A Multi-Time Step Method to Compute Optical Flow with Scientific Priors for Observations of a Fluidic System

Ranil Basnayake and Erik M Bollt

Abstract Optical flow is a classical problem in computer vision, but the concepts must be adapted for applications to other fields such as fluid mechanics and dynamical systems. Our approaches are based on an inverse problems formalism, considering imposed scientific priors in the form of a cost function that rewards an assumed infinitesimal generator commensurate with assumed physics of the observed density evolution. This leads to a practical and principled approach to analyze an observed dynamical system. Additionally we present here for the first time a new multi-frame version of the functional coupling of multiple images. Following the calculus of variations, this yields a coupled set of Euler-Lagrange PDEs which serve as an assimilation method that inputs video frames as driving terms. The solution of the PDE which follows is the vector field, as designed. Data from an oceanographic system will be highlighted. It is also shown here how these flow fields can be used to analyze mixing and mass transport in the fluid system being imaged.

1 Introduction

The goal of an optical flow method is to compute vector fields, considering the apparent motion between time-adjacent images of the same scene. Horn and Schunck in [11] introduced the original optical flow algorithm to detect the motion field of a moving object making two assumptions. The first assumption, a brightness constraint, states that the image brightness of a point on the brightness pattern is constant over time for small motions. This includes a notion of rigid body motion. Then an energy functional is obtained measuring the errors of the brightness constraint over the image domain so that the velocity components u and v along the x and y di-

Ranil Basnayake
Clarkson University, Box 5815, Potsdam, NY 13699,USA. e-mail: basnayrk@clarkson.edu

Erik M Bollt
Clarkson University, Box 5815, Potsdam, NY 13699,USA. e-mail: bolltem@clarkson.edu

rections respectively are obtained by minimizing the energy functional. However, in general, minimizing this energy functional is an ill-posed problem. Hence the above functional is regularized by making the second assumption that the expected flow is smooth, called the smoothness constraint. While the Horn and Schunck derivation was made in terms of local considerations only, the same PDE could be derived as a conservation law. The functional obtained from the brightness constraint is called the data term and the smoothness constraint is called the regularization term. The total energy functional includes a regularization term to the data term with weighting factor α, which is called the regularization parameter.

After regularizing the energy functional, the flow components u and v can be re-constructed by minimizing the derived energy functional. We achieve this by choosing a suitable regularization parameter α [10, 13, 14] to balance the desire to match the data fidelity but also to compromise with some form of regularity. We minimize the energy functional using a calculus of variations [7] approach. We apply Euler-Lagrange equations, first order necessary conditions to have an extremum and a functional as explained in Sect. 2.1, to the optical flow energy functional. For this problem, we have two coupled Euler-Lagrange equations to be solved for u and v. Fortunately, the resulting PDE system allows for known and relatively simpler numerical techniques such the Gauss-Seidel method, the gradient descent method, or the LU factorization with Gaussian elimination.

The above general optimization framework of optical flow computations allows re-construction of both flow components u and v for given image data. However, the only measured data is the image intensity. Since there is only one measurement, it is more stable to re-construct just one unknown function in such a way that the flow components u and v can be obtained from this computed function. One way we suggested in [15] to help with the stability of the optimization problem while at the same time representing the fluid nature of our problems of interests is to re-construct the stream/potential function ψ and then compute u and v by taking the Hamiltonian/usual gradient on the stream/potential function.

Our main goal here is to analyze unsteady fluid flow dynamics inferred using a sequence of image data of the system, taken by a movie camera or even from a satellite. Our previous methods required that there were only small changes of scene between each image, but otherwise the optimization approach just described yields a spatially regularized vector field in as far as the regularity term in the cost function is emphasized. However, if the scene in the movie data changes significantly between frames of the movie due to a relatively fast changing non-autonomous system, then there can be undesirable irregularity in the inferred vector field. This motivates us to develop a new approach to emphasize temporal regularity, a new approach of computing optical flow, using multiple images rather than just two. We will call this approach a *multi-time step optical* flow computation, which we introduce and develop here for the first time. For the computation of one stream function at a time, the multi-time step method and the stream function methods are the same. However,

when we compute more than one stream function for consecutive time points of the system simultaneously we wish to adjust the functional to emphasize that the stream functions of two consecutive time adjacent motions have similar behavior. This new assumption would incorporate an additional term in to the energy functional with a weighting factor β. We will demonstrate our method with a benchmark data set representing a gyre, and also the sea surface temperature (SST) data from the coast of Oregon, U.S.A as shown in Sect. 6.

Finally, we will demonstrate a use of the computed vector field to analyze mixing and mass transport in the fluid system being imaged. Several methods such as determining Lagrangian Coherent Structures (LCS) [3, 8, 18] and coherent pairs [6] are available to achieve this goal . In this endeavour we compute finite time Lyapunov exponents (FTLE) at each and every point in the system to determine LCSs. In the computation of FTLEs, we consider two nearby points at time t_0 and measure the separation of the trajectories over the time period $[t_0, T]$. If the separation is relatively high, the set of corresponding points are barriers for mixing and mass transport in the fluid system. These separation barriers on the FTLE fields are the LCSs. The LCS for the SST data set are computed using the vector fields obtained from the multi-time step method and shown in Fig. 9.

2 Classical optical flow method

According to the original Horn and Schunck formulation of optical flow [11], the image brightness $I(x,y,t)$ at a point (x,y) is assumed to be locally conserved over the time if the motion is small. This implies that

$$I(x,y,t) = I(x+u, y+v, t+1), \tag{1}$$

where u and v are the velocity components along the x and y directions respectively and here we consider a unit time interval. Note that $I(x,y,t)$ is the assumed data function. $I : \Omega \times \mathbb{R}^+ \to \mathbb{R}$ represents the gray scale color intensity of an image at time t, where (x,y) is in the scene domain Ω. Generally, actual data from a digital movie camera is pixelated at discrete spatial locations $\{x_i, y_j\}_{i,j=1}^{p,q}$ at times t_k, $\{I_{i,j,k}\}_{i,j,k=1}^{p,q,T}$ at times t_k as a complete data set, where $t_1 = 0$ and t_k is the time after $k-1$ units. In other words, the image one was taken at $t = 0$ and the image k was taken after $k-1$ time units from image 1 has been taken. Here p and q are the number of rows and the number of columns of the input images. The Taylor series expansion on Eq. (1) is

$$I_t + I_x u + I_y v = \delta, \tag{2}$$

where δ is the errors accumulated from the higher order terms and I_t, I_x and I_y are partial derivatives of I with respect to t, x and y respectively. The data term can be obtained by integrating the errors of the brightness constraint over the image domain as given by

$$E_b(u,v) = \int_\Omega (I_t + I_x u + I_y v)^2 \, d\Omega. \tag{3}$$

To develop u and v, as suggested by the brightness constraint objective in Eq. (2) we state the functional in Eq. (3). However minimizing $E_b(u,v)$ in Eq. (3) is an ill-posed problem. To avoid ill-posedness of this problem, the data term must be regularized. We approach the regularization by assuming the expected flow is smooth. This implies that the partial derivatives of the velocity components u and v exist and hence suggests the regularization term becomes

$$R(u,v) = \int_\Omega \left(u_x^2 + u_y^2 + v_x^2 + v_y^2\right) \, d\Omega. \tag{4}$$

