partial P_k(x=0, y=0)}{\partial y} \right) \qquad (26)$$

In this method, the set of basis functions and the samples around the point of interest can be seen as parameters of the method since different possibilities are possible. In our

study, as proposed in Fouras and Soria (1998), we have chosen the following set of second order polynomials in x and y as basis functions:

$$P(x,y) \in \{1, x, x^2, y, y^2, xy, x^2y, xy^2, x^2y^2\} \qquad (27)$$

This results in $M = 9$ coefficients to be fitted in the decomposition described by Eqs. 24 and 25. Thus, we need at least 9 samples from a 3×3 neighborhood centered at point (i,j), which is the neighborhood used in our experiments.

4 Numerical experiments

In order to show the performance of the proposed approach, we present here the numerical experiments we have performed to evaluate the accuracy of the method. We perform experiments on synthetic and real data. In the case of synthetic experiments we use flow models with a known analytic expression, in this way, we are able to compare our results with the true solution we are looking for, also called *ground truth* in the field of computer vision. In this sense we focus our attention in the flow estimation improvement we obtain as well as in the vorticity estimation because its interest in flow analysis. In addition, we compare our vorticity estimation with the methods detailed in Sect. 3.

In any case, the algorithm proposed in this paper starts from an initial estimation of the flow which can be obtained by any desired method. This initialization must be, at least, a rough approximation of the true motion vector field in order to the algorithm be able to converge. In our work, we use two different methods drawn from the state of the art to achieve such initialization in order to evaluate its influence on the obtained results.

The first method (PDE) used in our experiments is based on a classical PDE scheme, presented in Alemán et al. (2005), composed by a data term that assumes that the image intensity is equal in two corresponding points and a regularization term introduced in Anandan (1989) and studied in Alvarez et al. (2000) which allows discontinuities preservation on the flow. In our implementation, we use a pyramidal approach in order to speed up the algorithm and to avoid the convergence towards spurious local minima.

The second method (COR) used in our experiments is based on the multi-step iterative computation of cross-correlation measures where the position of the interrogation window is displaced with subpixel precision at each iteration depending on the precomputed flow, as it is described in Scarano (2002). To find the peak location in the correlation map, we use a Gaussian interpolation scheme based on a 3×3 kernel Westerweel (1993),

Fig. 1 *Left* Synthetic vortex flow generated with Eq. 28. *Right* True Vorticity field obtained from the analytical expression (Eq. 29)
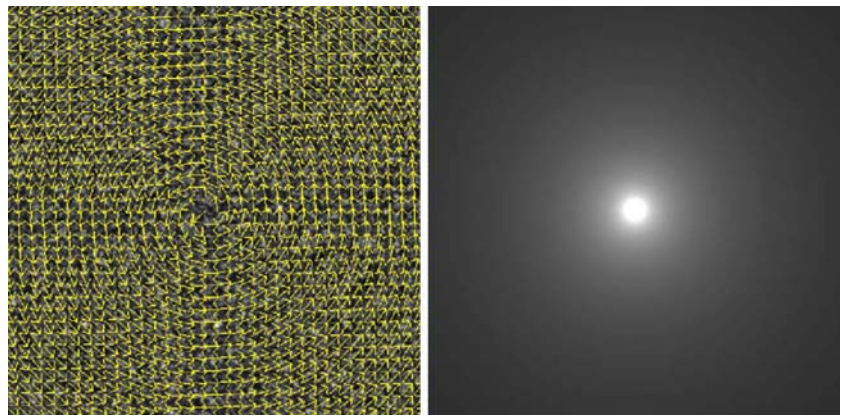
whereas a bilinear interpolation scheme to perform the displacement of the interrogation window with subpixel precision.

4.1 Experiment 1: Synthetic vortex flow

In our first experiment, we have built a sequence of two PIV images of 1,024 × 1,024 pixels. The first image was synthesized by the research institute CEMAGREF (Rennes, France) using a uniform random distribution of the particles over the whole image, with a constant particle concentration of 256 particles each 32 × 32 window. The particle size follows a random normal distribution with an average of 1.25 pixels and a standard deviation of 0.25 pixels. The second image is computed by applying the synthetic vortex flow modeled in Eq. 28 and shown on the left side of Fig. 1. In this sense, we are able to obtain the analytic expression of the vorticity, given in Eq. 29 and shown on the right side of Fig. 1, which can be used as the ground truth to evaluate our approach.

$$\mathbf{u}(\mathbf{x}) = \begin{cases} u = \frac{-3y}{\sqrt{x^2 + y^2}} \\ v = \frac{3x}{\sqrt{x^2 + y^2}} \end{cases} \tag{28}$$

