NUMERICAL MATHEMATICS: Theory, Methods and Applications
Numer. Math. Theor. Meth. Appl., Vol. **xx**, No. x, pp. 1-23 (20xx)

Wavelets and Optical Flow Motion Estimation

P. Dérian*, P. Héas, C. Herzet and E. Mémin

INRIA Rennes - Bretagne Atlantique,
Campus de Beaulieu, 35042 Rennes CEDEX, France.

Abstract. This article describes the implementation of a simple wavelet-based optical-flow motion estimator dedicated to continuous motions such as fluid flows. The wavelet representation of the unknown velocity field is considered. This scale-space representation, associated to a simple gradient-based optimization algorithm, sets up a well-defined multi-resolution framework for the optical flow estimation. Moreover, a very simple closure mechanism, approaching locally the solution by high-order polynomials, is provided by truncating the wavelet basis at fine scales. Accuracy and efficiency of the proposed method is evaluated on image sequences of turbulent fluid flows.

AMS subject classifications: 68T45, 65T60, 76F99

Key words: wavelets, optical flow, motion estimation, large displacements, fluid flows.

1. Introduction

Recent years have seen significant progress in signal processing techniques for fluid motion estimation. The wider availability of image-like data, whether coming from experimental facilities (e.g. particle image velocimetry) or from larger-scale geophysical study systems such as lidars or meteorological and oceanographical satellites, strongly motivates the development of computer-vision methods dedicated to their analysis. Correlation-based and variational methods have proven to be efficient in this context. However, the specific nature of fluid motion highly complicates the process. Indeed, one has to deal with continuous fields showing complex structures evolving at high velocities. This is particularly problematic with optical flow methods, where the problem non-linearity requires to resort to an ad-hoc multi-resolution strategy. Although leading to good empirical results, this technique is known to have a number of drawbacks. Moreover, the underdetermined nature of the optical flow estimation problem imposes to add some prior information about the sought motion field. In many contributions dealing with rigid-motion estimation, first-order regularization is considered with success. However, when tackling more challenging problems such as motion estimation

---

*Corresponding author. *Email addresses:* `{Pierre.Derian, Patrick.Heas, Cedric.Herzet, Etienne.Memin}@inria.fr`

of turbulent fluids, this simple prior turns out to be inadequate. Second-order regularizers [5, 16, 17], or a first order regularizer [7] allowing to enforce physically-sound properties of the flow, are considered; but their implementation raises up several issues.

In this paper, we propose an optical flow estimation procedure based on a wavelet expansion of the velocity field. This approach turns out to offer a nice mathematical framework for multi-resolution estimation algorithms, which avoids some of the drawbacks of the usual approach. Note that algorithms based on wavelet expansion of the data [1], of the velocity field [15] or even both [4] have been previously proposed. However, unlike the algorithm presented hereafter, their computational complexity and/or lack of multiscale mechanism significantly limits their application to small images and/or the estimation of the coarsest motion scales in [15], and might raise up issues when dealing with large displacements in [4]. Finally, the multi-scale wavelet framework also suggests a very simple regularization by neglecting smallest scales coefficients; it turns out to be particularly adapted to "smooth enough" motions.

This article processes as follows: Sec. 2 recalls concepts behind optical flow estimation. Sections 3 and 4 introduce the wavelet framework, then describe its integration into the optical flow problem and the implementation of the resulting algorithm. Behavior and efficiency of the proposed estimator are finally assessed in Sec. 5 and 6, using both synthetic and real flow visualization images.

2. Optic Flow Problem

The optical flow problem consists in estimating the apparent 2D displacement within a 3D scene depicted by a sequence of images, e.g. obtained from a camera. The time- and space-variations of an observable image quantity, e.g. its brightness, are used to infer the underlying motion. In the following, we denote by $I(\boldsymbol{x}, t)$ the brightness of the image at pixel $\boldsymbol{x} \in \Omega$, with $\Omega \subset \mathbb{R}^2$ the image domain, and at a discrete time $t \in \mathbb{N}$. The optical flow, as a 2D vector field $\boldsymbol{v}(\boldsymbol{x}, t) : \Omega \times \mathbb{N} \mapsto \mathbb{R}^2$, is the projection on the image plane of the actual 3D motion. It is a *dense* field, since it provides one velocity vector per pixel of the input images. The optical flow estimation involves two main aspects: the *data term*, which links the motion $\boldsymbol{v}$ to be estimated to the input data – here, image brightness I –, and a *regularization mechanism* to overcome the ill-posedness of the problem.

2.1. Data Term

Data terms are built upon assumptions on the behavior of the observed image quantity. The most simple and most widely used is a conservation assumption of the image brightness:

$$\frac{dI}{dt}(\boldsymbol{x}, t) = \frac{\partial I}{\partial t}(\boldsymbol{x}, t) + \boldsymbol{v}(\boldsymbol{x}, t) \cdot \nabla I(\boldsymbol{x}, t) = 0\,. \tag{2.1}$$

Its time integration leads to the well-known *displaced frame difference* (DFD). Let us denote by $I_0(\boldsymbol{x}) \triangleq I(\boldsymbol{x}, t)$ and $I_1(\boldsymbol{x}) \triangleq I(\boldsymbol{x}, t+1)$ the brightness of two consecutive images from the sequence. Under stable lightning conditions and the hypothesis of conservative transport of the image brightness by a velocity field $\boldsymbol{v}(t) = \boldsymbol{v} = (v_1, v_2)^T$ constant in the time range between the two consecutive images I_0, I_1, the velocity field $\boldsymbol{v}$ satisfies the DFD equation:

$$\forall \boldsymbol{x} \in \Omega, \quad I_0(\boldsymbol{x}) - I_1(\boldsymbol{x} + \boldsymbol{v}(\boldsymbol{x})) = 0\,. \tag{2.2}$$

The estimated optical flow is then obtained via a minimization problem of the form

$$\hat{\boldsymbol{v}} = \arg\min_{\boldsymbol{v}} J_{\text{DFD}}(I_0, I_1, \boldsymbol{v})\,, \tag{2.3}$$

where the functional J_{DFD} may write:

$$J_{\text{DFD}}(I_0, I_1, \boldsymbol{v}) = \frac{1}{2}\int_\Omega \left[I_0(\boldsymbol{x}) - I_1(\boldsymbol{x} + \boldsymbol{v}(\boldsymbol{x}))\right]^2 d\boldsymbol{x}\,. \tag{2.4}$$

Alternatively to quadratic penalization, robust functions (so-called M-estimators) can be used to penalize discrepancies with respect to the brightness conservation model [2].

2.2. Regularization Mechanisms

The estimation problem defined by (2.3) is under-constrained, as there are twice much unknowns (two velocity components) as equations (at each pixel). A common approach to close the problem consists in enforcing some spatial coherence to the solution.

Explicit regularization consists in adding another term to the functional (2.4), balanced by a parameter μ. The estimation problem then writes

$$\hat{\boldsymbol{v}} = \arg\min_{\boldsymbol{v}} J_{\text{DFD}}(I_0, I_1, \boldsymbol{v}) + \mu J_{\text{reg}}(\boldsymbol{v})\,. \tag{2.5}$$

A first-order regularizer, encouraging weak spatial gradients for both components v_1 and v_2, is often used since its introduction by Horn & Schunck [9]:

$$J_{\text{reg}}(\boldsymbol{v}) = \frac{1}{2}\int_\Omega |\nabla v_1(\boldsymbol{x})|^2 + |\nabla v_2(\boldsymbol{x})|^2 \, d\boldsymbol{x}\,. \tag{2.6}$$

In particular, it constitutes a relevant regularization model for rigid motions.

Implicit regularization aims at reducing the number of unknowns by adopting a parametric formulation for the optical flow motion: $\boldsymbol{v}(\boldsymbol{x}) \triangleq \Phi(\boldsymbol{x}, \Theta)$, where Φ is a function parametrized by Θ. Estimation of the motion field $\boldsymbol{v}$ is then replaced by the estimation of its parameters Θ, and the use of a low-order parametric representation (such as piece-wise polynomials) drastically reduces the number of unknowns. The solution then writes

$$\hat{\boldsymbol{v}} = \Phi\left(\arg\min_{\Theta} J_{\text{DFD}}\left(I_0, I_1, \Phi(\Theta)\right)\right)\,. \tag{2.7}$$