Now the problem can be reformulated in terms of an energy functional obtained by combining the data term and the regularization term with weighting the second term by non-negative regularization parameter α. Then the total energy functional to be minimized is given by,

$$E(u,v) = \int_\Omega (I_t + I_x u + I_y v)^2 \, d\Omega + \alpha \int_\Omega \left(u_x^2 + u_y^2 + v_x^2 + v_y^2\right) \, d\Omega. \tag{5}$$

2.1 Euler-Lagrange Equations

To minimize the functional in Eq. (5), we use a Calculus of variations approach and in this subsection we will demonstrate methods to minimize our functionals as in [7]. First consider minimization of a simple functional and necessary conditions to have a minimum to be solved for u and v.

Suppose we are given a functional $J(u)$ and to be determined is the optimizer $u(x) = \hat{u}(x)$ over the interval $a \leq x \leq b$

$$J(u) = \int_a^b F(x,u,u_x)\, dx. \tag{6}$$

Here $F(x,u,u_x)$ is a function with continuous first and second partial derivatives with respect to all of its arguments. Also, let $u(x)$ be a continuously differentiable function on $[a,b]$ which satisfies the boundary conditions

$$u(a) = A \qquad \text{and} \qquad u(b) = B. \tag{7}$$

Optimization necessitates that the first variations are stationary. Analogous to the first derivative of a function, here we obtain the first variation of the given functional. First we give an increment $h(x)$ to the function $u(x)$ with the boundary conditions

$$h(a) = h(b) = 0. \tag{8}$$

The corresponding increment ΔJ in Eq. (6) with respect to the increment of $h(x)$ is

$$\Delta J = \int_a^b [F(x,u+h,u_x+h_x) - F(x,u,u_x)]\,dx. \tag{9}$$

The first variation of the functional Eq. (9) with respect to the argument u is obtained by taking the first order terms of the Taylor series expansion on the integrand of Eq. (9) as

$$\delta J = \int_a^b [F_u(x,u,u_x)u + F_{u_x}(x,u,u_x)u_x]\,dx. \tag{10}$$

Theorem 1. [7] *A necessary condition for the differentiable functional $J(u)$ to have an extremum for $u = \hat{u}$ is that its variation vanish for $u = \hat{u}$, i.e., that*

$$\delta J(h) = 0 \tag{11}$$

for $u = \hat{u}$ and all admissible h.

From the above theorem, the necessary condition, called the Euler-Lagrange equation, for the functional in Eq. (6) to have an extremum can be obtained as

$$\frac{\partial F}{\partial u} - \frac{d}{dx}\left(\frac{\partial F}{\partial u_x}\right) = 0. \tag{12}$$

This statement can be expanded to allow for many variables and we are interested to expand the results to a functional of the form

$$J(u,v) = \int_\Omega F(x,y,u,v,u_x,u_y,v_x,v_y)\,d\Omega, \tag{13}$$

allowing for vector fields $\langle u(x,y), v(x,y)\rangle$ in the plane $(x,y) \in \mathbb{R}^2$. Since there are two argument functions u and v, we may have two coupled Euler-Lagrange equations. The Euler-Lagrange equations for the functional in Eq. (13) follow as

$$\begin{aligned} \frac{\partial F}{\partial u} - \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial u_x}\right) - \frac{\partial}{\partial y}\left(\frac{\partial F}{\partial u_y}\right) &= 0 \\ \frac{\partial F}{\partial v} - \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial v_x}\right) - \frac{\partial}{\partial y}\left(\frac{\partial F}{\partial v_y}\right) &= 0. \end{aligned} \tag{14}$$

The above Euler-Lagrange equations are specialized below for functionals such as of the form in Eq. (5) and hence we can determine the velocity components by solving the resulting Euler-Lagrange PDE equations for u and v.

Another way to compute the Euler-Lagrange equations for a functional $J(u)$ in Eq. (6) is to compute the Gateaux derivative $J(u)$ with respect to u and set it to zero. The Gateaux derivative of the functional $J(u)$ is obtained as

$$DJ(u) = \frac{d}{d\tau} J(u+\tau h)|_{\tau=0} \tag{15}$$

for all admissible h.

2.2 Solution to the optical flow problem

In general, when we minimize the optical flow problem, assuming existence and uniqueness of the solution, we determine the minima by solving the Euler-Lagrange equations for each argument variable. The issues of existence and uniqueness of solutions of the PDEs in Eq. (14) follow the theory of convex optimization including discussion of convexity, coercivity, and lower semi-continuity of the specific functional $J(u,v)$. This goes beyond the needs of the discussion here, but we refer to the excellent text book [19]. The following theorem from [19] explains sufficient conditions to have a unique minimum for the functional $J(u,v)$.

Theorem 2. [19] *Assume that $J : H \to \mathbb{R}$ is weakly lower semi-continuous and coercive and that C is a closed, convex subset of H. Then J has a minimizer over C. If furthermore, J is also strictly convex, then the minimizer is unique.*

In the minimization process, first we apply the equations in Eq. (14) to the functional in Eq. (5) to obtain the Euler-Lagrange equations, the gradients of the energy functional with respect to u and v, as

$$u_t = I_x(I_t + I_x u + I_y v) + \alpha\Delta^2 u, \qquad \text{and} \qquad v_t = I_y(I_t + I_x u + I_y v) + \alpha\Delta^2 v. \quad (16)$$

When we numerically solve the above system for u and v, one way of reaching the solution is to use numerical iterative methods such as the gradient descent method or the Gauss-Seidel method. For instance, the gradient decent algorithm is an iterative method which updates u and v for given initial conditions u_0 and v_0 as

$$\begin{aligned} u^{(k+1)} &= u^{(k)} + \delta t[I_x(I_t + I_x u^{(k)} + I_y v^{(k)}) + \alpha\Delta^2 u^{(k)}] \\ v^{(k+1)} &= v^{(k)} + \delta t[I_y(I_t + I_x u^{(k)} + I_y v^{(k)}) + \alpha\Delta^2 v^{(k)}] \end{aligned} \quad (17)$$

Here k represents the iteration number and δt is the time step. Recall $u^{(k)}$ and $v^{(k)}$ must be discretely represented on the grid $\{x_i, y_j\}_{i,j=1}^{p,q}$ and derivatives must be numerically approximated by finite differences.

