Recovering Fluid-type Motions Using Navier-Stokes Potential Flow

Feng Li[1] Liwei Xu[2] Philippe Guyenne[2] Jingyi Yu[1]
[1]Department of Computer and Information Sciences
University of Delaware
Newark, DE 19716, USA
{feli,yu}@cis.udel.edu
[2]Department of Mathematical Sciences
University of Delaware
Newark, DE 19716, USA
{xul,guyenne}@math.udel.edu

Abstract

The classical optical flow assumes that a feature point maintains constant brightness across the frames. For fluid-type motions such as smoke or clouds, the constant brightness assumption does not hold, and accurately estimating the motion flow from their images is difficult. In this paper, we introduce a simple but effective Navier-Stokes (NS) potential flow model for recovering fluid-type motions. Our method treats the image as a wavefront surface and models the 3D potential flow beneath the surface. The gradient of the velocity potential describes the motion flow at every voxel. We first derive a general brightness constraint that explicitly models wavefront (brightness) variations in terms of the velocity potential. We then use a series of partial differential equations to separately model the dynamics of the potential flow. To solve for the potential flow, we use the Dirichlet-Neumann Operator (DNO) to simplify the 3D volumetric velocity potential to 2D surface velocity potential. We approximate the DNO via Taylor expansions and develop a Fourier domain method to efficiently estimate the Taylor coefficients. Finally we show how to use the DNO to recover the velocity potential from images as well as to propagate the wavefront (image) over time. Experimental results on both synthetic and real images show that our technique is robust and reliable.

1. Introduction

Optical flow is one of the most studied problems in computer vision. It refers to the process of calculating the motion between two frames. Let $\eta(\mathbf{x}, t)$ [1] denote the intensity at pixel $\mathbf{x} = (x, y)$ at time t. Assuming that the brightness of every feature point remains constant across the frames and its movement is small, we have

$$\frac{d\eta}{dt} = \frac{\partial \eta}{\partial t} + (\frac{\partial \eta}{\partial x}, \frac{\partial \eta}{\partial y}) \cdot \mathbf{u} = 0 \tag{1}$$

where $\mathbf{u} = (u, v)^\top = (dx/dt, dy/dt)^\top$ denotes the optical flow, i.e., the instantaneous velocity between two consecutive frames.

It is well understood that the optical flow problem is inherently under-constrained: there are more unknowns (u and v) than constraint (Eq. (1)). Additional constraints, hence, have been used to resolve the ambiguity problem. These include algorithms based on phase correlation [8], smooth prior [4], spatial statistics prior [20], derivative constraints of the image or the flow field [2, 19]. We refer the readers to the optical flow evaluation database by Baker et al [3] for a complete review.

Recent optical flow methods have been focused on recovering fluid-type motions such as clouds, ocean/river waves, and smoke. These natural phenomena do not satisfy the constant brightness constraint. For example, in ocean engineering [13], irregular and multidirectional properties of waves force both the smoothness and discontinuity in image brightness to change along the physical dynamics. Therefore, relaxing the constant brightness constraint based on linear illumination model [9] often produces large errors.

To recover fluid-type motions, a number of approaches have been proposed to integrate the basic optical flow solution with fluid dynamics constraints, e.g., the continuity equation that describes the fluid property [6, 17, 23] or the divergence-curl (div-curl) equation [1, 6] to describe spreading and rotation. Many previous approaches [5, 10, 15, 16] have shown that the constant brightness constraint is violated for fluid-type motions. Their solutions were to treat the physical model as regularization terms to the constant brightness constraint. Our work is inspired by the physical-based brightness variation model [11] for computing the optical flow under illumination changes and heat transport. Our goal is to directly model brightness variations caused by fluid dynamics. Other methods attempt to

---

[1] Since our approach treats the image as a surface, we use η instead of I for consistency.

derive the closed-form flow model using base waves [21]. Motion flow estimation can then be formulated as to find their coefficients to match the input sequence. In practice, a large number of base waves are required to faithfully model the flow. Finding their coefficients is not only computationally expensive but also unreliable.