2.3. Known Drawbacks or Limitations

Previously introduced data model functional (2.4) is non-linear w.r.t. the velocity field $\boldsymbol{v}$, which in particular makes difficult the problem of large displacements estimation. Therefore estimation of optical flow requires a specific optimization approach. A standard method to tackle non-linearity is to rely on an incremental multi-resolution strategy. This approach consists in choosing some sufficiently coarse low-pass-filtered version of the images at which the linearity assumption is valid, and to estimate a first displacement field assumed to correspond to a coarse representation of the motion. Then, a so-called Gauss-Newton strategy is used by applying successive linearizations around the current estimate, together with a warping procedure accordingly to a hierarchical image representation of increasing resolution. More explicitly, let us introduce the following incremental decomposition of the displacement field at resolution[†] 2^j:

$$\boldsymbol{v}_j = \tilde{\boldsymbol{v}}_j + \boldsymbol{v}'_j \tag{2.8}$$

where $\boldsymbol{v}'_j$ represents the unknown incremental displacement field at resolution 2^j and $\tilde{\boldsymbol{v}}_j \triangleq \sum_{i<j} \mathcal{P}_j(\boldsymbol{v}'_i)$ is a coarse motion estimate computed at the previous scales; $\mathcal{P}_j(\boldsymbol{v}'_i)$ denotes a projection operator which projects $\boldsymbol{v}'_i$ onto the grid considered at resolution 2^j. In order to respect the Shannon sampling theorem, the coarse scale data term is derived by a low-pass filtering of the original images with a kernel[‡] $\mathcal{G}_j$, followed by a subsampling of period 2^j. Using (2.8), at coarse scale, image $I_j(\boldsymbol{x})$ and the *motion-compensated* image $\tilde{I}_j(\boldsymbol{x})$ are then defined as:

$$\begin{cases} I_j(\boldsymbol{x}) = \downarrow_{2^j} \circ \left(\mathcal{G}_j \star I_0(\boldsymbol{x})\right) \\ \tilde{I}_j(\boldsymbol{x}) = \downarrow_{2^j} \circ \left(\mathcal{G}_j \star I_1(\boldsymbol{x} + \tilde{\boldsymbol{v}}_j(\boldsymbol{x}))\right), \end{cases} \tag{2.9}$$

where $\downarrow_{2^j}$ denotes a 2^j-periodic subsampling operator. It yields a functional J^j_{DFD} defined as a linearized version of (2.4) around $\tilde{\boldsymbol{v}}_j(\boldsymbol{x})$:

$$J^j_{\text{DFD}}(I_j, \boldsymbol{v}'_j) = \frac{1}{2}\int_{\Omega_j} \left[\tilde{I}_j(\boldsymbol{x}) - I_j(\boldsymbol{x}) + \boldsymbol{v}'_j(\boldsymbol{x}) \cdot \nabla \tilde{I}_j(\boldsymbol{x})\right]^2 d\boldsymbol{x}\,. \tag{2.10}$$

Finally, the sought motion estimate $\hat{\boldsymbol{v}}$ is given by solving a system of coupled equations associated to resolutions increasing from 2^C to 2^F:

$$\begin{cases} \hat{\boldsymbol{v}} = \boldsymbol{v}'_F + \tilde{\boldsymbol{v}}_F = \boldsymbol{v}'_F + \sum_{i=C}^{F-1} \mathcal{P}_F(\boldsymbol{v}'_i)\,, \\ \boldsymbol{v}'_j = \arg\min_{\boldsymbol{v}'} J^j_{\text{DFD}}(I_j, \boldsymbol{v}')\,, \forall j \in \{C, \cdots, F\}\,. \end{cases} \tag{2.11}$$

where the finest scale $s = 2^{-F}$ corresponds to the pixel whereas the coarsest scale is noted $s = 2^{-C}$. In practice, equations in (2.11) are usually solved successively, starting

---

[†] In this paper, we shall use the following convention: indices $j \geq 0$ represent the *resolution* 2^j. Corresponding *scale* is 2^{-j}.

[‡] A Gaussian kernel of variance proportional to 2^j is commonly used.

from the coarsest to the finest scale. This coarse-to-fine approach has the drawback of freezing (i.e. leaving unchanged), at a given scale, all the previous coarser estimates. Moreover, the major weakness of this strategy is the arbitrary approximation of the original functional in (2.4) by a set of coarse scale data terms (2.10), which are defined at different resolutions by a modification of original input images with (2.9) and by a linearization of model (2.2) around the previous motion estimate. Authors in [3] have nevertheless successfully given a theoretical formulation of the warping strategy. In the next sections, we will see that the framework of wavelet representations also offers an mathematically-sound formulation, as already investigated in [15].

3. Wavelet Formulation

The wavelet framework introduces a decomposition of a signal into several sets of *details* at various scales and a remaining *coarse approximation.* The projection of the two components v_1, v_2 of the velocity field $\boldsymbol{v}$ onto such *multi-resolution approximation spaces* proves to be efficient within the context of optical flow estimation, enabling to handle large displacements while providing a simple implicit closure to the problem (Sec. 2.2). Let us first recall the main characteristics of the wavelet formalism for real scalar signals.

3.1. Wavelet Formalism

Multiresolution approximations We consider a multi-resolution approximation of $\mathbf{L}^2(\mathbb{R})$ a sequence $\{V_j\}_{j\in\mathbb{Z}}$ of closed subspaces, so-called *approximation spaces*, notably verifying[§]

$$\begin{aligned}
&V_j \subset V_{j+1}\,;\\
&\lim_{j\to-\infty} V_j = \bigcap_{j=-\infty}^{+\infty} V_j = \{0\}\,;\\
&\lim_{j\to+\infty} V_j = \text{Closure}\left(\bigcup_{j=-\infty}^{+\infty} V_j\right) = \mathbf{L}^2(\mathbb{R})\,.
\end{aligned}$$

Since approximation spaces are sequentially included within each other, they can be decomposed as $V_{j+1} = V_j \oplus W_j$. Those W_j are the orthogonal complements of approximation spaces, they are called *detail spaces.*

In practice, signals have a finite number of samples. We consider those signals to be defined on a discretization of unit segment $[0, 1]$; the resolution of this discretization increasing with the number of samples. Let $w[k]$ be an 2^F-sample approximation, i.e. $\in V_F$, of signal $w(x)$:

$$w[k] = w(\frac{k}{2^F}) = w(x_k)\,,\ \forall\, 0 \le k < 2^F\,. \tag{3.1}$$

[§] See [12] for a complete presentation of wavelet bases.

Then, applying recursively the decomposition of approximation spaces V_j,

$$w \in V_F = V_C \oplus W_C \oplus W_{C+1} \oplus \cdots \oplus W_{F-1} \subset \mathbf{L}^2([0,1])\,, \tag{3.2}$$

where $C \in [0; \cdots; F-1]$ denotes the coarsest scale considered, and F is the finest.

Wavelet bases The projection of w onto the set of spaces in (3.2) writes:

$$w(x) = \sum_{k=0}^{2^C-1} \langle w, \phi_{C,k} \rangle_{\mathbf{L}^2}\, \phi_{C,k}(x) + \sum_{j=C}^{F-1} \sum_{k=0}^{2^j-1} \langle w, \psi_{j,k} \rangle_{\mathbf{L}^2}\, \psi_{j,k}(x)\,,\ \forall\, x \in [0,1]\,. \tag{3.3}$$

Here, $\{\phi_{C,k}\}_k$ and $\{\psi_{j,k}\}_k$ are orthonormal bases of V_C and W_j, respectively. They are defined by *dilations* and *translations*[¶] of the so-called *scale function* ϕ and its associated *wavelet function* ψ. The representation of a signal projected onto the multi-scale wavelet basis is given by the set of coefficients appearing in (3.3): $a_{C,k} \triangleq \langle w, \phi_{C,k} \rangle_{\mathbf{L}^2}$ and $d_{j,k} \triangleq \langle w, \psi_{j,k} \rangle_{\mathbf{L}^2}$ are approximation and detail coefficients, respectively.

Those results are extended to the case of 2D signals, in order to obtain *separable multi-scale orthonormal bases* of $\mathbf{L}^2([0,1]^2)$. With the *isotropic* 2D wavelet transform, basis functions are obtained by dilations and (2D) translations of:

$$\begin{aligned} \phi(\boldsymbol{x}) &\triangleq \phi(x_1)\phi(x_2); & \psi^1(\boldsymbol{x}) &\triangleq \phi(x_1)\psi(x_2); \\ \psi^2(\boldsymbol{x}) &\triangleq \psi(x_1)\phi(x_2); & \psi^3(\boldsymbol{x}) &\triangleq \psi(x_1)\psi(x_2)\,. \end{aligned} \tag{3.4}$$

3.2. Wavelet Data Term

Each scalar component v_i of the velocity field is projected onto the multi-scale wavelet basis. We denote by Θ_i the set off all coefficients – both approximation and details – related to component v_i, and $\boldsymbol{\Theta}^T = \left(\Theta_1^T, \Theta_2^T\right)$ the superset of all coefficients related to motion $\boldsymbol{v}$. The reconstruction operation in (3.3) – from the wavelet basis to the canonical basis – is linear, we denote by Φ its operator and write:

$$\boldsymbol{v}(\boldsymbol{x}) = \Phi(\boldsymbol{x})\boldsymbol{\Theta}\,. \tag{3.5}$$

One may already recognize the parametric form introduced in Sec. 2.2, although up to now the number of unknowns does not change: there are as much coefficients in $\boldsymbol{\Theta}$ as scalar values in $\boldsymbol{v}$. From there, we insert (3.5) into DFD equation (2.2), and the optical flow estimation problem finally writes:

$$\begin{cases} \hat{\boldsymbol{v}} = \Phi\hat{\boldsymbol{\Theta}} \in V_F \\ \hat{\boldsymbol{\Theta}} = \arg\min\limits_{\boldsymbol{\Theta}} J_{\text{DFD}}(I_0, I_1, \boldsymbol{\Theta})\,. \end{cases} \tag{3.6}$$

[¶] Written in a general form $f_{j,k}(x) = 2^{j/2} f(2^j x - k)$.