Table 1 Angular and Euclidean error computed between the vortex flow in Eq. 28 and the estimations performed with the different approaches studied in this paper: PDE, COR and our method, where the initialization flow is specified in brackets

| Error	Mean angular error	Mean Euclidean error
True, PDE	0.4308	3.1e-2
True, COR	0.2737	1.9e-2
True, Our method (PDE)	0.0328	9.3e-4
True, Our method (COR)	0.0347	9.7e-4
True, Our method (True)	0.0155	2.9e-4

$$\omega = v_x - u_y = \frac{3}{\sqrt{x^2 + y^2}} \tag{29}$$

Table 1 shows some quantitative results that demonstrate the performance of our method in comparison with other methods drawn from the state-of-the-art, which were briefly described at the beginning of this section. In our experiments, we computed the average angular error and average Euclidean error of the flow components directly obtained from the PDE and COR methods. Then, we use those responses as an initialization of our method, yielding to a significant improvement of the error measures, as it can be seen in the table. Finally, to evaluate the noise introduced in our approach, we initialize our numerical scheme with the ground truth flow field, yielding to an error rate of the same order of magnitude as in the previous case.

Figure 2 shows a detail of the results obtained with the different algorithms under study using a PDE scheme to compute the initial estimation of the flow. On the left side, we show the vorticity obtained with our variational method. In the middle, the vorticity obtained with the circulation method detailed in Sect. 3.1 is shown, while the image on the right corresponds to the χ^2 fitting method with analytic differentiation (χ^2 method, from now on) detailed in Sect. 3.2. The results show that, in all cases, a data validation step is required in order to eliminate the outliers that appear on the images, which mainly appear by the fact that usually not all the particles are detected correctly in a PIV sequence, as it is justified in Ruhnau et al. (2005a). Hence, this effect leads to a bad convergence of the algorithm on spread points over the whole image. It is important to remember that the proposed energy minimization is resolved *locally* for every point of the image and, since no *global* regularization term is used to impose *global* regularity on the estimated vector field. Hence, the convergence from the input data may lead to a local minimum different from the global one. Fortunately, the noise pattern that appears can be easily removed using a median filter, as it is described in Senel et al. (2002), as a data
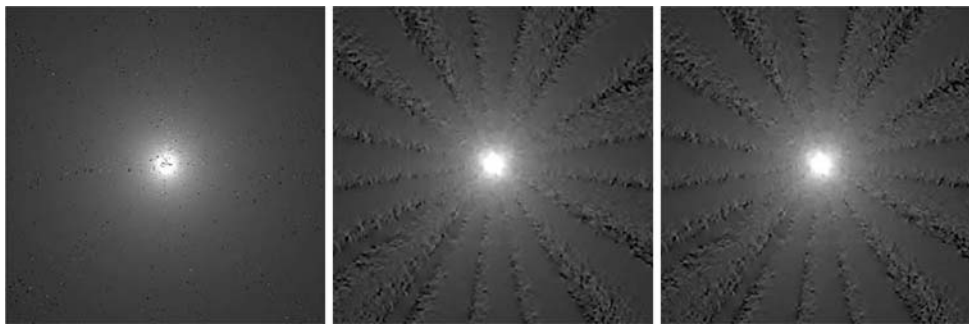
Fig. 2 Estimated vorticity before removing the outliers with a vortex flow model using PDE scheme for initial estimation. *Left* Results with our variational approach. *Middle* Results with the circulation method. *Right* Results with the χ^2 method
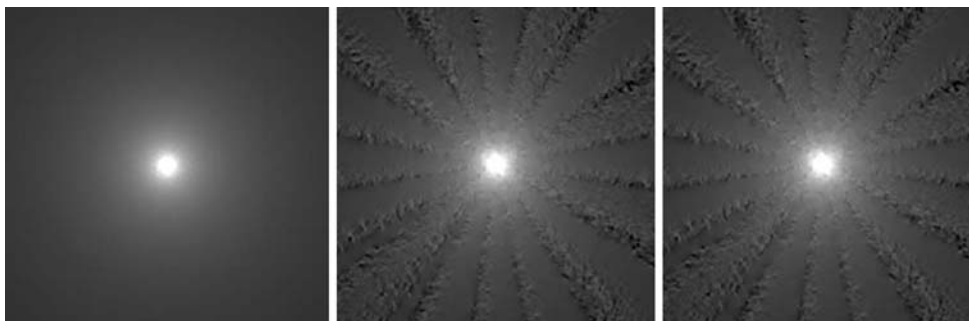
Fig. 3 Estimated vorticity after removing the outliers with a vortex flow model using PDE scheme for initial estimation. *Left* Results with our variational approach. *Middle* Results with the circulation method. *Right* Results with the χ^2 method
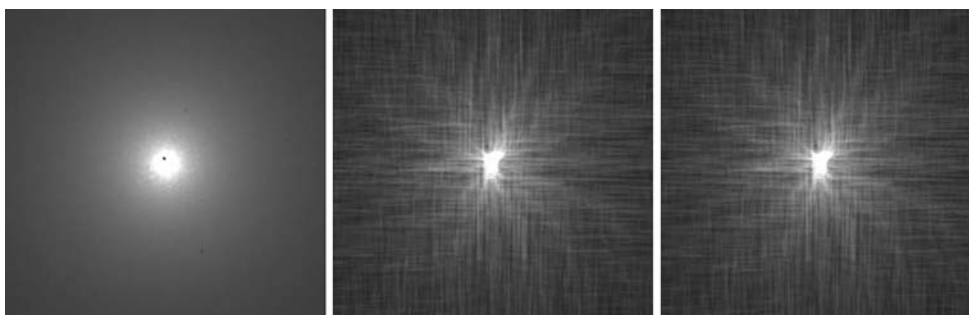
Fig. 4 Estimated vorticity after removing the outliers with a vortex flow model using a correlation scheme for initial estimation. *Left* Results with our variational approach. *Middle* Results with the circulation method. *Right* Results with the χ^2 method