In this paper, we introduce a novel Navier-Stokes potential flow framework for estimating fluid-type motions from images. Our method treats image intensity as a wavefront surface and models the 3D *velocity potential* beneath the surface. The gradient of the velocity potential describes the motion flow at every voxel. Unlike previous methods that regularize the constant brightness constraint, we derive a *general brightness constraint* that explicitly models brightness variations in terms of the velocity potential. We then use a set of partial differential equations (PDEs) to separately model the dynamics of the velocity potential. These include a fluid dynamics constraint for the wavefront, a Neumann boundary condition at the fluid bottom, and a Laplace equation for mass conservation.

To solve for the velocity potential, we use the Dirichlet-Neumann Operator (DNO) to simplify the 3D volumetric velocity potential to 2D surface velocity potential. In particular, it defines two PDEs that separately model how the wavefront evolves and how the motion flow propagates. We approximate the DNO via Taylor expansions and develop a Fourier domain method to efficiently estimate the Taylor coefficients. Finally we show how to use the DNO to estimate the velocity potential from two frames and to propagate the wavefront over time. We demonstrate our techniques on synthetic and real images of waves, smoke, and clouds. We show that our NS potential flow model surpasses existing optical flow methods. For example, the smooth and discontinuous clouds motions estimated by our method is coherent with the image sequence, whereas the previous methods estimate only smooth and uniform motion.

2. Navier-Stokes Potential Flow

Our approach models the image as a wavefront surface $\eta(\mathbf{x}, t)$, where $\mathbf{x} = (x, y)$ represent the pixel coordinates and t represents time. We use z to represent the height dimension. While traditional optical flow has focused on estimating 2D flow field on surface η, we reformulate the problem to solve for the 3D flow field beneath η, as shown in Fig. 1. Specifically, we aim to estimate the 3D motion $\tilde{\mathbf{u}}$ as:

$$\tilde{\mathbf{u}} = (u, v, w)^\top = (dx/dt, dy/dt, dz/dt)^\top ,$$

where u, v and w represent the velocity along the x, y and z directions respectively. The classical 2D optical flow $\mathbf{u}$, hence, can be viewed as the horizontal projection of $\tilde{\mathbf{u}}$ on the wavefront $z = \eta$.

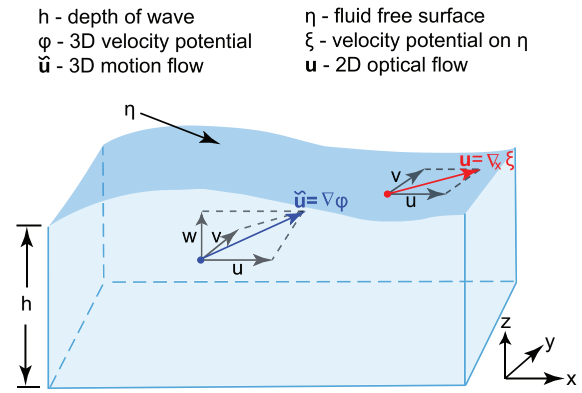

Figure 1. Navier-Stokes Potential Flow. We model the image as a height field η and study its 3D velocity potential φ. The gradient of φ describes the 3D motion flow $\tilde{u}$. The 2D optical flow u can be viewed as the 2D gradient of the surface potential ξ on η.