3.3. Implicit Regularization

In order to close the minimization problem, the number of unknowns has to be lowered. A very simple way to achieve this reduction consists in formulating the motion estimation problem on a *truncated wavelet basis*, i.e. by neglecting coefficients corresponding to smallest scales. As a consequence, the solution $\hat{\boldsymbol{v}}$ belongs to a lower-resolution space $V_L \subset V_F$. Details coefficients corresponding to the neglected detail scales (W_j, with $L \leq j \leq F-1$) are set to zero. Writing $\boldsymbol{\Theta}_{|j}$ to describe the unknown coefficients vector $\boldsymbol{\Theta}$ having all finer scales than j cancelled, so that $\boldsymbol{v} = \Phi \boldsymbol{\Theta}_{|j} \in V_j \subset V_F$, the now regularized optical flow problem reads:

$$\begin{cases} \hat{\boldsymbol{v}} = \Phi \hat{\boldsymbol{\Theta}}_{|L} \in V_L \,, \ L < F \\ \hat{\boldsymbol{\Theta}}_{|L} = \underset{\boldsymbol{\Theta}_{|L}}{\arg\min}\, J_{\text{DFD}}(I_0, I_1, \boldsymbol{\Theta}_{|L})\,. \end{cases} \tag{3.7}$$

From the dyadic structure of the wavelet decomposition, the number of coefficients is multiplied by 4 whenever a new detail scale is added. As a consequence, it is theoretically possible to estimate coefficients up to the penultimate small scale (Sec. 2.2), i.e. $\boldsymbol{v} \in V_{F-1}$. In practice, this remains often impossible due to underdeterminations in regions of poor luminance contrast, or in areas where photometric gradients are aligned. As a consequence, a lower value must be imposed. Obviously, adaptive strategies relying on some sparsity constraint could be proposed; this requires however to setup schemes allowing to fix locally the corresponding scales. This will not be explored in this study, so we stick here to a global thresholding methodology. Finally, a full-scale estimation ($\boldsymbol{v} \in V_F$) requires the use of explicit smoothing terms, as previously mentioned in Sec. 2.2. The design of high-order regularization schemes for the proposed wavelet-based estimator has been investigated in [10] and led to promising results; its description is however out of the scope of this paper.

3.4. Properties of the Solution

The choice of the wavelet basis is of major importance, especially since smallest scales are neglected. Indeed, the regularity of the solution, as well as the quantity of energy "lost" from small scales cancellation, may highly depends on the wavelet basis, through the number of *vanishing moments*.

Vanishing moments The notion of vanishing moments (VM) simply reflects the orthogonality of a given wavelet function to polynomials up to a certain order:

$$\psi \text{ has } n \text{ VM } \Leftrightarrow \langle \psi, x^p \rangle_{\mathbf{L}^2} = \int \psi(x) x^p dx = 0\,,\ \forall\, 0 \leq p < n\,. \tag{3.8}$$

The number of VM is linked to the size of the support of basis functions ϕ, ψ: the higher the number of VM, the wider the support. Hence it is also related to the wavelet basis ability to cope with the previous underdetermination problem (referred to as the "aperture problem" in optical flow literature).

3.4.1. Polynomial approximations

From (3.8), a wavelet with n VM is hence orthogonal to any polynomial of degree $n-1$. Consequently, piecewise[‖] polynomials of degree $n-1$ belonging to V_F are exactly described in V_{F-1}, since the elements of the basis that belong to its orthogonal complement W_{F-1} have vanishing coefficients. Therefore $\hat{\boldsymbol{v}} \in V_L$, solution of (3.7), is a piecewise polynomial of order $n-1$ in V_{L+1} over the support of scaling functions.

3.4.2. Truncation error

Because of the energy conservation provided by the wavelet transform, truncating small scales coefficients certainly introduces an error. Using the Lipschitz regularity of the estimated motion, it is possible to obtain a bound for the number of VM, above which the truncation error no longer depends on the number of VM, but on the motion regularity only.

Lipschitz regularity The Lipschitz regularity gives a measure of the local regularity of a given signal. A function $w(t)$ is *pointwise-Lipschitz* $\alpha \geq 0$ at t_0 if there exists a local polynomial $P_{t_0}(t)$ of degree $n = \lfloor \alpha \rfloor$ and K constant such that

$$|w(t) - P_{t_0}(t)| \leq K|t - t_0|^{\alpha} \tag{3.9}$$

It is then *uniformly-Lipschitz* α over $[a, b]$ if it satisfies (3.9) for any $t_0 \in [a, b]$, with K independent of t_0.

Coefficients decay and vanishing moments The uniform Lipschitz regularity of a signal can be related to the decay across scales of its wavelet coefficients amplitude. Let us consider a signal $w \in \mathbf{L}^2([0, 1])$, uniformly Lipschitz α over $[0, 1]$, and its projection on a wavelet basis. The wavelet has n vanishing moments and is $\mathcal{C}^n$ with fast decay derivatives. At fine scales,

- if $n < \alpha$, the decay of coefficient amplitude depends on n:

$$\left| \langle w, \psi_p^j \rangle_{\mathbf{L}^2} \right| = |d_{j,p}| \sim 2^{-j(n+1/2)} \, ;$$

- if $n \geq \alpha$, it depends on α [12]:

$$\exists A > 0 \text{ such that } |d_{j,p}| \leq A 2^{-j(\alpha+1/2)} \, .$$

[‖] On the support of $\{\phi_{F-1,k}\}$.

Consequences of the truncation The amplitude of truncated small-scales coefficients depends on either α the signal regularity, or n the number of vanishing moments of the analyzing wavelet. Therefore the amount of energy lost by neglecting small scales will depend either on the wavelet basis, when $n < \alpha$, or on the signal regularity if $n \geq \alpha$.

Consequently, using a "high enough" number of vanishing moments should ensure that the amount of neglected energy does not depend on the wavelet basis. The interest of this simple rule however has to be balanced, since the increase of VM also results in a significant increase of the computational burden, due to the wider support of basis functions.

4. Implementation

4.1. Minimization

Multi-scale estimation Coefficients are estimated sequentially from the coarsest scale C to the chosen finest one $L < F$, using a gradient-descent algorithm for the minimization of the functional. At current scale j, $C \leq j \leq L$, the unknown vector $\boldsymbol{\Theta}_{|j}$ includes all coefficients from coarse scale C to j, coefficients estimated at previous coarser scales (C to $j-1$) being used as the initialization point of the gradient descent. This strategy enables to update (to correct) those coarser coefficients while estimating "new" details at current scale j. In other words, the solution is sequentially sought within higher resolution spaces $V_C \subset V_{C+1} \subset \cdots \subset V_L$, which recalls the subspace correction methods investigated in [14]. This way, the projection of the current solution $\hat{\boldsymbol{v}} \in V_j$ onto every coarser space V_p with $C \leq p < j$ is constantly updated, contrary to the standard incremental approach.

Gradient-descent algorithm At each refinement level j, minimization of functional J_{DFD} is efficiently achieved with a gradient-based quasi-Newton algorithm (l-BFGS) [13], to seek the optimum $\hat{\boldsymbol{\Theta}}_{|j}$. For any coefficient $\theta_{i,p} \in \Theta_i \subset \boldsymbol{\Theta}$, it is straightforward to show that

$$\frac{\partial J_{\mathrm{DFD}}}{\partial \theta_{i,p}}(I_0, I_1, \boldsymbol{\Theta}) = \left\langle \frac{\partial I_1}{\partial x_i}(\cdot + \Phi(\cdot)\boldsymbol{\Theta}) \left[I_1(\cdot + \Phi(\cdot)\boldsymbol{\Theta}) - I_0(\cdot) \right], \Phi_p \right\rangle_{\mathbf{L}^2([0,1]^2)} \tag{4.1}$$

where Φ_p is the wavelet basis atom related to $\theta_{i,p}$. As a consequence, components of the spatial gradient of the data-term functional are simply given by the coefficients resulting from the projection on the considered wavelet basis of the two following terms:

$$\frac{\partial I_1}{\partial x_i}(\boldsymbol{x} + \Phi(\boldsymbol{x})\boldsymbol{\Theta}) \left[I_1(\boldsymbol{x} + \Phi(\boldsymbol{x})\boldsymbol{\Theta}) - I_0(\boldsymbol{x}) \right], \quad i = 1, 2\,.$$

Note that conversely to the algorithm proposed in [15] which involves the Hessian matrix to minimize an incremental quadratic error, here only the gradient is required; its computation is efficiently achieved via the fast wavelet transform presented hereafter. Therefore, the overall reasonable complexity of this algorithm does not restrict motion estimation to the very coarsest scales and/or to images of small size.