Table 2 Mean error computed for the different vorticity estimation methods

Init flow	Our method	Circulation	χ^2 method
True	3.07e-5	3.45e-6	4.10e-6
PDE	1.35e-4	2.37e-3	2.39e-3
COR	1.38e-4	1.80e-3	1.80e-3
Our method (PDE)	N/A	2.50e-4	1.91e-4
Our method (COR)	N/A	2.75e-4	2.01e-4

validation post-processing step. In our experiments, to avoid the modification of correct signal values, we only substitute the current value by the median when we detect that it might

be an outlier, since its value is quite different from the median.

Figure 3 shows the validated results after the data validation step. Now, it can be seen that the dispersed noise has successfully been removed improving the quality of the estimations.

In order to check the influence of the initial flow estimation on the results, we use again the COR scheme as initialization to compute the vorticity. Figure 4 shows a detail of the results obtained using the new initial estimation as input for the vorticity computation algorithms once the outliers have been removed with the median filter, as it was done in the last case.

Finally, in Table 2, we present quantitative error measurements, following Eq. 30, to evaluate the accuracy of our approach for vorticity estimation in comparison with the ground truth model. The first column indicates the estimated vector field from which the vorticity is computed with the corresponding schemes described in Sect. 3. In our experiments, we first compute the vorticity field from the ground truth vector field. In this case, the error rate obtained with the circulation and χ^2 methods outperforms our results, since those schemes perform the computation directly from the true field and our method slightly modifies it. On the other situations, our method outperforms their results because of the numerical amplification of the initial error on the estimated flow. In this sense, even the vorticity given by the circulation or χ^2 methods, using the flow provided by our method, yields to a lower error rate than using the PDE or COR flows.

$$\text{Error} = \frac{\sum_{i=1}^{N_x} \sum_{j=1}^{N_y} |\omega_{i,j}^{\text{TRUE}} - \omega_{i,j}^{\text{est}}|}{N_x N_y} \quad (30)$$
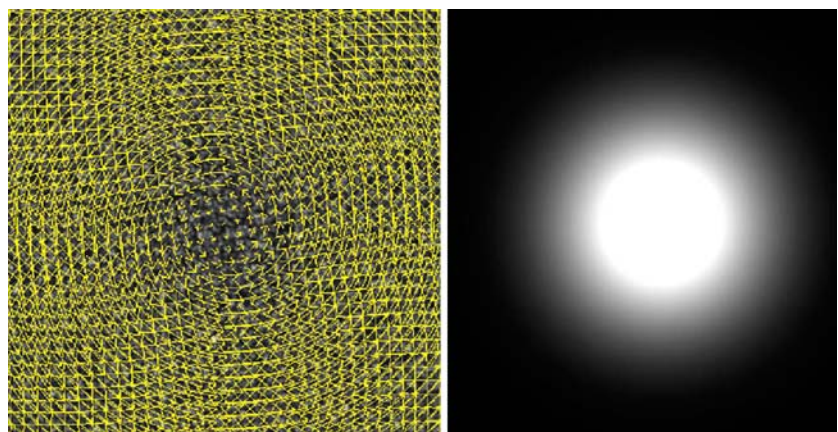
4.2 Experiment 2: Synthetic Lamb–Oseen flow

To validate our algorithm, we have again built a sequence of two PIV images of $1{,}024 \times 1{,}024$ pixels using the same procedure detailed in the previous experiment, but following the mathematical flow model shown in Eq. 31, also known as Lamb–Oseen vortex:

$$\mathbf{u}(\mathbf{x}) = \begin{cases} u = \dfrac{-\Gamma_0 y}{2\pi\sqrt{x^2+y^2}}\left(1 - \exp(-\tfrac{x^2+y^2}{4vt})\right) \\ v = \dfrac{-\Gamma_0 x}{2\pi\sqrt{x^2+y^2}}\left(1 - \exp(-\tfrac{x^2+y^2}{4vt})\right) \end{cases} \quad (31)$$

where v is the cinematic viscosity and Γ_0 is the initial circulation. In our experiments we used $\Gamma_0 = 0.05$ m^2/s and $\sqrt{4vt} = \frac{1}{6}$, which can be seen as the radius of the