On the other hand, we can linearize the Laplacian terms in the Euler-Lagrange equations by introducing operators to compute second partial derivatives of u and v and then use direct methods to solve the resulting linear system. We will explain how to use the operator matrices to simplify a system in Sect. 3.

3 Stream function method

The Horn and Schunck optical flow method was developed to capture rigid body motion. Later research [15] applies this method to capture vector fields in fluid systems but changes must better respect expected fluid behavior. When we deal with a fluid system, dealing with the stream function of the motion is an accurate and precise way to develop velocity components. Note in order to use the stream function, we assume the fluid is incompressible. Therefore we convert the optical flow method into a stream function formulation. If the stream function is $\psi(x,y)$, then the velocity components are obtained as

$$\langle u, v\rangle = \langle -\psi_y, \psi_x\rangle. \tag{18}$$

Substituting $-\psi_y$ and ψ_x in to Eq. (5), the energy functional in terms of the stream function ψ becomes

$$E(\psi) = \int_\Omega (I_t - I_x\psi_y + I_y\psi_x)^2 \, d\Omega + \alpha \int_\Omega (\psi_{xx}^2 + \psi_{yy}^2 + \psi_{xy}^2 + \psi_{yx}^2) d\Omega \tag{19}$$

The one of the other advantage of the stream function method is we can impose the regularity directly to the stream function rather than flow components. For instance, if the resulting flow is known to be sparse, then the appropriate regularization term [17] in the usual optical flow computation "uv- method" is,

$$R(u,v) = \int_\Omega (\mid u \mid + \mid v \mid) \, d\Omega. \tag{20}$$

Minimization of this regularization term emphasizes u to be zero in some places or v to be zero in some places which does not imply the flow to be zero. However, in the stream/potential function method, we can apply the scientific prior directly to the flow by regularizing the energy functional by the choice

$$R(\psi) = \int_\Omega \mid \bigtriangledown \psi \mid d\Omega. \tag{21}$$

This would measure the sparsity of the flow rather than components of the flow.

Next we review the development of Euler-Lagrange equations for the stream function method. Taking the Gateaux derivative as in Eq. (15), the gradients of the data fidelity

$$\int_\Omega (I_t - I_x\psi_y + I_y\psi_x)^2 \, d\Omega, \quad \text{and the regularization term}$$

$$\int_\Omega \left(\psi_{xx}^2 + \psi_{yy}^2 + \psi_{xy}^2 + \psi_{yx}^2\right) d\Omega, \quad \text{are obtained as}$$

$$A^*(I_t + A)\psi \quad \text{and} \quad (B + B^*)\psi,$$

respectively. Here A^* is the adjoint of operator A, and A and B are obtained as

$$A = -I_x D_y + I_y D_x \quad \text{and}$$

$$B = D_{xx} D_{xx}^* + D_{yy} D_{yy}^* + D_{xy} D_{xy}^* + D_{yx} D_{yx}^*,$$

following standard considerations of calculus variations [19]. In the operators A and B, the operator $D_{\bullet\bullet}$ is a matrix operator of size $m \times m$ to compute the partial derivatives of a given vector of size $m \times 1$ with respect to indices $\bullet\bullet$. Here we stack the given matrix of size $p \times q$ into $m \times 1$ vector and $m = pq$. To develop the operator matrices $D_{\bullet\bullet}$, we use finite difference approximations with suitable boundary conditions. Finally, choosing a suitable regularization parameter α, the stream function ψ can be determined by solving the following Euler-Lagrange equation

$$[A^* A + \alpha(B + B^*)]\psi = -A^* I_t. \tag{22}$$

Now the system can be solved for ψ by taking the LU decomposition on $A^*A + \alpha(B + B^*)$ and then applying Gaussian elimination. The velocity components u and v can be derived from Eq. (18).

4 Multi-time step method

Now we introduce the new multi-time step method of computing optical flow for a sequence of several time dependent images to compute n time dependent vector fields simultaneously. Furthermore, we can use higher order finite difference approximations to compute I_t rather than the forward difference approximation necessary when just two images are available. In this endeavor our multi-time step optical flow algorithm is obtained based on the stream/potential optical flow algorithm and the energy functional for $n = 1$ is given in Eq. (19); the solution can be obtained by solving the system Eq. (22). When we consider $n = 2$ or more, we include the additional term in the data fidelity by introducing regularity in the time direction by assuming two consecutive stream functions have similar behavior. Suppose we are given T time adjacent images as a movie of a dynamical system. Then evolving the window is slow enough that considerations of continuously evolving frame views allows inference of the underlying dynamical systems though the flow is unsteady. In other words assume $I(x,y,t)$ is continuous with respect to t throughout the scene. In fact we cope with this assumption by including a new term with a weighting factor. For instance, if there are only two stream functions ψ_1 and ψ_2, the additional minimizing integral would be

$$\int_\Omega (\psi_1 - \psi_2)^2 \, d\Omega \tag{23}$$

added to the chosen data fidelity and regularization terms already designed for assumed scientific priors. In this case we use three images at a time to compute the

flow in two different time instances unless we use higher order finite difference approximations.

We include into the new functional a parameter $\beta > 0$ in the previous energy functional and the resulting energy functional is given by,

$$\begin{aligned} E(\psi_1,\psi_2) = \int_\Omega \left(I_{1t} - I_{1x}\psi_{1y} + I_{1y}\psi_{1x}\right)^2 + \left(I_{2t} - I_{2x}\psi_{2y} + I_{2y}\psi_{2x}\right)^2 d\Omega \\ +\beta \int_\Omega (\psi_1 - \psi_2)^2 \, d\Omega \\ +\alpha \int_\Omega (\psi_{1xx}^2 + \psi_{1yy}^2 + \psi_{1xy}^2 + \psi_{1yx}^2) d\Omega \\ +\alpha \int_\Omega (\psi_{2xx}^2 + \psi_{2yy}^2 + \psi_{2xy}^2 + \psi_{2yx}^2) d\Omega. \end{aligned} \tag{24}$$

Thus we have data fidelity in two time instances at once, the term to emphasize time regularity and the spatial regularity.