4.2. Fast Wavelet Transform

Evaluation of J_{DFD} gradient and value requires the computation of motion-compensated image $I_1(\boldsymbol{x} + \boldsymbol{v}(\boldsymbol{x}))$. Therefore each evaluation involves:

- two *inverse* wavelet transforms (one per scalar component), from $\boldsymbol{\Theta}$ to $\boldsymbol{v}$, in order to compute $I_1(\boldsymbol{x} + \boldsymbol{v}(\boldsymbol{x}))$;
- two *forward* wavelet transforms to compute gradient components, from (4.1).

The fast wavelet transform (FWT) uses the two-scales relation verified by scaling and wavelet functions:

$$\phi(x) = \sqrt{2} \sum_{k \in \mathbb{Z}} h[k] \phi(2x - k) \; ; \; \psi(x) = \sqrt{2} \sum_{k \in \mathbb{Z}} g[k] \phi(2x - k) \, . \tag{4.2}$$

Sequences $h[k] = \langle \phi(x), \sqrt{2}\phi(2x - k) \rangle$ and $g[k] = \langle \psi(x), \sqrt{2}\phi(2x - k) \rangle$ are called *conjugate mirror filters*. Those filters, as well as their time-reverses[**] $\bar{h}$ and $\bar{g}$, enable the fast implementation through *filter banks* of forward and inverse wavelet transforms using decimation, expansion and (circular) convolution operations. For a given 2D signal of $N = 2^F \times 2^F$ pixels and filters h, g of length K, forward and inverse transforms are computed with fewer than $\frac{8}{3}KN$ operations – App. A.

Optimizing wavelet transform From the implicit regularization introduced in Sec. 3.3 and the sequential estimation process in 4.1, we see that most of wavelet transforms (to reconstruct $\boldsymbol{v}$ or to compute gradient) involve several fine scales with null coefficients. A very simple optimization consists in using "smart" filter banks which do not take null-coefficients scales into account, thus saving 62.5% of operations at each step – see App. A. Figure 1 shows the typical gain obtained thanks to the use of this modified filter bank.

4.3. Pseudo-Code

Pseudo-code of the sequential estimation is presented Fig. 2. Contrary to local approaches (patches) such as the well-known Lucas and Kanade [11], motion is searched globally so the process can not be parallelized. However, the convolution-based wavelet transforms makes it likely to be suitable for a GPU implementation. Furthermore, as each pair is processed independently, long image sequences can be processed efficiently on multicore computers. At the moment, this algorithm has been implemented in C++, using `libLBFGS` [13] for the optimization routines, `CImg` for the image processing aspects and a custom wavelet library. Much low-level optimization is still to be done; computation time for a single image pair is of order $5 - 30s$ for 256×256px images of Sec. 5.

[**] More explicitly, $\bar{f}[n] = f[-n]$.

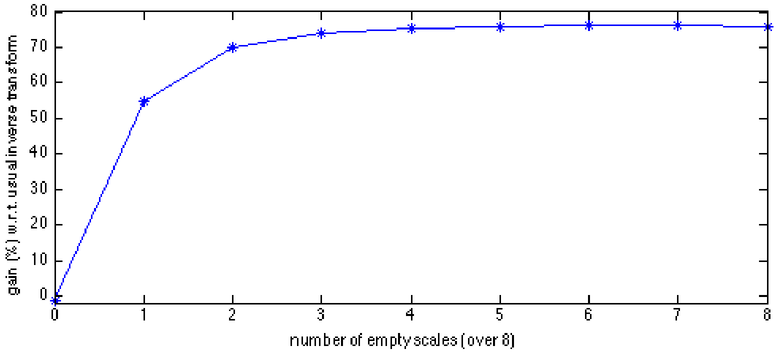


Figure 1: Typical gain observed by the use of a modified filter bank, to save computations by excluding null-coefficients scales. When one scale only is empty, the observed gain is $\simeq 55\%$, below the theoretical 62.5%.

```
//——— parameters
$I_0$, $I_1$; //input images of size $2^F \times 2^F$
L; //motion finest estimated scale, $0 \le L < F$
C; //motion coarsest scale considered, $0 \le C \le L$

//——— loop over scales
for (s=C; s<=L; s++){
    $\mathbf{\Theta}_{|s,0}$ := (s>C)? $\hat{\mathbf{\Theta}}_{|s-1}$ : 0; // starting point
    $\hat{\mathbf{\Theta}}_{|s}$ := l−BFGS_min($J_{\text{DFD}}(I_0, I_1, \mathbf{\Theta}_{|s})$, $\mathbf{\Theta}_{|s,0}$);
}
return $\Phi\hat{\mathbf{\Theta}}_{|L}$;
```

Figure 2: Sequential Estimation Pseudocode

5. Validation with Synthetic Data

Influence of parameters L (*finest estimated motion scale*), C (*coarsest scale considered*) and n (*number of vanishing moments*) on the estimated motion are characterized by several experiments in Sec. 5.2, 5.3 and 5.4. The choice of the *wavelet family* is then discussed in Sec. 5.5. Finally two image sequences of 100 frames long are processed. Results are compared to other state-of-the-art estimators.

5.1. Input Data

This algorithm has been originally designed to extract motions from sequences of fluid flow images such as satellite and lidar imagery or fluid flow visualization techniques, e.g. particle imagery velocimetry (PIV), Schlieren photography or laser-induced fluorescence (LIF). The following experiments investigate the capacities of the proposed algorithm, when applied to images resulting from two characteristic types of flow visualization methods: flows seeded with particles (hereafter referred to as "particle images" or as PIV) and passive scalar advection (called "scalar images"). Particle images are highly textured (see Fig. 3a for a synthetic image or Fig. 10b for a real one), hence particularly well suited to the optical flow computation. On the contrary, scalar advection-diffusion pictures (synthetic example Fig. 3c) present much more almost-constant-value areas. Besides, the brightness conservation assumption (2.1) is not respected due to the diffusion process; this makes scalar images far more difficult to process.

The first data set used for evaluation is a synthetic sequence of PIV images of size 256×256 pixels (i.e. finest scale available F=8), representing small particles (of radius smaller than 4 pixels) advected by a forced 2D turbulent flow with periodic boundary conditions. The dynamic of the fluid flow is supplied by a numerical simulation of 2D Navier-Stokes equations at $Re = 3000$, using the vorticity-velocity formulation and the Lagrangian equation for non-heavy particles transported by the flow (simulation details can be found in [8]). This simulated flow has a null-divergence by construction. It features relatively small displacements, with a maximum magnitude of 3.5 pixels. Most of the following experiments used the two first frames of the sequence; the whole 100-frame sequence being processed at last. First image I_0 is displayed in Fig. 3a. The underlying ground-truth flow motion, that is supposed to be recovered from optical flow estimation, will be referred to as $\boldsymbol{v}_{\text{ref}}$ hereafter; its *vorticity* (i.e. the curl of the velocity field) is displayed in Fig. 3b. Estimated velocity field evaluation is based on the Root Mean Square end-point-norm Error (RMSE).

The second dataset consists in a synthetic sequence of images depicting the advection-diffusion of a passive scalar transported by the *same 2D flow* described above. A sample frame is presented Fig. 3c.

5.2. Influence of the Truncation

The finest estimated scale parameter L fixes the approximation space $V_L \in V_F$ in which the solution is sought. Its choice is a tradeoff between the need to reduce the number of unknowns, in order to close the estimation problem, and the will to estimate the finest scales of the motion. Figure 4 presents a 2D-plot of RMSE values as a function of the number of VM and the finest scale L. For this benchmark, it appears that minimum of RMSE are obtained in the region corresponding to L=5 and 6. As expected from remarks in Sec. 3.3, best results are obtained with $L = 6 = F - 2$, i.e. $\hat{\boldsymbol{v}} \in V_6$ (and a number of VM $n > 4$, the influence of which being discussed in Sec. 5.4). This corresponds to a number of unknowns reduced by 87.5%.

A similar analysis carried out on scalar images gives $L = 5$ as the optimal value of this parameter. As previously mentioned, scalar diffusion images have much more low-gradient areas (Fig. 3c). Uncertainties due to these low gradients arise sooner than with particle images (as we proceed towards finer scales), therefore it is not surprising to find that the optimum solution space is coarser with scalar images than with particles.