Table 3 Angular and Euclidean error computed between the Lamb–Oseen flow in Eq. 31 and the estimations performed with the different approaches studied in this paper: PDE, COR and our method, where the initialization flow is specified in brackets

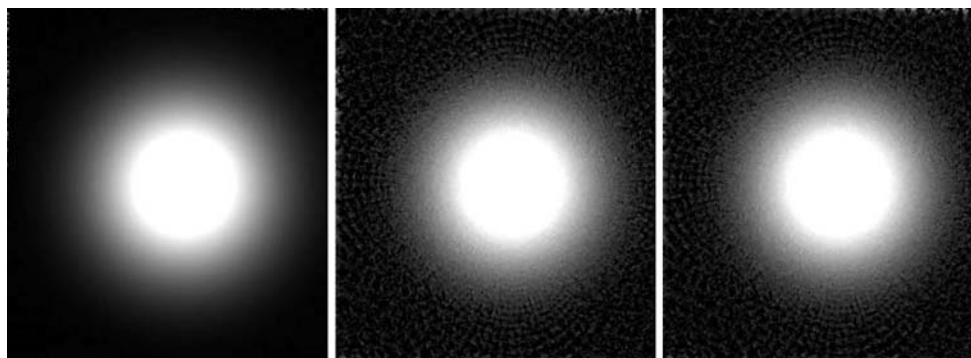| Error	Mean angular error	Mean Euclidean error
True, PDE	0.1132	0.0574
True, COR	0.9642	0.4601
True, Our method (PDE)	0.0571	0.0083
True, Our method (COR)	0.8938	0.1968
True, Our method (True)	0.0112	0.0015

Lamb–Oseen Vortex. Then, Eq. 32 presents the analytic expression of the vorticity.

$$\omega = v_x - u_y = \frac{\Gamma_0}{\pi 4vt} \exp\left(-\frac{x^2+y^2}{4vt}\right) \quad (32)$$

Figure 5 shows in the left side the synthetic Lamb–Oseen vortex and in the right side the vorticity computed with the parameters used in the computation of the flow.

In Table 3 we show some quantitative results that demonstrate the behavior of our method. As it was done in the previous experiment, we computed the average angular error and average Euclidean error of the flow components directly obtained from the PDE and COR methods. Then, these flows are used as an initialization of our approach and again the flow obtained is improved. To emphasize the behavior of our approach with respect to the noise, we computed the error rate using the ground truth flow as initialization. In this case, we obtain a very low error rate, which shows that the noise introduced by our method is almost negligible.

Figure 6 shows the estimated out-of-plane vorticity obtained with our approach (image on the left) in comparison to the circulation method (in the middle) and the χ^2 fitting method (on the right) once a median filter has been used to eliminate the outliers as a data validation postprocessing step.

Fig. 5 *Left* Synthetic Lamb–Oseen vortex generated with Eq. 31. *Right* True vorticity field obtained from the analytical expression (Eq. 32)

Fig. 6 Estimated vorticity after removing the outliers with a Lamb–Oseen flow model using PDE scheme for initial estimation. *Left* Results with our variational approach. *Middle* Results with the circulation method. *Right* Results with the χ^2 method
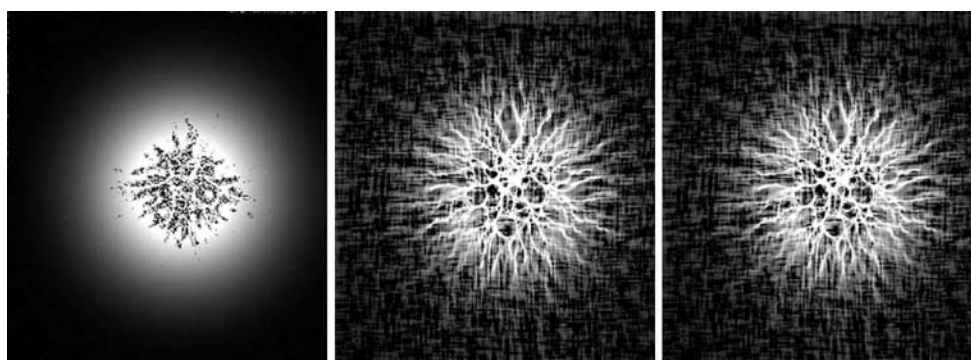
Fig. 7 Estimated vorticity after removing the outliers with a Lamb–Oseen flow model using a correlation scheme for initial estimation. *Left* Results with our variational approach. *Middle* Results with the circulation method. *Right* Results with the χ^2 method

Table 4 Mean error computed for the different vorticity estimation methods

Init flow	Our method	Circulation	χ^2 method
True	9.12e-5	1.01e-6	8.11e-7
PDE	4.49e-4	3.88e-3	3.94e-3
COR	1.98e-2	4.57e-2	4.59e-2
Our method (PDE)	N/A	6.09e-4	5.77e-4
Our method (COR)	N/A	2.45e-2	2.37e-2

Again, to validate the robustness of our approach, we use the correlation-based technique previously presented in the our experiments in order to perform an initial estimation of the flow. Figure 7 shows the results obtained for the computation of the out-of-plane vorticity after applying the median filter to eliminate the outliers. From that image, we can conclude the quality of our variational estimator.