2.1. Potential Flow

We first introduce the notion of *potential (or irrotational) flow* [24] which describes the (3D) motion flow field $\tilde{\mathbf{u}}$ as the gradient of the velocity potential φ:

$$\tilde{\mathbf{u}} = \nabla \varphi \, , \tag{2}$$

where $\nabla = (\partial_x, \partial_y, \partial_z)^\top$ is the gradient operator. On wavefront η, we have $z = \eta(\mathbf{x}, t)$, therefore,

$$\frac{d}{dt}(\eta(\mathbf{x}, t) - z) = \frac{\partial \eta}{\partial t} + \frac{\partial \eta}{\partial x}\frac{dx}{dt} + \frac{\partial \eta}{\partial y}\frac{dy}{dt} - \frac{dz}{dt} = 0 \tag{3}$$

We can rewrite Eq. (3) as

$$\frac{\partial \eta}{\partial t} - (-\nabla_{\mathbf{x}}\eta, 1)^\top \cdot \nabla\varphi = 0\big|_{z=\eta} \, ,$$

or

$$\frac{\partial \eta}{\partial t} = (-\nabla_{\mathbf{x}}\eta, 1)^\top \cdot \nabla\varphi\big|_{z=\eta} \, , \tag{NS.1}$$

where $\nabla_{\mathbf{x}} = (\partial_x, \partial_y)^\top$ is the gradient in the (x, y)-plane. Recall that $(-\nabla_{\mathbf{x}}\eta, 1)^\top$ is the normal vector at each point on wavefront η, Eq. (NS.1) reveals that wavefront (brightness) variation is determined by the normal flow.

We call Eq. (NS.1) the *general brightness constraint* as it expresses image brightness variations in terms of the velocity potential φ. The constant brightness constraint (Eq. (1)) is a special case of Eq. (NS.1) when $\partial\varphi/\partial z$, the motion along the z direction, is zero. More importantly, the *general brightness constraint* allows us to separately model φ via specific physics models.

2.2. Fluid Dynamics Constraints

In this paper, we model φ using the fluid dynamic constraints. Recall the wavefront η satisfies the Navier-Stokes (NS) equation:

$$\rho(\partial_t\tilde{\mathbf{u}} + \tilde{\mathbf{u}} \cdot \nabla\tilde{\mathbf{u}}) = -\nabla p + \mu\nabla^2\tilde{\mathbf{u}} + \mathbf{f} \tag{4}$$

where ρ is the fluid density, p is the pressure, μ is the viscosity, and $\mathbf{f}$ is body forces. For ocean or river waves, we have $\mathbf{f} = \nabla(-\rho g z)$, where g is the gravitational acceleration.

To simplify Eq. (4), we assume the fluid is inviscid, i.e., $\mu = 0$, and irrotational, i.e., the convective acceleration term $\tilde{\mathbf{u}} \cdot \nabla \tilde{\mathbf{u}} = \nabla(\frac{|\tilde{\mathbf{u}}|^2}{2})$. Finally, we assume there is no pressure jump across η, i.e., $\nabla p = 0$, and ρ is constant. Notice that these assumptions hold for many natural fluid-type motions such as smoke, clouds, waves, etc. The NS equation, Eq. (4), hence, can be simplified as:

$$\rho(\partial_t \tilde{\mathbf{u}} + \nabla(\frac{|\tilde{\mathbf{u}}|^2}{2})) + \nabla(\rho g z) = 0 \tag{5}$$

Substituting $\tilde{\mathbf{u}} = \nabla\varphi$ into Eq. (5) and evaluating on η, we have

$$\partial_t \varphi + \frac{1}{2}|\nabla\varphi|^2 + g\eta = 0 \quad \text{on} \quad z = \eta\,. \tag{NS.2}$$

We call Eq. (NS.2) the *NS constraint* for the potential flow.

For most natural fluids such as a liquid, or a gas at low Mach numbers, it is common to assume that they are incompressible, i.e., $\nabla \cdot \tilde{\mathbf{u}} = 0$ which implies that the velocity potential satisfies Laplace's equation:

$$\nabla^2 \varphi = 0\,, \tag{NS.3}$$

for all voxels beneath η.