5.3. Importance of Multiscale Bases

Having chosen the fine scale parameter L, we may now wonder whether the various equivalent bases of space V_L lead to identical solutions. Indeed, thank to wavelet formalism (3.2), several multiscale decompositions of V_L may be considered:

$$\begin{aligned} V_L &= V_{L-1} \oplus W_{L-1} \\ &= V_{L-2} \oplus W_{L-2} \oplus W_{L-1} \\ &= \cdots \\ &= V_C \oplus W_C \oplus \cdots \oplus W_{L-1} \\ &= V_0 \oplus W_0 \oplus W1 \oplus \cdots \oplus W_{L-1}. \end{aligned} \tag{5.1}$$

Parameter C (coarsest scale considered) fixes the number of scales comprised within coarse approximation space V_C, and the number of remaining detail scales $(L-C)$. The closer is C to L, the lesser is the number of detail scales to be estimated sequentially (algorithm Fig. 2). This choice has two main consequences:

(i) with C close to L, there are less steps in the sequential optimization process. Moreover, forward and inverse wavelet transforms require less operations, resulting in an overall much faster process;

(ii) however, in the presence of strong non-linearities, in particular for large displacement, a high value of C will result in the impossibility to capture those high-amplitude motions.

This is studied by adding a mean displacement of magnitude $\simeq 8.5$ pixels to the reference motion $\boldsymbol{v}_{\text{ref}}$ presented above (Sec. 5.1). This large-magnitude motion will be referred to as $\tilde{\boldsymbol{v}}_{\text{ref}}$ hereafter. Figure 5 shows plots of RMSE as a function of parameters (L, C) for both motions estimated on a 5 VM wavelet basis, according to observations from Sec. 5.2.

Experiments confirm previous results (Sec. 5.2): optimum is obtained for $L = 6$ for both cases. Regarding $\boldsymbol{v}_{\text{ref}}$ estimation (reasonably small displacements), estimated motions $\hat{\boldsymbol{v}}$ are identical for any $C \in [0; 5]$. From above remark (i), the best choice of parameters is therefore $(L, C) = (6, 5)$, i.e. a coarse approximation and 1 scale of details only. However, those same parameters used for $\tilde{\boldsymbol{v}}_{\text{ref}}$ estimation (large displacements) result in a complete failure, which confirms remark (ii). Here the available range is $C \in [0, 2]$, the optimum choice being $(L, C) = (6, 2)$, i.e. 4 detail scales considered. This lower value of C results in a twice longer computation time.

These results suggest that a sequential estimation on a multiscale basis is always preferable to a single-step estimation on the equivalent (monoscale) basis. Indeed estimating $\boldsymbol{v}_{\text{ref}}$ directly in V_6 (parameters $(6, 6)$) fails, but choosing instead a 2-step estimation in $V_5 \oplus W_5 = V_6$ (parameters $(6, 5)$) leads to proper convergence. Moreover, the choice of coarse approximation space V_C strongly influences the success of the estimation process. With larger displacements, it is necessary to consider low values for C: basis functions with a larger support are required to capture the high-amplitude motions.

5.4. Influence of the Number of Vanishing Moments

From remarks in Sec. 3.4, the number of vanishing moments influences the regularity of the solution as well as the reconstruction error and the computation time.

Figure 6 takes estimated motions $\hat{\boldsymbol{v}} \in V_6$ and projections of $\boldsymbol{v}_{\text{ref}}$ onto V_6: $\boldsymbol{v}_{\text{ref}|6} \triangleq P_{V_6}(\boldsymbol{v}_{\text{ref}})$ (by canceling small scales coefficients in W_6 and W_7), obtained using different numbers of VM. Those two motions are compared to the ground truth motion $\boldsymbol{v}_{\text{ref}}$, in terms of RMSE and kinetic energy. RMSE values (Fig. 6a) rapidly decrease as the number of VM increases, converging towards 0.089 and 0.014 pixels for $\hat{\boldsymbol{v}}$ and $\boldsymbol{v}_{\text{ref}|6}$, resp. Regarding the kinetic energy (Fig. 6b), the percentage w.r.t. $\boldsymbol{v}_{\text{ref}}$ of kinetic energy contained by motions $\hat{\boldsymbol{v}}$ and $\boldsymbol{v}_{\text{ref}|6}$ also rapidly converges towards 98.5% and 99.98%, resp.

Those results indicate that $\boldsymbol{v}_{\text{ref}|6}$ is a "reasonably good" approximation of $\boldsymbol{v}_{\text{ref}}$, using a wavelet with $n > 4$ VM, since in that case the two finest scales that were cancelled by the projection contain only $\simeq 0.02\%$ of the overall kinetic energy. However, it appears that our estimations $\hat{\boldsymbol{v}}$ never manage to reach the quality of $\boldsymbol{v}_{\text{ref}|6}$, either in terms of RMSE or energy. This remaining gap can be explained by looking at maps of end-point-norm error obtained for $n = 1$ and $n = 20$ VM – Fig. 7. Only small structures, tube-shaped, are not estimated using $n = 20$. This is notably due to a lack of information at small scales in the input images: the smallest motion structures are not necessary "visible" (detectable) from particles displacement. Those small structures might need more elaborate, physically-sound regularization terms to be recovered.

Figure 8 shows the evolution of the computation time as a function of the number of VM n, for couples $(L, C) = (6, 5)$ and $(6, 0)$ and using either usual or "smart" filter banks (Sec. 4.2). As already pointed out in Sec. 5.3 - (i), picking $C = 5$ instead of $C = 0$ results in a significative drop of the computational burden ($\sim 60\%$ decrease) . Reducing n by 1 has a much lower impact ($\sim 5\%$ decrease). The use of smart filter banks also contributes to reduce the computational burden, from a $\sim 10\%$ decrease at small VM up to 50% for higher values.

5.5. Influence of the Wavelet Family

Several orthogonal wavelet families have been implemented for this study:

(i) Daubechies wavelets, known for having the shortest compact support for a given number of VM (less computations);

(ii) Coiflets, which feature a compact support as well as interpolating scaling functions;

(iii) Battle-Lemarié wavelets, polynomial splines with fast decay but no compact support.

Previous experiments used Daubechies wavelets, since filters are available for a wide range of VM. Tab. 1 shows RMSE obtained using those three wavelet families, for

$(L, C) = (6, 0)$ and VM $n = 2, 4, 6$. The improvement brought by the use of more elaborate bases is somewhat negligible ($\simeq 2\%$), regarding especially the computation time which increases very quickly due to much larger supports (e.g. +55% with Battle-Lemarié instead of Daubechies wavelets, $n = 6$ VM).

Family \ VM	2	4	6
Daubechies	0.11	0.091	0.090
Coiflets	0.096	0.091	0.089
Battle-Lemarié	0.092	0.088	0.088

Table 1: RMSE obtained using different wavelet families, for $n = 2, 4, 6$ VM.

5.6. Comparison to State-of-the-Art

A total of 100 frames from both sequences of particle (Fig. 3a) and scalar advection-diffusion (Fig. 3c) images have been processed. Figure 9 compares results, in terms of RMSE, of the proposed wavelet-based estimator using parameters $(L, C) = (6, 5)$ and () and $n = 20$, to those of several state-of-the-art estimators. Results are projected to the null-divergence space when necessary, in order to compare all methods on the same basis; this applies as well to our estimator. Regarding the particle image sequence, our algorithm clearly outperforms most of other estimators and compete on par with [7] which use a far more complex self-similarity prior. However, results of the scalar advection images are worse than those obtained by any other estimator, except the correlation-based method. Scalar advection pictures are much less textured than particle ones, thus more difficult to process due to the aperture problem. More elaborate regularization mechanisms, using for instance the aforementioned prior on self-similarity [7], on the power spectrum [6] or exact high-order regularization [10] are required in order to obtain more acceptable estimates.

6. Results on Real Images

Performances of the algorithm are then assessed on actual PIV images, which were acquired and kindly provided by D. Heitz and A. Guibert, from IRSTEA Rennes. The sequence of 1024 × 1024 pixels frames depicts the classical case of a cylinder wake at Reynolds number $Re = 3900$. Configuration of the experiment and a sample frame subregion are shown Fig. 10. The whole sequence has been processed; several estimated motions corresponding to a single image pair are compared in Fig. 11. No ground truth is available here. In order to highlight the differences between the different solutions, estimated motions are compared on the basis of their vorticity. This differential quantity enlightens the structures of the flow and also emphasizes its local variations. Several estimates with $n = 10$ VM and decreasing parameter L are presented, as well as an estimate obtained from a similar yet more elaborate wavelet-based estimator featuring explicit second-order smoothing terms [10], and a last estimate given by correlations method. With $L = 5$ (Fig. 11a), motion structures look too coarse when compared to the estimate with explicit smoothing (Fig. 11e). However we can see this solution corresponds quite well to the motion provided by the correlations technique (Fig. 11f), but with much more regularity. Setting $L = 6$ enables to retrieve more structures than the correlations, yet structures still look too coarse. Increasing further the number of estimated scales ($L = 7$, 8) enables to recover some finer structures, however the vorticity now presents an increasing noisy aspect. It corresponds to local variations of the motion field that are consequences of poorly-estimated fine scale coefficients, due to the ambiguities arising from the aperture problem. Consequently, taking the vorticity aspect as a quality criterion, the use of an explicit smoothing term (Fig. 11e) is mandatory in order to properly estimate finest scales. Here, the best compromise with our simple estimator is probably $L = 6$. It is remarkable to note that this estimate has exactly the same number of unknowns as the correlation results (Fig. 11f): 2×4096, i.e. 0.4% of the total unknowns for a full fine-scale estimation. Yet structures are better shaped and the whole vorticity field is not too noisy.