To conclude, we present in Table 4 the mean error computed following Eq. 30 in order to evaluate the quality of the estimated vorticity in comparison with the ground truth model. In the table, the first column indicates the estimated vector field used to compute the vorticity and the numerical scheme used is indicated on the first row of the other columns. As in the previous case, the circulation and χ^2 methods outperforms our results, since those schemes perform the computation directly from the true field and our method performs computations on a modified flow field. However, on the other cases, our method outperforms their results due to numerical amplification of the initial error. It is interesting to notice that even when the vorticity is computed with the circulation and χ^2 method from the flow given by our variational approach, the error rate is reduced.

4.3 Experiment 3: Sensitivity to main parameters

The main parameter in our numerical scheme is the standard deviation of the Gaussian Kernel σ. The role of this parameter is to determine the influence on the estimated value of every sample within the neighborhood around the target point. In this sense, the smaller value of sigma, the smaller is the neighborhood and less important are their associated weights.

Figure 8 shows the evolution of the error magnitude in the estimation of the vorticity field (top left) and the
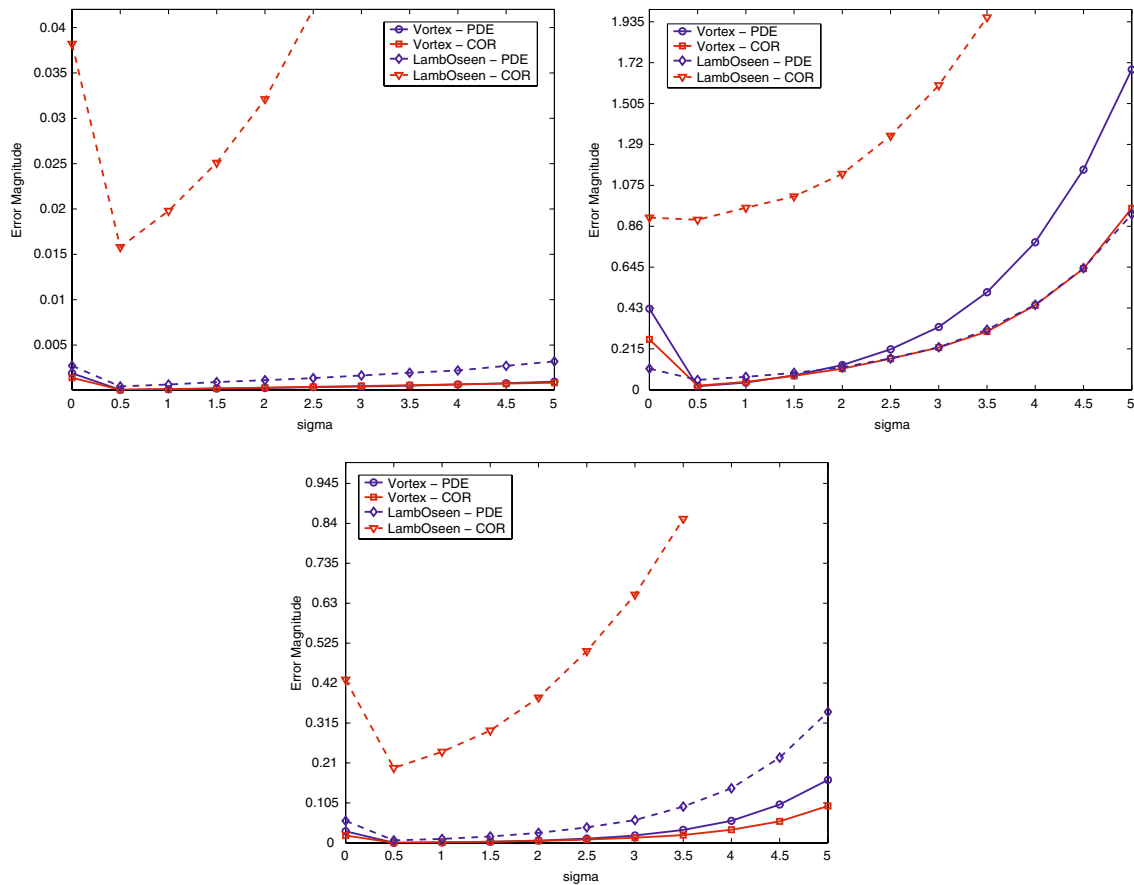
Fig. 8 Evolution of the error with respect to the parameter σ and the initialization method (PDE in *red*, COR in *blue*) for the vortex flow (*solid line*) and Lamb–Oseen flow (*dashed line*). *Top left* Error magnitude on the vorticity field. *Top right* Angular error on the flow field. *Bottom* Euclidean error on the flow field

angular (top right) and Euclidean error (bottom) of the motion vector field. As it can be seen, when $\sigma = 0$ the error in all cases is equivalent to the error on the initial estimation, since only one sample is taken to perform the computation. Then, for $\sigma \in (0.5, 1)$ we obtain the minimum error independently of the flow used to initialize our approach. Finally, as long as σ increases, the error magnitude also increases due to an overweighting of the samples in the neighborhood.