Finally, we introduce the boundary constraint at the bottom of the fluid $z = -h$. We assume that there is no vertical flow (in the z direction) at the bottom, i.e., the fluid bottom is impermeable:

$$\partial_z \varphi = 0 \quad \text{on} \quad z = -h\,. \tag{NS.4}$$

We call Eqs. (NS.2)–(NS.4) the fluid dynamics constraints of φ. Together with the *general brightness constraint* Eq. (NS.1), they form the Navier-Stokes potential flow. The main advantage of our formulation is that the *general brightness constraint* is separated from the dynamics constraints. Although our analysis is focused on fluid-type motions, we can modify or even replace the fluid dynamics constraints with other physical models.

2.3. Limitations

Our NS potential flow model does not include all the characteristics of fluids that are encountered in the real world. For example, since we model the wave surface as a height field, it cannot be a multi-valued function of x and y. Thus, our model excludes turbulence, which is commonly encountered in nature. Incompressible potential flow also makes a number of invalid predictions, e.g., d'Alembert's paradox, which states that the drag on any object moving through an infinite fluid otherwise at rest is zero.

Finally, we simplify the convective acceleration term in the Navier-Stokes equation by assuming the flow is irrotational, i.e., the vorticity is zero. This is a major limitation of our model compared with the closed-form representations [21], where rotational flows can be effectively modeled with a sufficiently large number of base waves. However, it is important to note that the vorticity only acts as a measure of the *local* rotation of fluid elements and it does not imply any global behavior of a fluid: it is possible for a fluid traveling in a straight line to be rotational, and the one moving in a circle to be irrotational. Therefore, the NS potential flow can still model rotational flow motions to some extent.

3. Solving for the NS Potential Flow

The fluid dynamics constraints Eqs. (NS.2)–(NS.4) and the *general brightness constraint* (NS.1) form an over-constrained system. In theory, we can discretize φ and η and solve for them using finite element/finite difference methods. In practice, since φ is defined everywhere inside the volume beneath η, one needs to discretize the volume fine enough to achieve numerically accurate results. Furthermore, solving for such a large system is computationally prohibitive.

Recall that our ultimate goal is to compute the optical flow on wavefront η, therefore we present a new technique to simplify the 3D volumetric velocity potential to the 2D surface velocity potential. Specifically, we introduce a new variable $\xi(\mathbf{x}, t) = \varphi(\mathbf{x}, \eta(\mathbf{x}, t), t)$ to represent the velocity potential on η. We then use the Dirichlet-Neumann operator (DNO) [24] to simplify the normal flow as:

$$(-\nabla_{\mathbf{x}}\eta, 1)^\top \cdot \nabla\varphi\big|_{z=\eta} = G(\eta)\xi\,. \tag{6}$$

The DNO takes Dirichlet data ξ on η, solves Laplace's equation (NS.3) for φ together with the no-flow bottom condition (NS.4), and returns the corresponding Neumann data, i.e., the fluid normal velocity on η.

Under the DNO, the *general brightness constraint* (NS.1) reduces to:

$$\partial_t \eta = G(\eta)\xi\,, \tag{7}$$

and the fluid dynamics constraint (NS.2) becomes:

$$\begin{aligned} \partial_t \xi \;=\; & -g\eta - \frac{1}{2(1+|\nabla_{\mathbf{x}}\eta|^2)}\Big[|\nabla_{\mathbf{x}}\xi|^2 - (G(\eta)\xi)^2 \\ & -2(G(\eta)\xi)\nabla_{\mathbf{x}}\xi \cdot \nabla_{\mathbf{x}}\eta + |\nabla_{\mathbf{x}}\xi|^2|\nabla_{\mathbf{x}}\eta|^2 \\ & -(\nabla_{\mathbf{x}}\xi \cdot \nabla_{\mathbf{x}}\eta)^2\Big]\,. \end{aligned} \tag{8}$$

The other two constraints Eqs. (NS.3)–(NS.4) are already incorporated through the DNO.