This sequence is made of 3072 frame pairs in total. Processing such a large amount of data takes a long time, hence it might be advantageous to optimize parameter C prior to start working on the whole sequence. How to proceed, when no ground truth is available? Having fixed fine scale parameter $L = 5$, we may solve, for a single image pair, the optical flow problem for every coarse scale parameter $C \in [L; 0]$. Then, we compare these various solutions between each other. Under the hypothesis that there exists a range for parameter C for which the solution is identical, the upper bound of this range gives as a rule of thumb the optimal parameter in terms of computing time. Indeed, experiments on a single image pair give identical estimates for $C \in [0, 3]$. Hence $C = 3$ is the optimal parameter, since it minimizes the number of steps and the cost of wavelet transforms: computing time drops by 58% with respect to $C = 0$ solution.

Conclusions

We introduced an optical flow estimation algorithm based on the wavelet expansion of the velocity field. The multi-scale framework brought by the wavelet formalism enables to design a simple multi-scale motion estimation algorithm, without requiring to resort to the standard incremental approach. A simple closure is provided by neglecting small scales coefficients, the truncation error being independent of the number of vanishing moments. The use of a gradient-based optimization method avoids the cumbersome computation of the Hessian matrix, hence enabling to process large images up to relatively fine scales. Evaluation of the gradient is done through forward wavelet transforms, the later being optimized to reduce computations. The proposed method is evaluated on fluid flows image sequences; it proves to be surprisingly efficient on particle images, but fails to process less textured scalar-advection pictures. This algorithm is however generic enough and should not be restricted to fluid flow images processing; it shall be well-adapted to estimation of any kind of “relatively smooth” motions for which the small-scales truncation is relevant. The main issue remains the choice of the truncation scale; a criterion could probably be derived from the analysis of input image gradients. More complex estimators, allowing a full estimation up to finer scales and featuring explicit high-order regularization schemes based on wavelet coefficients properties, are being investigated [10].

Acknowledgments

The authors acknowledge the support of the French Agence Nationale de la Recherche (ANR), under grant MSDAG (ANR-08-SYSC-014) "Multiscale Data Assimilation for Geophysics".

References

[1] C. Bernard. *Wavelets and ill posed problems: optic flow and scattered data interpolation.* PhD thesis, École Polytechnique, 1999.

[2] M. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, 1996.

[3] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. *Computer Vision-ECCV 2004*, pages 25–36, 2004.

[4] L. Chen, H. Liao, and J. Lin. Wavelet-based optical flow estimation. *Circuits and Systems for Video Technology, IEEE Transactions on*, 12(1):1–12, 2002.

[5] T. Corpetti, E. Mémin, and P. Pérez. Dense estimation of fluid flows. *Pattern Anal Mach Intel*, 24(3):365–380, 2002.

[6] P. Dérian, P. Héas, C. Herzet, and E. Mémin. Wavelets to reconstruct turbulence multifractals from experimental image sequences. In *7th Int. Symp. on Turbulence and Shear Flow Phenomena, TSFP-7*, Ottawa, Canada, July 2011.

[7] P. Héas, E. Mémin, D. Heitz, and P. Mininni. Power laws and inverse motion modelling: application to turbulence measurements from satellite images. *Tellus A*, 64(10962), 2012.

[8] D. Heitz, J. Carlier, and G. Arroyo. Final report on the evaluation of the tasks of the workpackage 2, FLUID project deliverable 5.4. Technical report, INRIA - Cemagref, 2007.

[9] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[10] S. Kadri Harouna, P. Dérian, P. Héas, and E. Memin. Divergence-free Wavelets and High Order Regularization. 2011.

[11] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereovision. In *Int. Joint Conf. on Artificial Intel. (IJCAI)*, pages 674–679, 1981.

[12] S. Mallat. *A Wavelet Tour of Signal Processing: The Sparse Way.* Academic Press, 2008.

[13] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer Series in Operations Research. Springer-Verlag, New York, NY, 1999.

[14] X. Tai and J. Xu. Global and uniform convergence of subspace correction methods for some convex optimization problems. *Mathematics of Computation*, 71(237):105–124, 2002.

[15] Y. Wu, T. Kanade, C. Li, and J. Cohn. Image registration using wavelet-based motion model. *Int. J. Computer Vision*, 38(2):129–152, 2000.

[16] J. Yuan, C. Schnörr, and E. Memin. Discrete orthogonal decomposition and variational fluid flow estimation. *Journ. of Math. Imaging & Vison*, 28:67–80, 2007.

[17] J. Yuan, C. Schnörr, and G. Steidl. Simultaneous higher-order optical flow estimation and decomposition. *SIAM Journal on Scientific Computing*, 29(6):2283–2304, 2007.

A. Gain from Smart Filter Banks

We consider filters h, g of length K and a current approximation a_j at a given scale $j \geq 0$ (i.e. with $2^j \times 2^j$ coefficients). Applying one step of the decomposition filter bank requires $8K2^{2(j-1)}$ multiplications, using the factorized form:

(i) for each of the 2^j rows:

- $2^{(j-1)}$ convolutions with $\bar{h}$;
- $2^{(j-1)}$ convolutions with $\bar{g}$;

which give $4K2^{2(j-1)}$ multiplications;

(ii) for each of the $2 \times 2^{(j-1)}$ columns:

- $2^{(j-1)}$ convolutions with $\bar{h}$;
- $2^{(j-1)}$ convolutions with $\bar{g}$;

giving again $4K2^{2(j-1)}$ multiplications;

So that the grand total is $8K2^{2(j-1)}$ multiplications. For an input image of $N = 2^F \times 2^F$ pixels, iteration of the filter bank to reach coarse scale C gives an upper bound of $\frac{8}{3}KN$ operations. Then, by dropping computation of details d_j^i:

(i) for each of the 2^j rows:

- $2^{(j-1)}$ convolutions with $\bar{h}$;

which give $2K2^{2(j-1)}$ multiplications;

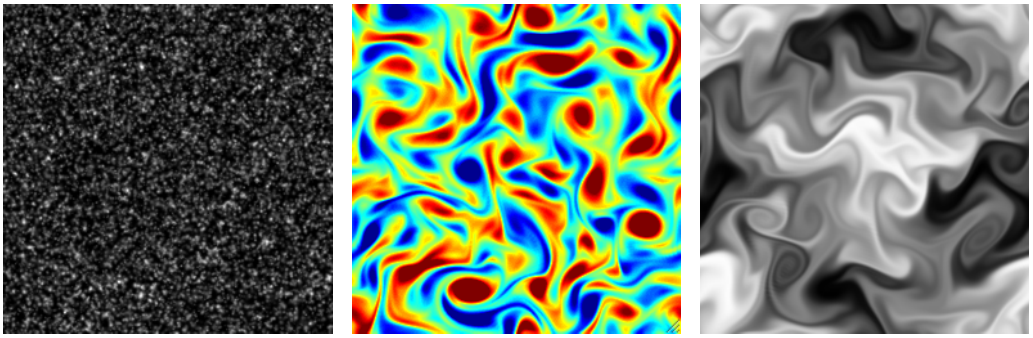
(ii) for each of the $2^{(j-1)}$ columns left:

- $2^{(j-1)}$ convolutions with $\bar{h}$;

which give $K2^{2(j-1)}$ multiplications;

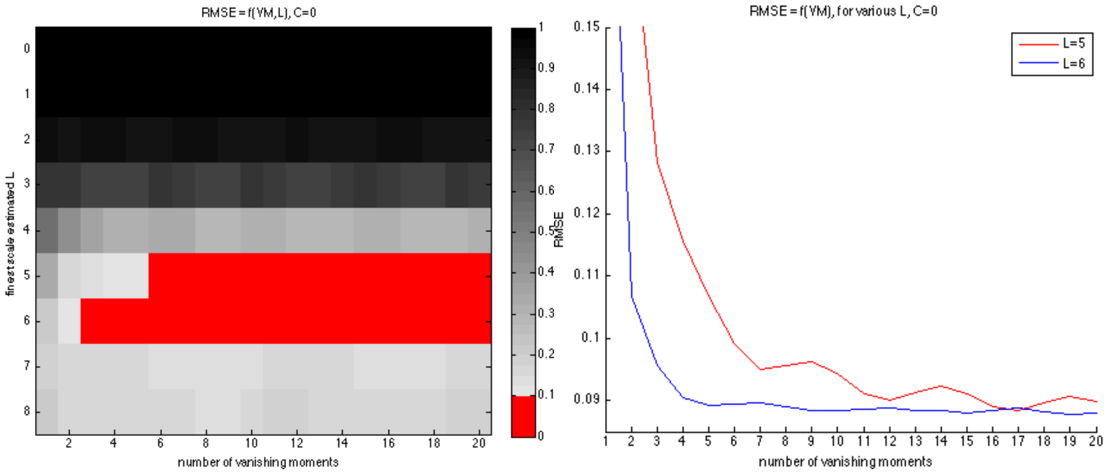
then the total number of required multiplications drops down to $3K2^{2(j-1)}$, hence saving 62.5% of multiplications at each step involving useless details. A step of the reconstruction filter bank to get a_j from a_{j-1} (and eventually $\{d_{j-1}^1, d_{j-1}^2, d_{j-1}^3\}$) requires the same amount of operations for both cases.