Now taking the Gateaux derivative as in Eq. (15) of the functional in Eq. (24), the Euler-Lagrange equations corresponding to ψ_1 and ψ_2 are the system of PDEs as,

$$\begin{aligned} [A_1^* A_1 + \beta(\psi_1 - \psi_2) + \alpha(B + B^*)]\psi_1 &= -A_1^* I_{1t} \\ [A_2^* A_2 - \beta(\psi_1 - \psi_2) + \alpha(B + B^*)]\psi_2 &= -A_2^* I_{2t}, \end{aligned} \tag{25}$$

where,

$$\begin{aligned} A_1 &= (-I_{1x}D_y + I_{1y}D_x), \\ A_2 &= (-I_{2x}D_y + I_{2y}D_x) \quad \text{and} \\ B &= D_{xx}D_{xx}^* + D_{yy}D_{yy}^* + D_{xy}D_{xy}^* + D_{yx}D_{yx}^*. \end{aligned}$$

We can generalize the energy functional for n stream functions at n successive time instances as shown,

$$\begin{aligned} E(\psi_1,\psi_2,...,\psi_n) = \sum_{k=1}^{n} \int_\Omega \left(I_{kt} - I_{kx}\psi_{ky} + I_{ky}\psi_{kx}\right)^2 d\Omega \\ +\beta \sum_{k=1}^{n-1} \int_\Omega (\psi_k - \psi_{k-1})^2 \, d\Omega \\ +\alpha \sum_{k=1}^{n} \int_\Omega (\psi_{kxx}^2 + \psi_{kyy}^2 + \psi_{kxy}^2 + \psi_{kyx}^2) d\Omega. \end{aligned} \tag{26}$$

Similarly to $n = 2$, the Euler-Lagrange equations can be obtained for any number n, using the Gateaux derivative as in Eq. (15) of the functional in Eq. (26) with respect to $\psi_1, \psi_2, ...,$ and ψ_n. The system of Euler-Lagrange equations for any integer n is a system of n partial differential equations and is obtained as

$$A_1^*(I_{1t} + A_1\psi_1) + \beta(\psi_1 - \psi_2) + \alpha(B + B^*)\psi_1 = 0$$

$$A_k^*(I_{kt}+A_k\psi_k)+\beta(-\psi_{k-1}+\psi_k-\psi_{k+1})+\alpha(B+B^*)\psi_k=0, \quad \text{for } k=2,3,...n-1$$
$$A_n^*(I_{nt}+A_n\psi_n)+\beta(\psi_{n-1}-\psi_n)+\alpha(B+B^*)\psi_n=0,$$

where,

$$A_k=\left(-I_{kx}D_y+I_{ky}D_x\right) \ \ \forall k \ \text{ and}$$
$$B=D_{xx}D_{xx}^*+D_{yy}D_{yy}^*+D_{xy}D_{xy}^*+D_{yx}D_{yx}^*.$$

The above set of Euler-Lagrange equations can be reformulated as a linear system as

$$[K+\alpha L]\,z=b \tag{27}$$

where

$$K=\begin{bmatrix} A_1^*A_1+\beta & -\beta & & & & \\ -\beta & A_2^*A_2+\beta & -\beta & & & \\ \ddots & \ddots & \ddots & & & \\ & & -\beta & A_{n-1}^*A_{n-1}+\beta & -\beta & \\ & & & \beta & A_n^*A_n-\beta \end{bmatrix},$$

$$L=\begin{bmatrix} B+B^* & & & & \\ & B+B^* & & & \\ & \ddots & & & \\ & & B+B^* & & \\ & & & B+B^* \end{bmatrix}, \quad z=\begin{bmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_{n-1} \\ \psi_n \end{bmatrix} \ \text{ and } \ b=\begin{bmatrix} A_1^*I_{1t} \\ A_2^*I_{2t} \\ \vdots \\ A_{n-1}^*I_{(n-1)t} \\ A_n^*I_{nt} \end{bmatrix}.$$

All the entries of the matrices K and L are block matrices of the size $m\times m$, where $m=pq$ and $p\times q$ is the dimensions of the image domain. The entries of the vectors z and b are vectors of the size $m\times 1$.

To solve the above system, first we take the LU decomposition of the matrix $K+\alpha L$ and then use Gaussian elimination. Solution of the above system yields $\psi_1(x,y)\equiv\psi_1(x,y,t_1)$, $\psi_2(x,y)\equiv\psi_2(x,y,t_2)$, ..., $\psi_n(x,y)\equiv\psi_n(x,y,t_n)$ and taking the Hamiltonian gradient in Eq. (18) on each separately, the vector fields for each time instance, $t_1,t_2,...,t_n$ can be obtained. That is, the velocity components $u_1,u_2,...,u_n$ and $v_1,v_2,...,v_n$ are computed from $\psi_1,\psi_2,...,\psi_n$ using $\langle u_k,v_k\rangle=\langle-\psi_{ky},\psi_{kx}\rangle$ for $k=1,2,...n$. To improve the accuracy of the results, instead of first order finite differences for the time derivative of I, higher order finite difference approximations can be used when more than two images are available.

5 Scientific priors

Toward our discussion of designing scientific prior information into our functional, we first review some well known inverse problems theory. Recall that in linear algebra, solving a system $Au=z$ when A and u are given, is called a forward

problem. When A and z are available and the system is solved for u, it is called an inverse problem. For an inverse problem, a unique solution can be obtained by

$$u = A^{-1}z, \text{ if } A^{-1} \text{ exists.}$$

If A is not invertible, then the system may have infinitely many solutions or no solutions. In this case, it is standard to re-formulate the problem as a minimization problem in such a way that from infinitely many solutions we can select or emphasize a desired solution. Then the new problem can be written as

$$\arg\min_u \|Au - z\|^2. \tag{28}$$

In the modified problem, we find u in such a way that to minimize the distance between Au and z according to the associated norm. For instance, if the norm is taken to be Euclidean $\|\cdot\|_2$, then such a minimal solution emphasizes the Euclidean perspective that the good solution should be on a radial closest the origin. We call this notion the scientific prior of this simple linear problem. The procedure called imposing a scientific prior on the data is done by adding a new term, a regularization term $R(\psi)$, to the modified problem with a non-negative weighting parameter α. On the other hand, the scientific prior can be applied to the problem at the beginning of the construction of the model, by choosing different operators for A according to the prior knowledge of the data, and expected solutions.

Analogously, building scientific prior information into functionals allows our inverse problem solutions for vector fields to emphasize expected physics. Now we will discuss various operators and regularizations to emphasize expected scientific prior information.

In the nominal optical flow algorithm, Horn and Schunck assumed conservation of image brightness intensity locally for rigid body motion. For the entire domain Ω with time step t, the conservation of image brightness I is emphasized by

$$E(\psi) = \int_\Omega (I_t - I_x\psi_y + I_y\psi_x)^2 \, d\Omega, \tag{29}$$

where ψ is the corresponding stream function for the motion. However, later researchers were interested in fluid motion and assumed that the image brightness I behaves according to the continuity equation over time in order to allow for divergent flow fields. Therefore Corpetti in [5] proposed the data fidelity term,

$$E(\psi) = \int_\Omega (I_t - I_x\psi_y + I_y\psi_x - I\psi_{yx} + I\psi_{xy})^2 \, d\Omega. \tag{30}$$

Also J. Weickert in [20] improved the data fidelity by imposing the constancy of spatial brightness gradient, instead of brightness constancy, with the following,

$$E(\psi) = \int_{\Omega} (I_{xt} - I_{xx}\psi_y + I_{xy}\psi_x)^2 + (I_{yt} - I_{yx}\psi_y + I_{yy}\psi_x)^2 \, d\Omega. \tag{31}$$

Further, researchers in [4] have combined the data fidelities Eq. (29) and Eq. (31) together with a non-negative parameter β. The resulting data fidelity is obtained as

$$E(\psi) = \int_{\Omega} (I_t - I_x\psi_y + I_y\psi_x)^2 \, d\Omega + \beta \int_{\Omega} (I_{xt} - I_{xx}\psi_y + I_{xy}\psi_x)^2 \, d\Omega + \beta \int_{\Omega} (I_{yt} - I_{yx}\psi_y + I_{yy}\psi_x)^2 \, d\Omega. \tag{32}$$

In this data fidelity, either image brightness is constant or the gradient of the image brightness is constant, by adjusting β to emphasize either expected outcome or underlying physics. Through this kind of approach, the notion of modeling, the physics behind creating the scenes evolution in the movie-data is designed in to the functional.