We emphasize that Eqs. (7)–(8) describe fully nonlinear and fully dispersive fluid dynamics. In particular, the waves

```
1. Compute the Fourier transform $F(q)$.
2. Multiply $q_1 = k_x\,(k_x^2 + k_y^2)^{-1/2}\,F(q)$ in
      Fourier space, where $k_x$ and $k_y$ are wave
      numbers in the $x$ and $y$ directions respectively.
3. Compute the inverse Fourier transform $F^{-1}(q_1)$.
4. Multiply $q_2 = \eta\,F^{-1}(q_1)$ in physical space.
5. Compute the Fourier transform $F(q_2)$.
6. Multiply $q_3 = k_x\,(k_x^2 + k_y^2)^{-1/2}\,F(q_2)$ in
      Fourier space.
7. Compute the inverse Fourier transform $F^{-1}(q_3)$.
```

Figure 2. Algorithm 1: Computing $|D|^{-1} D_x\,\eta\,D_x |D|^{-1} q$.

are not assumed to be small nor weakly dispersive. This formulation of the flow model also has many practical advantages. First, the reduction of dimensionality implies that the wavefront η is determined explicitly from the equations of the problem, and thus needs not be reconstructed a posteriori. Moreover, the Laplace problem Eqs. (NS.2)–(NS.4) need not be solved directly, which saves us from solving a linear system. Second, given the initial wavefront η_0 and velocity potential ξ_0, we can directly propagate η and ξ using Eqs. (7)–(8); whereas given two wavefronts η_0 and η_1, as in the typical case of optical flow estimation, we can compute the velocity potential ξ_0 from Eq. (7) (Section 3.2). Finally, different from existing fluid models [1, 6, 21], our flow model possesses a Hamiltonian structure [25]:

$$\partial_t \eta = \delta_\xi H \ , \quad \partial_t \xi = -\delta_\eta H \ ,$$

with the Hamiltonian

$$H = \frac{1}{2} \iint \Big[\xi G(\eta)\xi + g\eta^2\Big]\, d\mathbf{x} \tag{9}$$

where the notation δ_j is shorthand for the variational derivative with respect to the subscript j. The Hamiltonian formulation naturally associates the energy H which is an important conserved quantity in the dynamics of the system.

3.1. Approximate the DNO

Next, we show how to approximate the DNO for practical use. To this aim, we adopt an operator expansion approach based on the fact that, under the regularity condition on η, the DNO is an analytic function of η. It follows that the DNO can be written in terms of a convergent Taylor series

$$G(\eta) = \sum_{j=0}^{\infty} G_j(\eta) \ , \tag{10}$$

where the Taylor polynomials G_j are homogeneous of degree j in η and can be obtained by a recursion formula [7, 24]. For $j = 2r > 0$,

$$\begin{aligned} G_{2r}(\eta) &= \frac{1}{(2r)!} G_0 (|D_{\mathbf{x}}|^2)^{r-1} D_{\mathbf{x}} \cdot \eta^{2r} D_{\mathbf{x}} \\ &- \sum_{s=0}^{r-1} \frac{1}{(2(r-s))!} (|D_{\mathbf{x}}|^2)^{r-s} \eta^{2(r-s)} G_{2s}(\eta) \\ &- \sum_{s=0}^{r-1} \frac{1}{(2(r-s)-1)!} G_0 (|D_{\mathbf{x}}|^2)^{r-s-1} \eta^{2(r-s)-1} G_{2s+1}(\eta) \ , \end{aligned} \tag{11}$$

and, for $j = 2r - 1 > 0$,

$$\begin{aligned} G_{2r-1}(\eta) &= \frac{1}{(2r-1)!} (|D_{\mathbf{x}}|^2)^{r-1} D_{\mathbf{x}} \cdot \eta^{2r-1} D_{\mathbf{x}} \\ &- \sum_{s=0}^{r-1} \frac{1}{(2(r-s)-1)!} G_0 (|D_{\mathbf{x}}|^2)^{r-s-1} \eta^{2(r-s)-1} G_{2s}(\eta) \\ &- \sum_{s=0}^{r-2} \frac{1}{(2(r-s-1))!} (|D_{\mathbf{x}}|^2)^{r-s-1} \eta^{2(r-s-1)} G_{2s+1}(\eta) \ , \end{aligned} \tag{12}$$

where $D_{\mathbf{x}} = -i\nabla_{\mathbf{x}}$ and $G_0 = G(0) = |D_{\mathbf{x}}| \tanh(h|D_{\mathbf{x}}|)$ represent Fourier multiplier operators.