The parameter σ is the parameter that has more influence in the computation time of our algorithm, apart from the image size, of course, since it determines the number of samples used to perform the estimation for every point. In this sense, when σ is large the neighborhood size is also large and the computation time increases. On the contrary, the smaller the sigma, the faster we get the response since fewer operations have to be carried out. For the optimal $\sigma \in (0.5, 1)$ we obtain an average computation time of about 1 min for a 256×256 image.

On the other hand, the sensitivity with respect to the parameter α, which is used to obtain a smooth solution, is much less important. In other words, there exists a wide range of values for parameter α that do not affect to the solution obtained. For instance, in the Experiments 1 and 2 we obtain the results shown in the corresponding tables using values of α from 1e-6 to 100. For $\alpha > 100$ the error rate starts increasing, although slowly.

4.4 Experiment 4: Real PIV data

Finally, we present the results we obtained with our approach using real PIV data, which were provided by the research institute CEMAGREF (Rennes, France). The experimental image sequence was carried out in a wind tunnel using a PIV technique in order to study the wake behind a circular cylinder, as detailed in Parnaudeau et al. (2006). An initial estimation of the motion vector field is shown in Fig. 9. The main control parameters of this experiment are the Reynolds number $Re = 3,900$, which is based on the free stream velocity ($U_e = 4.8$ m/s for this experiment), the cinematic viscosity $v = 15$e-6 m^2/s, the diameter of the circular cylinder $D = 12$ mm and the time interval between the two images of each pair $\Delta t = 25$ μs, but the time between pairs is
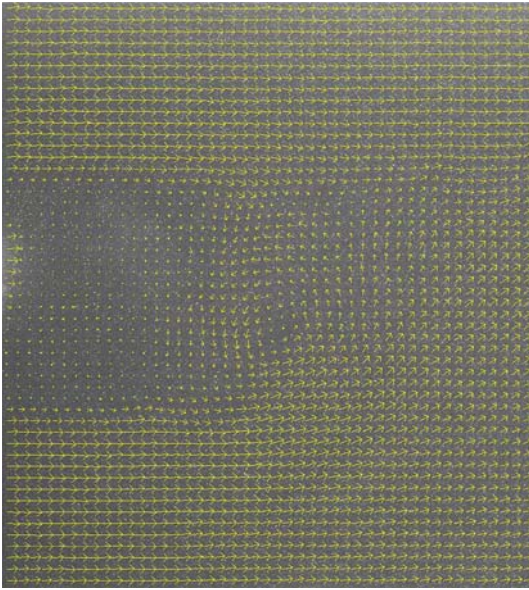
Fig. 9 Experimental motion vector field of the wake behind a circular cylinder

much larger so there is no time correlation between any two of them.

In real PIV images, the intensity pattern changes a lot from an image to the next one due to technological limitations of the camera and illumination procedures. Since variational methods are based on the assumption that pixel

intensity does not change from the initial to the final position, it is necessary to preprocess the input images in order to normalize them with respect to light conditions. To do that, we compute the mean m_1 of the image $I_1(x,y)$ and define the normalized images $\hat{I}_1(x,y)$ and $\hat{I}_2(x,y)$ by:

$$\hat{I}_1(x,y) = m_1 + I_1(x,y) - G_\sigma * I_1(x,y)$$

$$\hat{I}_2(x,y) = m_1 + I_2(x,y) - G_\sigma * I_2(x,y)$$

where G_σ is a Gaussian kernel with standard deviation σ, which is supposed to be quite big ($\sigma = 32$ in the numerical experiments). In this way, normalization is achieved by the subtraction to the original image of an average of the image in a large neighborhood. Then, the constant m_1 is added to obtain two images with a similar range to that of I_1. Of course, this normalization will not work for usual images, but with PIV images we can assume that the information of the particle displacement depend just on the local intensity variation and it is independent of the local intensity average, which is what we remove with this normalization.