In addition to the data fidelities, it is important to regularize the energy functional, as understood by Andrey Tikhonov as shown in [12] to extract important information from ill-posed problem, both for functional analysis and optimization theoretic reasons of well-posedness, and to further emphasize expected physics. For instance, if the resulting flow is expected to be sparse, then the regularization term that emphasizes sparsity is the total variation of the stream function. The regularization term can be written as,

$$R_1(\psi) = \int_{\Omega} \mid \bigtriangledown \psi \mid d\Omega. \tag{33}$$

If, however the flow field is expected to be smooth, then the appropriate regularization is the Horn and Schunck type regularization,

$$R_2(\psi) = \int_{\Omega} (\psi_{xx}^2 + \psi_{yy}^2 + \psi_{xy}^2 + \psi_{yx}^2) d\Omega. \tag{34}$$

On the other hand, if we impose the regularity on flow components, we would use,

$$R_3(\psi) = \int_{\Omega} (\psi_x^2 + \psi_y^2) d\Omega. \tag{35}$$

According to the above explanation, we can develop different data terms and regularization terms using the known physics of the data and the expected flow. We can construct various multi-time step algorithms according to the prior knowledge of systems being imaged. Hence applying a suitable algorithm, the motion field of the observed system can be determined and then used to analyze the dynamics of the system.

6 Results from multi-time step method

In this section, we will demonstrate the importance of using the data term in Eq. (29) and the regularization term in Eq. (34). Also we present the improvement of the accuracy of our algorithm with larger n. For the above purpose, we introduce a simple benchmark data set and an oceanic data set. Then we compare and benchmark some data terms and regularization terms introduced in Sect. 5.

6.1 Synthetic data

Since we are interested in applying our algorithm to capture fluid motions, we constructed a benchmark data set, called the gyre which is stereotypical of fluid motion. The stream function

$$\psi(x,y) = \sin(\pi x)\sin(\pi y) \tag{36}$$

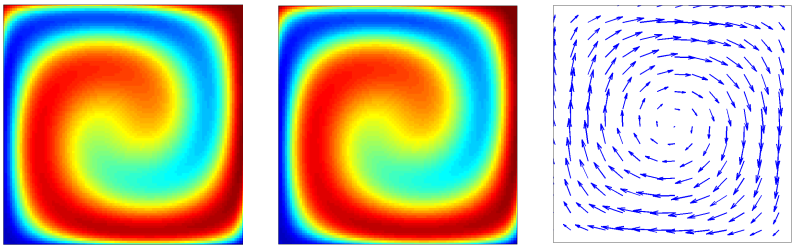
on the domain $[0,1]\times[0,1]$ was considered and the vector field governed by Eq. (36) is

$$\langle u,v\rangle = \langle -\pi\sin(\pi x)\cos(\pi y), \pi\cos(\pi x)\sin(\pi y)\rangle. \tag{37}$$

Evolving an initial condition over the domain $[0,1]\times[0,1]$ using the autonomous vector field in Eq. (37) according to the continuity equation

$$\frac{dI}{dt} = -(I_x u + I_y v + I u_x + I v_y), \tag{38}$$

a sequence of images is obtained. Two images are selected after transients and are shown in Fig. 1, images (a) and (b), and the vector field in Eq. (37) is shown in image (c).

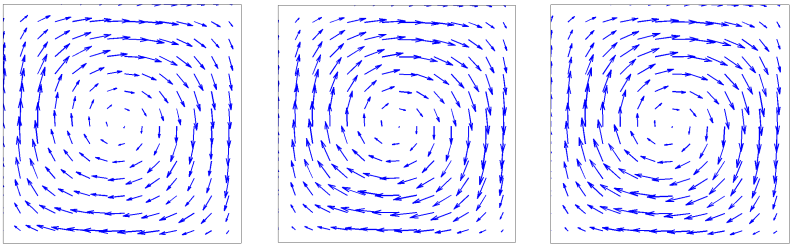


(a) (b) (c)

Fig. 1 Gyre data and true flow – Images (a) and (b) show two later time instances of an initial density that has been evolved according to (38) with velocity components given by (37). The true flow field is shown in (c).

When we compare the reconstructed vector fields with the true vector field, we need a figure of merit for comparison. Therefore, we computed angular error between the computed flow and the true flow and then the mean over the domain to obtain mean angular error (MAE) as explained in [16]. In this computation we include a third unit component to both computed and true velocity components to avoid the incorrect measurements near the points where flow magnitude is close to zero and sufficiently large.

As explained in Sect. 5, there are various ways to develop an energy functional to reconstruct vector fields. However, we choose the data term in Eq. (29) and regularization term in Eq. (34), introduced in the original Horn and Schunck method, to develop our multi-time step method. We introduce two other possible regularization terms in Eq. (33) and Eq. (35) to cope with optical flow functionals. We combine three regularization terms in Eq. (33), Eq. (34) and Eq. (35) with the data term in Eq. (29) and the flow reconstructions are shown in Fig. 2. Note that in this presentation, we use the Lagged Diffusivity fixed point iteration method as in [2] to reconstruct the vector fields, when the regularization term in Eq. (33) is combined with any data term. Furthermore, we use multi-time step method with $n = 1$ to compare the data terms and the regularization terms.

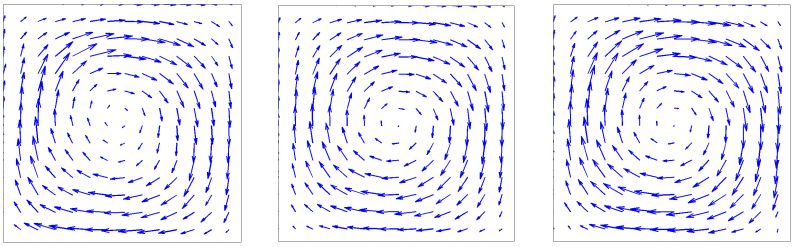


(a) (b) (c)

Fig. 2 Gyre flow with data term in Eq. (29) – Images (a), (b) and (c) show the computed flow fields for the images (a) and (b) shown in Fig. 1 from the data term in Eq. (29) with regularization terms in Eq. (33), Eq. (34) and Eq. (35) respectively.