The Taylor expansion allows us to evaluate the DNO in its series form Eq. (10) using the recursion formulas Eqs. (11)–(12). These formulas can be easily implemented and efficiently computed by the fast Fourier transform.

3.2. Potential Flow Estimation and Wavefront Propagation

To estimate the velocity potential between two wavefronts (images) η_i and η_{i+1}, we can first compute $\partial_t \eta_i = (\eta_{i+1} - \eta_i)/\Delta t$ (where Δt is the time step between two consecutive frames) and then invert the DNO G in Eq. (7) to compute ξ_i as:

$$\xi_i = G(\eta_i)^{-1} \partial_t \eta_i \ . \tag{13}$$

To compute the inverse DNO $G(\eta)$, we again use the Taylor expansion of the operator. In our experiments, we find it is often sufficient to just use the first two terms in the expansion since the analyticity of G implies fast convergence of its Taylor series, thus

$$\begin{aligned} G(\eta)^{-1} &= G_0^{-1} - G_0^{-1} D_{\mathbf{x}} \cdot \eta D_{\mathbf{x}} G_0^{-1} + \eta \\ &= |D_{\mathbf{x}}|^{-1} - |D_{\mathbf{x}}|^{-1} D_x\,\eta\,D_x |D_{\mathbf{x}}|^{-1} - |D_{\mathbf{x}}|^{-1} D_y\,\eta\,D_y |D_{\mathbf{x}}|^{-1} + \eta \end{aligned} \tag{14}$$

Moreover, since the function $\tanh(x)$ rapidly reaches ± 1 as $x \to \pm\infty$, we have

$$G_0 = |D_{\mathbf{x}}| \ . \tag{15}$$

To make our computation more efficient, we map $G(\eta)^{-1}$ in Fourier domain. This transforms complex partial derivative operators to functions of the wave number k_x and k_y in the x and y directions in Fourier space: D_x maps

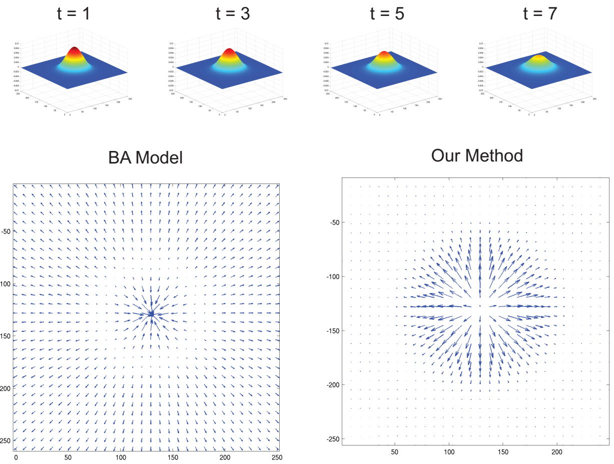

Figure 3. Motion Flow Estimation on Synthetic Data. Top row : Four frames ($t = 1, 3, 5, 7$) of an expanding Gaussian wave. Bottom row : The recovered motion flow between frame $t = 3$ and $t = 4$ using BA model [4] and our NS potential flow.