In practice, several scales – at least 2 – of null-coefficients are considered. These coefficients of the 2 finest scales represent 87.5% of the total, therefore the overall decrease in computation brought by the use of those detail-less filter banks is significant – Fig. 1.



(a) (b) (c)

Figure 3: The first frame of the synthetic *PIV image* sequence (a) with the *vorticity* of the underlying reference velocity field $\boldsymbol{v}_{\mathrm{ref}}$ (b). A sample frame from the synthetic *passive scalar advection-diffusion* sequence (c).



(a) (b)

Figure 4: RMSE between estimated $\hat{\boldsymbol{v}}$ and reference motion $\boldsymbol{v}_{\mathrm{ref}}$ as a function of the number of vanishing moments (VM) and the finest scale L, with $C = 0$ here. Region of interest, with RMSE < 0.1 pixel, is colored in *red*. It corresponds to $L = 5; 6$. Fig. (b) plots slices from Fig. (a) taken at $L = 5$ (*red*) and 6 (*blue*).

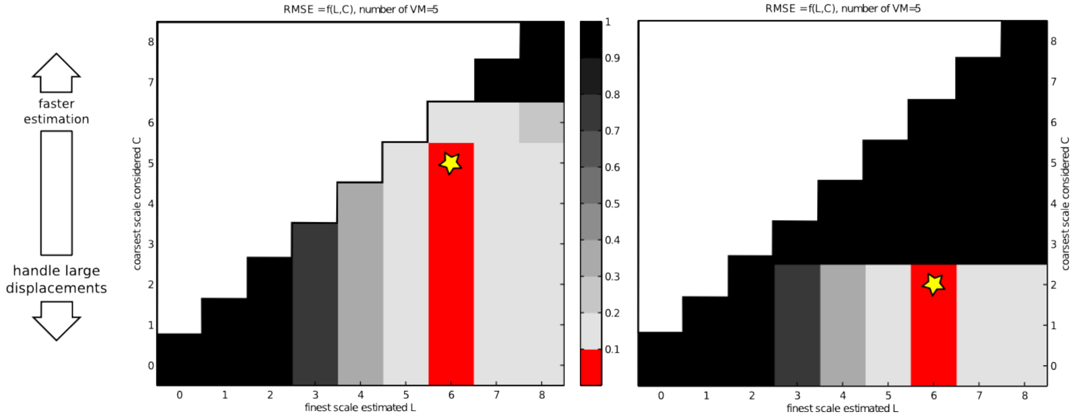
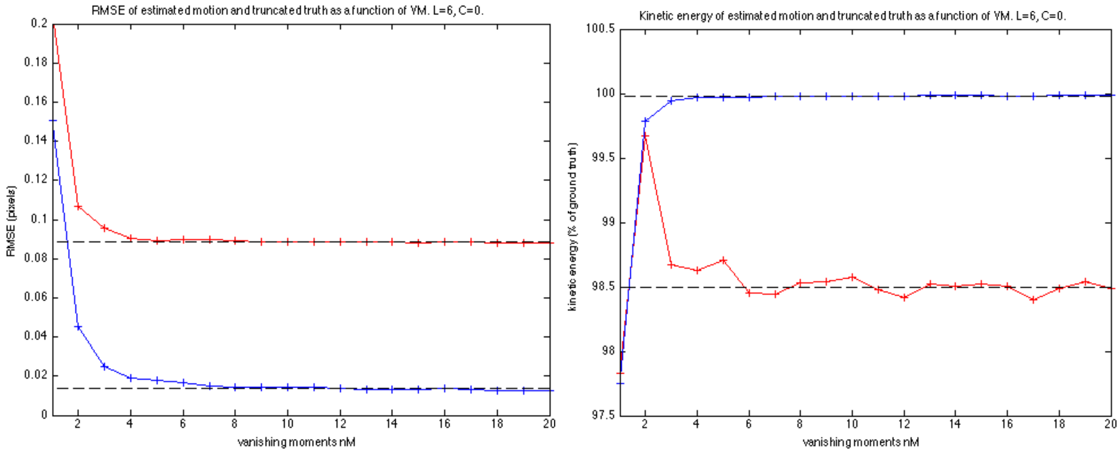
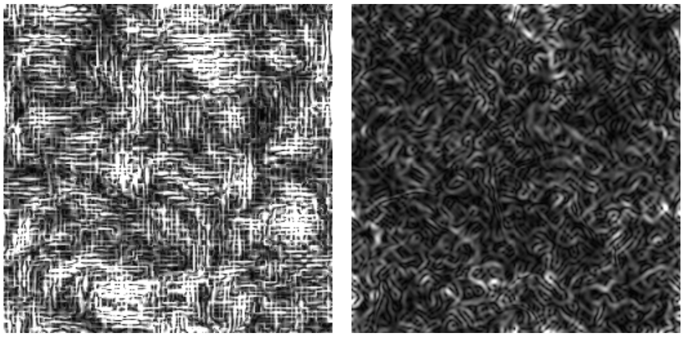


Figure 5: RMSE between estimated $\hat{\boldsymbol{v}}$ and reference motions $\boldsymbol{v}_{\mathrm{ref}}$ and $\tilde{\boldsymbol{v}}_{\mathrm{ref}}$ as a function of parameters L (finest motion coefficients scale 2^{-L}, *horizontal axis*) and C (coarsest motion coefficients scale 2^{-C}, *vertical axis*), with $L \geq C$. Low RMSE regions are colored in *red*. *Left* graph shows results obtained for $\boldsymbol{v}_{\mathrm{ref}}$ (*reference motion*), *right* graph presents results for $\tilde{\boldsymbol{v}}_{\mathrm{ref}}$ (*large magnitude*). Optimum couples of parameters, in terms of RMSE and computing time, are indicated for both cases by yellow stars.



(a) (b)

Figure 6: Experiments on the influence of the number of VM, for a given couple $(L, C) = (6, 0)$. Estimation results $\hat{\boldsymbol{v}}$ and "truncated" reference motion $\boldsymbol{v}_{\mathrm{ref}|6}$, both $\in V_6$, are compared to ground truth motion $\boldsymbol{v}_{\mathrm{ref}}$. Fig. sub6a compares RMSE of estimations (*red*) and truncated truth (*blue*). We observe a rapid convergence towards asymptots at 0.089 and 0.014 pixels for estimation and truncated truth, resp. Fig. (b) shows the kinetic energy of $\hat{\boldsymbol{v}}$ (*red*) and $\boldsymbol{v}_{\mathrm{ref}|6}$ (*blue*) as a percentage of $\boldsymbol{v}_{\mathrm{ref}}$ energy. We observe again a rapid convergence towards 98.5% and 99.98% for estimation and truncated truth, resp.



(a) (b)

Figure 7: Experiments on the influence of the number of VM. End-point-norm error maps obtained for estimates with $(L, C) = (6, 0)$, $n = 1$ (*left*) or $n = 20$ (*right*). Only small, tube-shaped structures remain un-estimated with $n = 20$.