The computed MAE for the Energy functionals constructed from the data term in Eq. (29) with the regularization terms in Eq. (33), Eq. (34) and Eq. (35) are 2.4247°,0.9837°, and 2.2790° respectively. Among the three regularization terms, the regularization term in Eq. (34) gives least MAE. Further, we can use the data term in Eq. (30) instead of Eq. (29) with the regularization terms in Eq. (33), Eq. (34) and Eq. (35) to develop another three different energy functionals and three different reconstructions are shown in Fig. 3.

The mean angular errors obtained from minimizing enery functionals of the data term in Eq. (30) with the regularization terms in Eq. (33), Eq. (34) and Eq. (35) are 3.3909°,0.9895° and 2.4787° respectively. In this case also, the energy functional



(a) (b) (c)

Fig. 3 Gyre flow with data term in Eq. (30) – Images (a), (b) and (c) show the computed flow fields for the images (a) and (b) shown in Fig. 1 from the data term in Eq. (30) with regularization terms in Eq. (33), Eq. (34) and Eq. (35) respectively.

obtained from data term in Eq. (30) with the regularization terms in Eq. (34) gives the least mean angular error even though it is slightly bigger than the combination of the data term in Eq. (29) with regularization terms in Eq. (34). This verifies our selection for $n = 1$ and hence for larger n also we use the data term in Eq. (29) with regularization terms in Eq. (34).

The next step is to compare the results by varying the step size n, the number of stream functions ψ computed at a time using multiple images. We applied our multi-time step method on a image sequence of gyre data set of which (a) and (b) of Fig.1 show the fourth and fifth images of that. For instance, if we apply a first order finite difference approximation to estimate I_t with the step size $n = 1$, then we compute one stream function ψ_1 using two images, image 1 and image 2. The velocity components $(u_1, v_1) = (-\psi_{1y}, \psi_{1x})$ represent the motion field between image 1 and image 2. If however we choose the step size $n = 2$, then we compute two stream functions ψ_1 and ψ_2 using three images. Then $(u_1, v_1) = (-\psi_{1y}, \psi_{1x})$ is the motion field between image 1 and image 2 and $(u_2, v_2) = (-\psi_{2y}, \psi_{2x})$ is the motion field between image 2 and image 3 respectively. Continuing in this manner, we can increase the step size n. Note that, if there is a sequence of nine images, we can compute eight vector fields for each consecutive image pair. When the step size is $n = 1$, eight separate computations are necessary but when $n = 2$, only four computations are necessary and so on. Again we emphasize that the advantage of choosing larger n is that the time regularity is emphasized as seen clearly in the computations. Addition of terms of the form Eq. (23) penalizes large changes of ψ between successive frames. As explained, the results from $n = 1, 2, 3$ and 4 for the gyre image sequence are shown in images (a), (b), (c) and (d) in Fig. 4.

The mean angular errors for $n = 1, 2, 3$ and 4 are 0.9837°, 0.9826°, 0.9807° and 0.9883° respectively. In this example, the accuracy of the algorithm improves until $n = 3$. In all the above reconstructions, the regularization parameter α was selected so that it minimizes the mean angular error and the parameter β was fixed at $\beta = 0.01$.

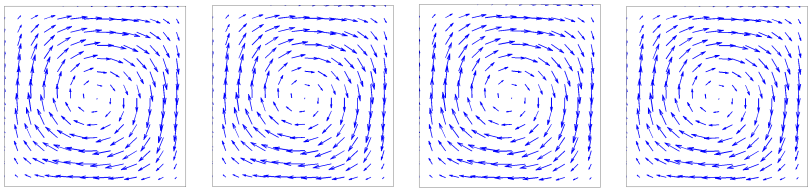


Fig. 4 Gyre flow from Multi-time step Method – Images (a), (b), (c) and (d) show vector fields computed on the image (b) in Fig.1 from multi-time step method with $n = 1, 2, 3$ and 4 respectively. The true flow field is shown in (a). While all estimated vector fields are visually similar, the mean angular errors are improving up to $n = 3$.

6.2 An oceanographic data-set

Now we apply the algorithm to a natural scenario which shows the sea surface temperature off the coast of Oregon, U.S.A. This data set was generated from an 3-D ocean model, using data obtained from Geostationary Operational Environmental Satellite (GOES) refers [1]. In Fig. 5, images (a), (b) and (c) show sea surface temperature data of three consecutive hours on August 1st, 2002. The image (d) represents the true vector field of the mixing temperature corresponding to the image (b).

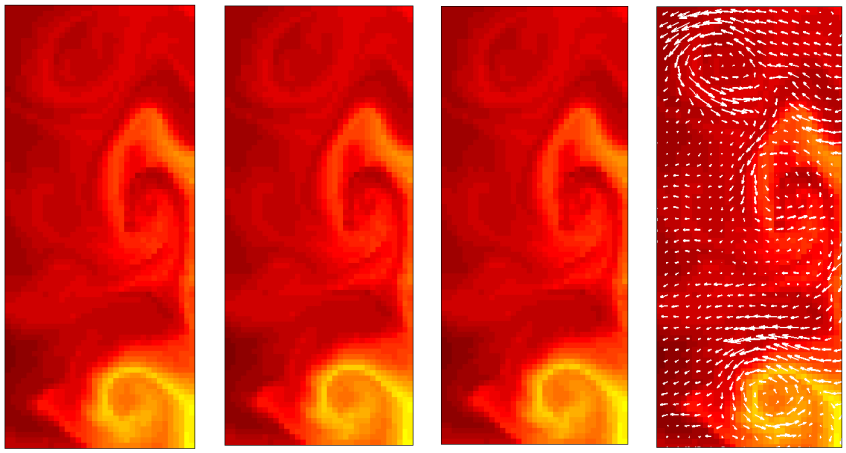


Fig. 5 SST data and true flow – Three consecutive images of the SST data set are shown in (a), (b) and (c) respectively, and the flow on image (b) is shown in the image (d)

Since we have a time dependent sequence of images of the SST data, we can apply the multi-time step method to compute the vector fields. When the step size is n, we compute n consecutive vector fields at a time and required $n+4$ images, if we apply fourth-order finite difference approximations to compute I_t. The Fig. 6 represents the computed flow fields on the image (b) shown in Fig. 5 with the step size $n = 1, 2, 3$ and 4 in images (a), (b), (c), and (d) respectively. In each case, the algorithm captures the gyres accurately and it is clearly visible when $n = 3$, the algorithm captures the laminar flow as appears just above the bottom gyre. Except for the above laminar flow, all the other vectors represent a similar behavior and the differences are not visible. Now we can compare the two vector fields consisting of large numbers of vectors by comparing single numbers and in the comparison of the step sizes; we use the percentage mean angular error.