On Fig. 10 we compare the regularized flow given by our approach with the initial flow estimation obtained from the normalized input sequence with the same variational approach used in the synthetic experiments. The first row in that figure shows on the left the u component and on the right the v component of the initial motion vector field. In

Fig. 10 Comparison of the initial estimation of the motion vector field and the regularized response from our variational approach. Top Row: Initial estimated flow (*Left u* component of the flow. *Right v* component of the flow). *Bottom row* Regularized flow obtained with our variational approach (*Left u* component of the flow. *Right v* component of the flow)
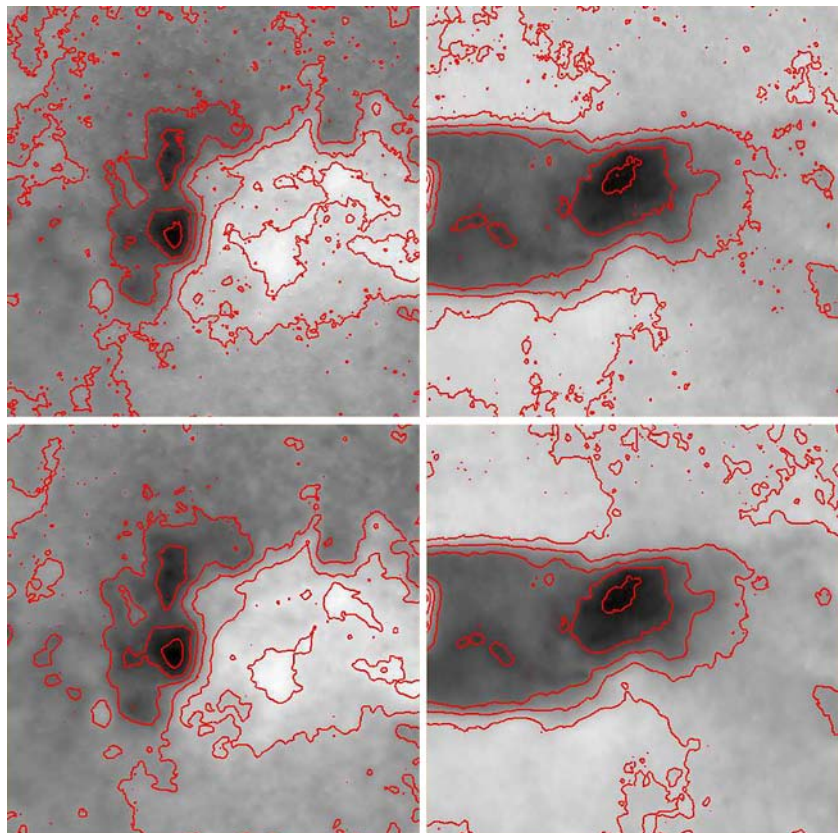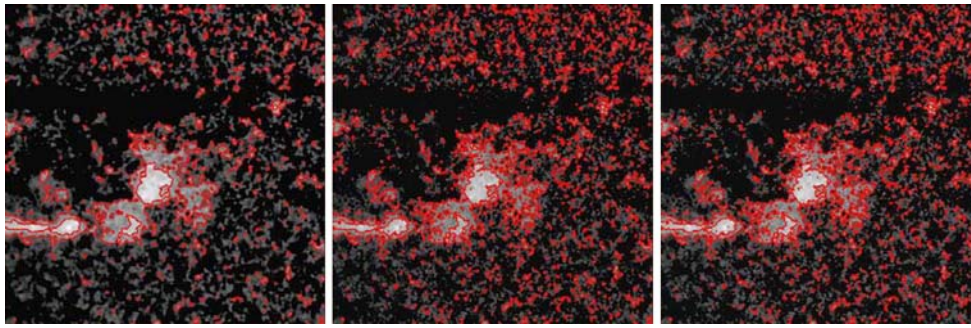
Fig. 11 Estimated vorticity for the experimental wake sequence. *Left* Results with our variational approach. *Middle* Results with the circulation method. *Right* Results with the χ^2 method

an analogous way, the second row shows on the left the u component and on the right the v component of the regularized motion vector field. In the images, some isocontour lines have been represented in order to make easier the comparison of the given results. In this sense, it can be seen that the components of the flow are smoother in the response of our variational approach.

Figure 11 shows the vorticity estimated with our approach in comparison with the results from the other methods described in this paper. Again, for an easier interpretation of the results, some contour lines have been represented for the same vorticity values obtained with the different methods. From the images, it can be seen that the vorticity field obtained with our variational approach is smoother than the field provided by the other methods. Moreover, a certain continuity on the derivatives is observed, while the response of the other methods contains abrupt changes in the vorticity field.

5 Conclusions

In this paper, we have presented a new variational method to estimate simultaneously the flow in a PIV sequence and second order motion magnitudes such as the out-of-plane vorticity and the strain rate tensor directly from the image sequence since they are directly related to the derivatives of the motion vector field.

In this sense, our variational method not only improves the quality of the motion vector field provided by the methods we use in our study, but also the estimation of the out-of-plane vorticity and strain rate tensor components is improved by means of an energy function that depends on the partial derivatives of the motion vector field. Our quantitative results show that the vorticity average error is at least an order of magnitude lower in comparison to the average error obtained with other standard procedures to compute the vorticity found in the literature. At the same

time, error measurements also indicate that the quality of the initial flow estimation is improved up to an order of magnitude.