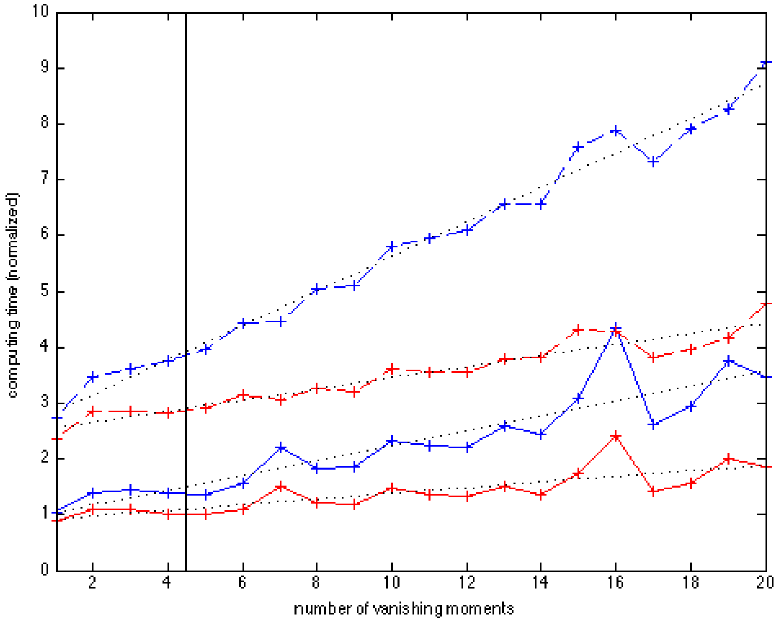
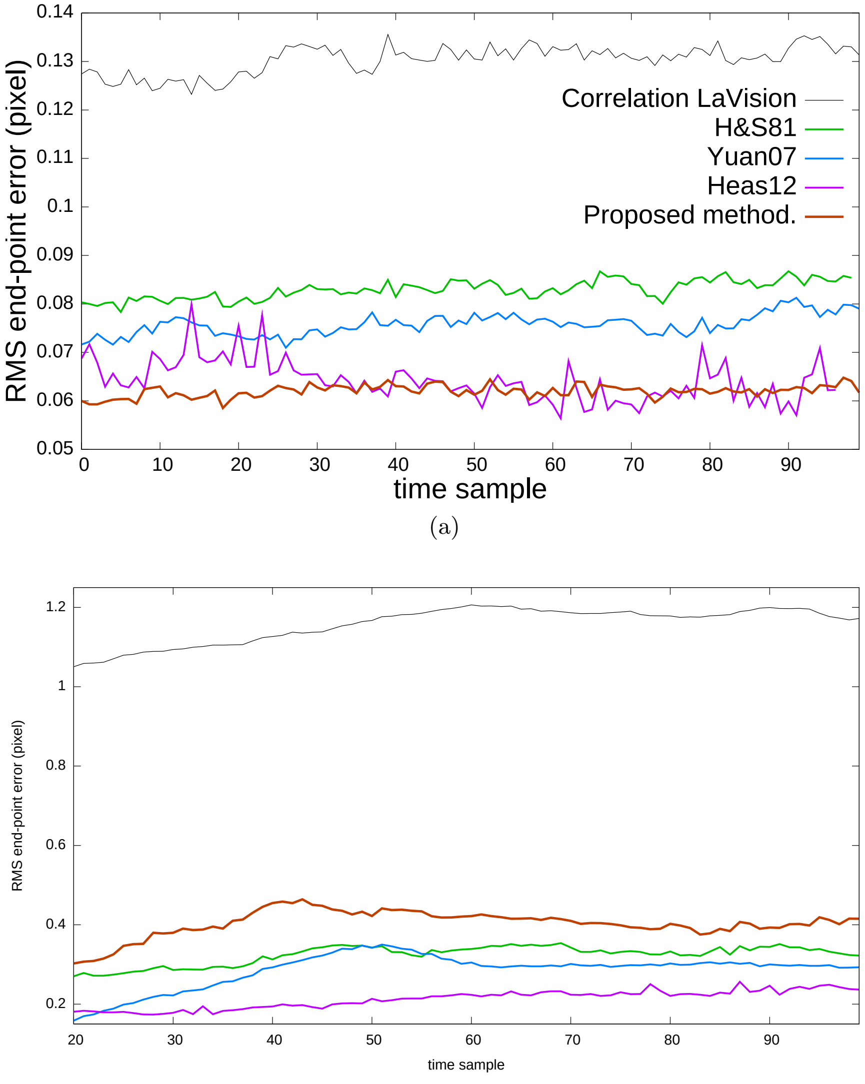


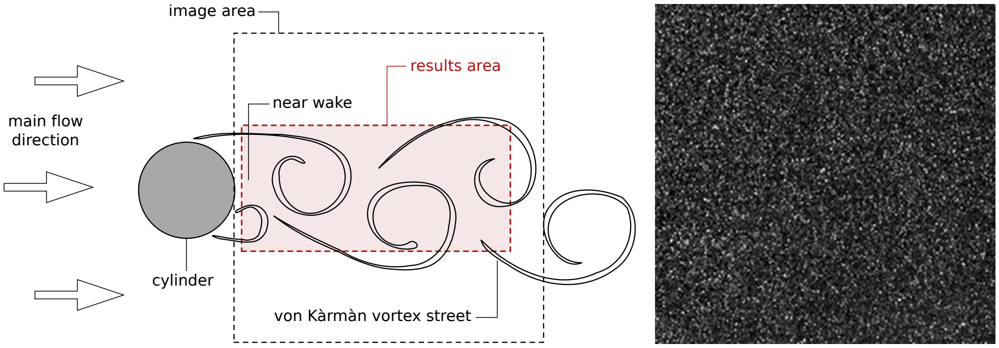
Figure 8: Experiments on the influence of the number of VM on the estimation computing time, using *usual filter banks* (*blue*) or *"smart" ones* (*red*), for $(L, C) = (6, 5)$ (*solid line*) and $(6, 0)$ (*dashed line*). Estimated motions are almost the same, in terms of RMSE, for any number of VM $n > 4$, any of the two couples (L, C) and of course any of the filter bank employed – the part of the graph located *right* of the vertical black line. Computation times are normalized w.r.t. optimum parameters, i.e. (L,C)=(6,5), $n = 5$ and smart filter bank. This reference time corresponds to Using usual filter banks, the computational burden rapidly increases with the number of VM, especially with low C (*dashed blue line*). This can be tempered by the use of smart filter banks (*dashed red*) and/or by choosing a higher value for C (*solid lines*).



(a)

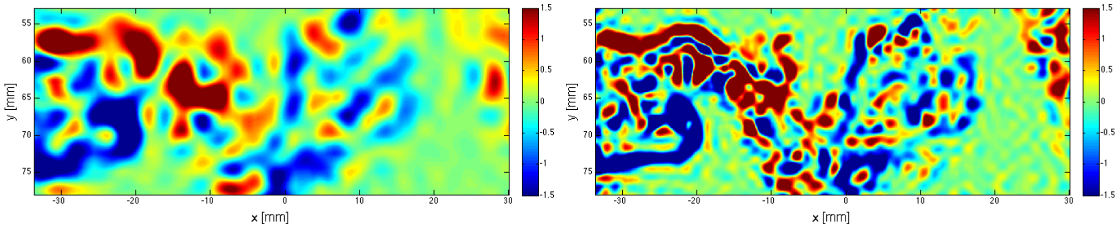
(b)

Figure 9: The two sequences introduced in Sec. 5.1 are processed with parameters $(L, C) = (6, 5)$ and $n = 20$ for particles, $(L, C) = (5, 5)$ and $n = 5$ for scalar images. Results (in *thick brown*) are compared, in terms of RMSE, to those of other estimators from state-of-the-art: correlation (*gray*), first order regularization [9] (*green*), div-curl regularization [16] (*blue*), multi-scale regularization [7] (*purple*). Fig. (a) shows *particles images* results, Fig. (b) concerns *passive scalar advection*.

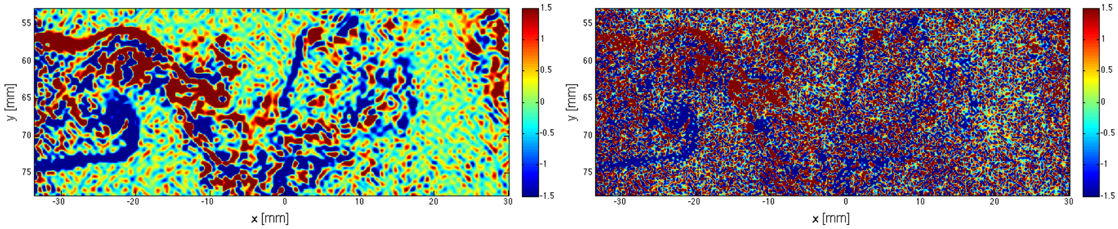


(a) (b)

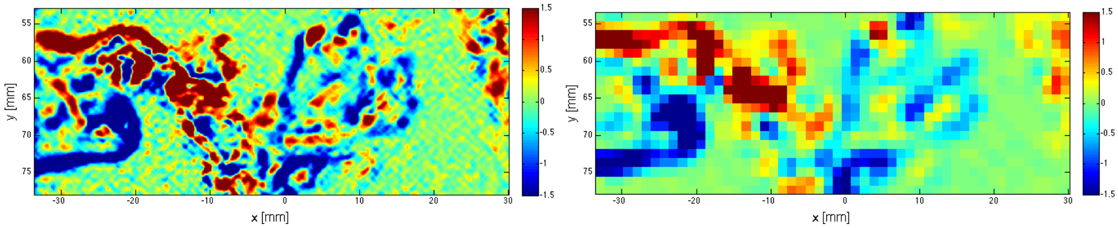
Figure 10: Configuration of the cylinder wake experiment (a); a 256 × 256 pixels subregion (1/16 of the total area) of a sample of PIV image (b).



(a) $L = 5$ (b) $L = 6$



(c) $L = 7$ (d) $L = 8$



(e) Order 2 smoothing (f) Correlations

Figure 11: Vorticity comparison. Estimates (a), (b), (c), (d) are obtained with proposed estimator, varying parameter L. Motion (e) is obtained with a more evolved estimator featuring explicit smoothing terms [10]. Last estimate (f) is a reference given by the correlation-based method.