The Fig. 7 shows the graph of percentage mean angular error of the computed flow relative to the true flow versus the step size used to compute the flow fields. According to the graph, the percentage mean angular error fluctuates and the minimum is achieved when $n = 3$ as we can see in the flow fields. Note that, for all the reconstructions in Fig. 5, the regularization parameter α was selected so that it minimizes the mean angular error and the parameter β was fixed at $\beta = 0.01$.

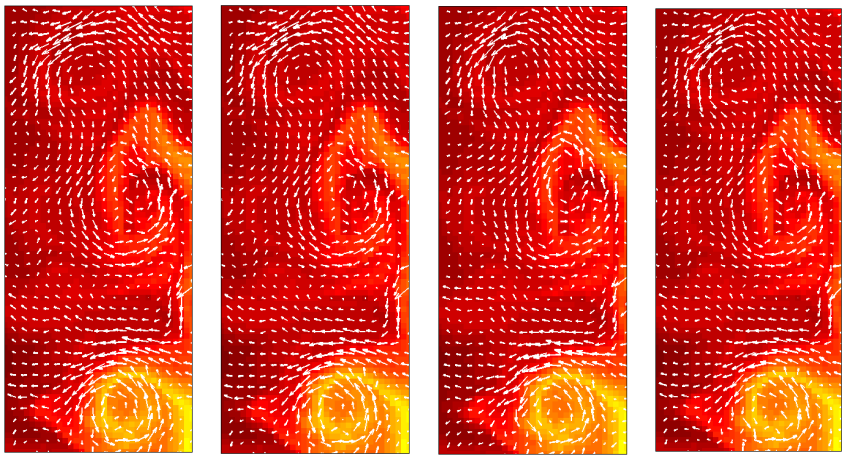


Fig. 6 SST Flow – The computed flow fields for the data showed in 5 with n equals to 1, 2 and 3 are shown in (a), (b), (c) and (d) respectively. While all these are roughly similar and so not immediately different to visual inspection, there are visible differences that are apparent in closer inspection.

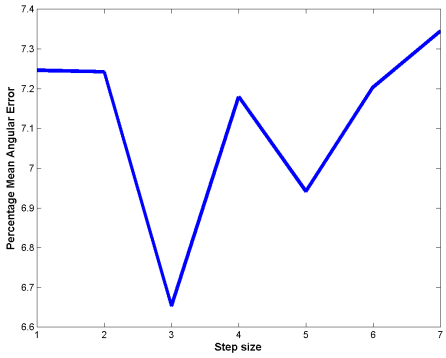


Fig. 7 Percentage MAE vs Step size – The graph shows the percentage of the mean angular error for the computed flow by changing the step size n on the image (b) shown in Fig. 5. For the parameters as we specified, the $n = 3$, multi-time step method is overall best.

7 Mixing and transport barriers

As an application, we will now discuss a transport analysis inferred directly from observed spatio-temporal movie data. Toward the identification of mixing and transport barriers, we compute finite time Lyapunov exponents. These are scalar values for each point in the domain D as explained in [8,9] and obtained LCS. In this case, for a given point $\mathbf{x} = \langle x(t), y(t)\rangle$, a flow map ϕ_T is obtained by evolving $\mathbf{x}$ over a time period $[t, t+T]$ using the velocity components $\mathbf{v} = \langle u(x,y,t), v(x,y,t)\rangle$. Then the Jacobian matrix of the flow map ϕ_T is obtained $J = \frac{d\phi_T(\mathbf{x})}{d\mathbf{x}}$ and therefore the finite time strain tensor of $\mathbf{v} = \langle u(x,y,t), v(x,y,t)\rangle$ along the trajectory $\mathbf{x} = \langle x(t), y(t)\rangle$ is obtained as

$$M = \frac{d\phi_T(\mathbf{x})^*}{d\mathbf{x}} \frac{d\phi_T(\mathbf{x})}{d\mathbf{x}}, \tag{39}$$

where A^* is the adjoint of A. Then the FTLE value at a point $\mathbf{x}$ over time T is given by

$$\sigma^T = \frac{1}{|T|} ln\sqrt{\lambda_{max}(M)}. \tag{40}$$

When the FTLEs are computed for the entire domain of the system, the set of points in the domain corresponding to relatively high FTLE values are suggested to act as pseudo barriers for mixing and transport of the system although it is known that some can be simplified due to shear behavior. For instance, if we consider the double gyre with the stream function $\psi(x,y,t) = C\sin(\pi f(x,t))\sin(\pi y)$, where $f(x,t) = \varepsilon\sin(\omega t)x^2 + (1-2\varepsilon\sin(\omega t))x$ over the domain $D = [0,2]\times[0,1]$, the corresponding vector field can be obtained as

$$\langle u, v\rangle = \langle -\pi C\sin(\pi f(x,t))\cos(\pi y), \pi C\cos(\pi f(x,t))\sin(\pi y)\frac{\partial f}{\partial x}\rangle, \tag{41}$$

where the constants are $C = 0.1, \omega = \frac{2\pi}{10}$ and $\varepsilon = 0.25$. The vector field that represents $\langle u, v \rangle$ at $t = 0$ and the computed FTLE field using $\langle u, v \rangle$ with $T = 10$ are shown in the Fig. 8. The red color represents relatively high FTLE values and the blue color represents the relatively small FTLE values. In [18], the authors have proven that flux across the LCS is close to zero and hence it is difficult for fluids in the system to cross the LCS. Therefore these LCSs act as barriers to the mixing and transport.

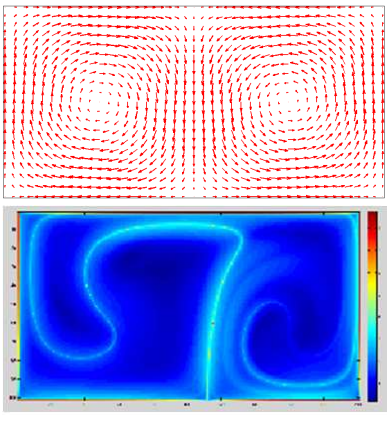
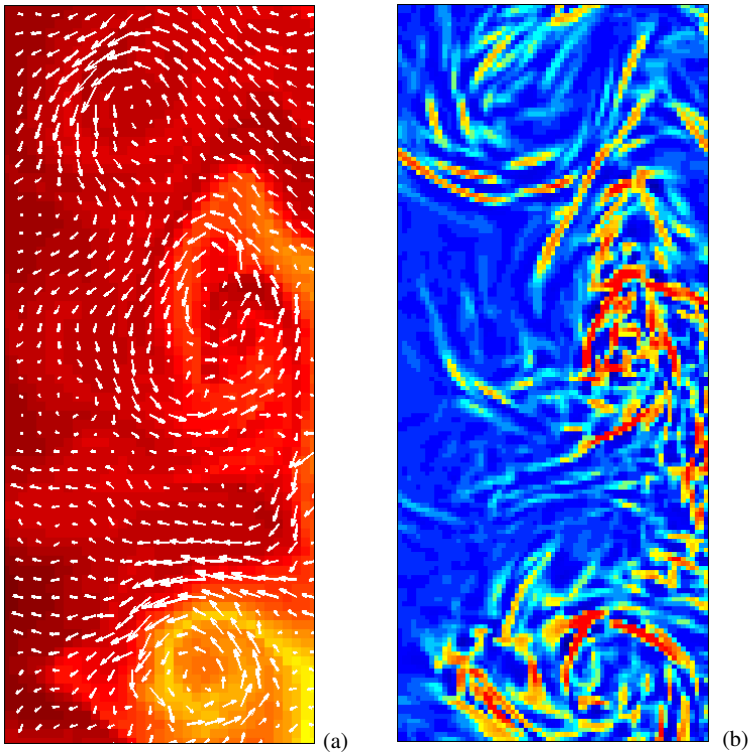