The robustness of our method is evaluated using two different initial flow estimations as input for our scheme, one of them a PDE-based approach and the other one a correlation-based technique. In any case, a data validation step is needed in order to eliminate some outliers inherent to the response of our approach, mainly due to the convergence towards a local minimum different from the absolute one, since no global regularization is imposed to the response of our approach. These outliers can be easily removed by means of a median filter since they appear at sparse locations all around the images.

For real data, we have also obtained encouraging results that demonstrate a good performance of our approach. In this case, it is required a normalization of the light conditions as a preprocessing step.

References

Abrahamson S, Lonnes S (1995) Uncertainty in calculating vorticity from 2d velocity fields using circultaion and least-squares approaches. Exp Fluids 20:10–20

Adrian R (1988) Statistical properties of particle image velocimetry measurements in turbulent flow. Laser Anemometry Fluid Mech 1:115–129

Alemßn M, Alvarez L, Gonzßlez E, Mazorra L, Sßnchez J (2005) Optic flow estimation in fluid images i. Cuadernos Instituto Universitario de Ciencias y Tecnologfas CibernTticas 31:1–25

Alvarez L, Weickert J, Sßnchez J (2000) Reliable estimation of dense optical flow fields with large displacements. Int J Comput Vis 39(1):41–56

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

Corpetti T, Memin E, Perez P (2002) Dense estimation of fluid flows. IEEE Trans Pattern Anal Mach Intell 24(3):365–380

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

Etebari A, Vlachos P (2005) Improvements on the accuracy of derivative estimation from dpiv velocity measurements. Exp Fluids 39:1040–1050

Fletcher R, Reeves C (1964) Function minimization by conjugate gradients. Comput J 7:149–154

Foucaut JM, Stanislas M (2002) Some considerations on the accuracy and frequency response of some derivative filters applied to particle image velocimetry vector fields. Meas Sci Technol 13:1058–1071

Fouras A, Soria J (1998) Accuracy of out-of-plane vorticity measurements derived from in-plane velocity field data. Exp Fluids 25:409–430

Hunt J (1987) Vorticity and vortex dynamics in complex turbulent flows. In: Transactions Canadian Society for Mechanical Engineering (ISSN 0315-8977), vol 11, pp 21–35

Keane R, Adrian R (1990) Optimization of particle image velocimeters. I. double pulsed systems. Meas Sci Technol 1(10):1202–1215

Keane R, Adrian R (1992) Theory of cross-correlation analysis of piv images. Appl Sci Res 49(3):191–215

Kim J, Lee D (1996) Optimized compact finite difference schemes with maximum resolution. AIAA J 34(5):887–893

Luff JD, Drouillard T, Rompage AM, Linne M, Hertzberg JR (1999) Experimental uncertainties associated with particle image velocimetry (PIV) based vorticity algorithms. Exp Fluids 26:36–54

Nogueira J, Lecuona A, Rodrfguez PA, Alfaro JA, Acosta A (2005) Limits on the resolution of correlation piv iterative methods. practical implementation and design of weighting functions. Exp Fluids 39:314–321

Parnaudeau P, Carlier J, Heitz D, Lamballais E (2006) Experimental and numerical studies of the flow over a circular cylinder at reynolds number 3900. Phys Fluids (submitted)

Press W, Teukolsky S, Vetterling W, Flannery B (1992) Numerical recipes in C: The Art of Scientific Computing. Cambridge University Press, New York, NY, USA

Raffel M, Willert C, Kompenhans J (1998) Particle image velocimetry. A practical guide. Springer, Heidelberg

Ruhnau P, Guetter C, Putze T, Schnorr C (2005a) A variational approach for particle tracking velocimetry. Meas Sci Technol 16:1449–1458

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

Scarano F (2002) Iterative image deformation methods in PIV. Meas Sci Technol 13:R1–R19

Senel H, Peters R, Dawant B (2002) Topological median filters. IEEE Trans Image Process 11(2):89–104

Soria J, Giles CL, Omlin C (1996) An investigation of the near wake of a circular cylinder using a video-based digital cross-correlation particle image velocimetry technique. Exp Thermal Fluid Sci 12(2):221–233

Thomas M, Misra S, Kambhamettu C, Kirby J (2005) A robust motion estimation algorithm for piv. Meas Sci Technol 16:865–877

Westerweel J (1993) Digital particle image velocimetry. Theory and application. Delft University Press, Delft

Westerweel J (1997) Fundamentals of digital particle image velocimetry. Meas Sci Technol 8:1379–1392

Westerweel J, Dabiri D, Gharib M (1997) The effect of a discrete window offset on the accuracy of cross-correlation analysis of digital piv recordings. Exp Fluids 23:20–28