Fig. 8 Double gyre flow and FTLE field – The flow field for the non-autonomous double gyre at time $t = 0$ and the computed FTLE field for $T = 10$ are shown in images (a) and (b) respectively

The next example represents the FTLE field for the SST data set using the computed flow from the multi time step method with $n = 3$. In Fig. 9, the flow field obtained for the SST data on August 2nd, 2002 is shown in image (a) and the FTLE field obtained from those computed vector fields is shown in (b). When we compute FTLE fields, we need more than three time adjacent vector fields. Since we have a sequence of images, we apply multi-time step method on the image sequence to obtain a sequence of non-autonomous vector fields. If we use fourth order finite difference approximations to estimate I_t, we need first 7 images of the sequence to compute three time adjacent vector fields on images 3, 4 and 5 respectively. To compute the vector fields on images 6, 7 and 8, we apply multi-time step method again on images 4 to 10, and we can continue the procedure until we reach the end of the image sequence. In this case, we evolve 10 time steps forward in time to get the FTLE field. The blue color represents relatively low FTLE values and the red color represents relatively high FTLE values. The red color ridges, LCSs, act as the mixing barriers to the heat.



(a) (b)

Fig. 9 SST flow and FTLE field – The computed flow field for the SST data and the FTLE field are shown in images (a) and (b) respectively

8 Conclusion

We have developed a new optical flow algorithm to extract vector fields of an observed system from spatio-temporal (movie) data. When a sequence of images is available for an observed system, we can emphasize regularity not only spatially, but also in time. In spatial regularization, we emphasize known physics of both the data and the expected flow, and in regularization in time we emphasize the similar behavior of the flow fields. From these computed vector fields, we can analyze dynamics of the system by computing Lagrangian coherent structures. However, according to the results shown in Sect. 6.2, when the number of frames n simultaneously used in our new multi-time step method increases, the accuracy of the algorithm improves. Furthermore, we use the same regularization parameter to compare the results and it does not need to be the same. Our future goal is to improve the algorithm to select the best step size n and the suitable regularization parameter.

References

1. Office of sattellite operation (2011). URL http://www.oso.noaa.gov/goes/
2. Basnayake, R., Luttman, A., Bollt, E.: A lagged diffusivity method for computing total variation regularized fluid flow. Contemporary Mathematics **586**, 59–66 (2013)
3. Bollt, E., Luttman, A., Kramer, S., Basnayake, R.: Measurable dynamics analysis of transport in the Gulf of Mexico during the oil spill. International Journal of Bifurcation and Chaos **22**(03) (2012)
4. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: T. Pajdla, J. Matas (eds.) Proc. 8th Eur. Conf. on Computer Vision, vol. 4, pp. 25–36 (2004)
5. Corpetti, T., Mémin, E., Pérez, P.: Adaptation of standard optic flow methods to fluid motion. In: 9th Int. Symp. Flow Visualisation, pp. 1–10 (2000)
6. Froyland, G., Santitissadeekorn, N., Monahan, A.: Transport in time-dependent dynamical systems: Finite-time coherent sets. arXiv preprint arXiv:1008.1613 (2010)
7. Gelfand, I., Fomin, S.: Calculus of variations. Dover publications (2000)
8. Haller, G.: Lagrangian coherent structures from approximate velocity data. Physics of fluids **14**, 1851 (2002)
9. Haller, G., Poje, A.: Finite time transport in aperiodic flows. Physica D: Nonlinear Phenomena **119**(3-4), 352–380 (1998)
10. Hansen, P.C.: The L-curve and its use in the numerical treatment of inverse problems. In: P. Johnston (ed.) Computational Inverse Problems in Electrocardiology, Advances in Computational Bioengineering, pp. 119–142. WIT Press (2000)
11. Horn, B.K.P., Schunck, B.G.: Determining optical flow. Artificial Intelligence **17**, 185–203 (1981)
12. Kabanikhin, S.I.: Inverse and Ill-Posed Problems: Theory and Applications, vol. 55. De Gruyter (2011)
13. Krawczyk-StańDo, D., Rudnicki, M.: Regularization parameter selection in discrete ill-posed problemsthe use of the U-curve. International Journal of Applied Mathematics and Computer Science **17**(2), 157–164 (2007)
14. Krawczyk-Stańdo, D., Rudnicki, M.: The use of L-curve and U-curve in inverse electromagnetic modelling. Intelligent Computer Techniques in Applied Electromagnetics pp. 73–82 (2008)
15. Luttman, A., Bollt, E., Basnayake, R., Kramer, S.: A stream function approach to optical flow with applications to fluid transport dynamics. PAMM **11**(1), 855–856 (2011)
16. McCane, B., Novins, K., Crannitch, D., Galvin, B.: On benchmarking optical flow. Computer Vision and Image Understanding **84**, 126–143 (2001). DOI 10.1006/cviu.2001.0930
17. Schmidt, M., Fung, G., Rosales, R.: Fast optimization methods for l1 regularization: A comparative study and two new approaches. Machine Learning: ECML 2007 pp. 286–297 (2007)
18. Shadden, S., Lekien, F., Marsden, J.: Definition and properties of Lagrangian coherent structures from finite-time Lyapunov exponents in two-dimensional aperiodic flows. Physica D: Nonlinear Phenomena **212**(3), 271–304 (2005)
19. Vogel, C.R.: Computational Methods for Inverse Problems. Frontiers in Mathematics. SIAM (2002)
20. Weickert, J., Bruhn, A., Papenberg, N., Brox, T.: Variational optic flow computation: From continuous models to algorithms. In: L. Alvarez (ed.) International Workshop on Computer Vision and Image Analysis, IWCVIA03, Las Palmas de Gran Canaria (2003)