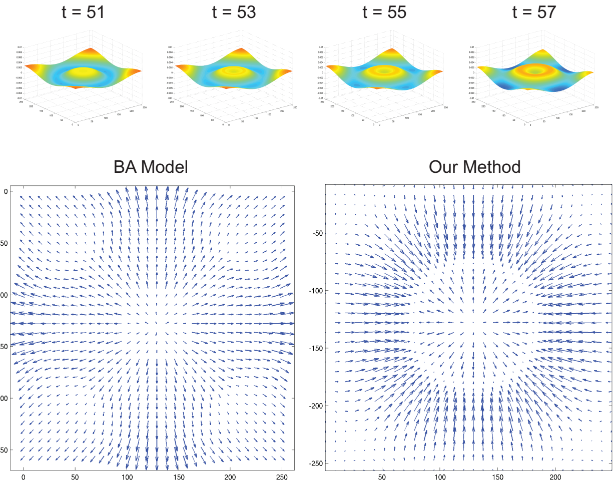

Figure 4. Motion Flow Estimation on Gaussian Wave. Top row: Four frames ($t = 51, 53, 55, 57$) of the same Gaussian wave as Fig. 3. Bottom row : The recovered motion flow between frame $t = 53$ and $t = 54$ using BA model [4] and our NS potential flow.

to k_x, D_y maps to k_y, and $|D_{\mathbf{x}}|$ maps to $\sqrt{k_x^2 + k_y^2}$. The middle two terms in Eq. (14) are then computed using the Fourier domain method as shown in Algorithm 1. We can further propagate the wavefront η_{i+1} to the next frame η_{i+2} using the estimated velocity potential ξ_i. We first compute the next velocity potential ξ_{i+1} by Eq. (8) and then apply Eq. (7) to obtain η_{i+2}.

4. Results

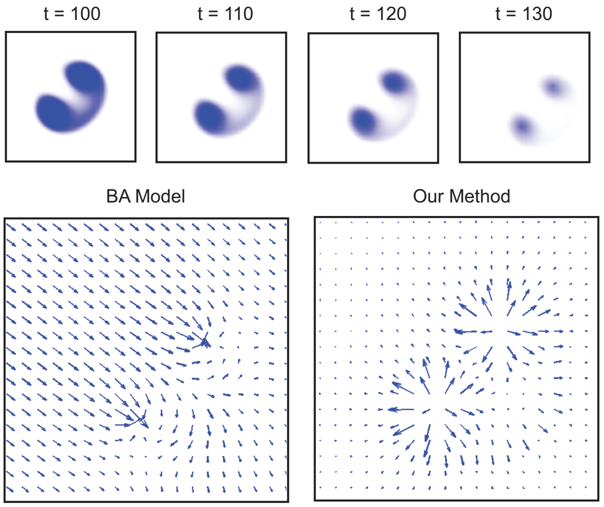

Figure 5. Motion Flow Estimation on a Gas Flow. Top row : Four frames of a gas flow simulated using the NVidia GPGPU Steady Flow SDK. Bottom row : The estimated optical flow by BA model [4] and our NS potential flow.

We have validated our NS potential flow framework on both synthetic and real images. To demonstrate the robustness of our method, we conduct all experiments using two consecutive images. We also include the preceding and the following frames to validate our estimated flow.

4.1. Gaussian Waves

Since it is difficult to quantify the errors of the estimated flow in real images, we first synthesize a test sequence with ground truth flow data.

We start with a synthetic Gaussian wave image with the initial wavefront $\eta_0(\mathbf{x}) = 0.01\, e^{-(x-\pi)^2-(y-\pi)^2}$ and velocity $\xi_0(\mathbf{x}) = 0$. The wave is propagated under gravity. We evolve the wave using the standard Navier-Stokes equation [12]. This creates a wavy fluid-type image sequence.

Bottom right of Fig. 3 shows our estimated motion flow between frame $t = 3$ and $t = 4$. Our NS potential flow faithfully captures the expanding flow motion. In contrast, the classical optical flow method [4] (BA model) estimates a contracting flow. This is because the constant brightness constraint forces the outward features to match the collapsing center.

Fig. 4 shows our NS potential flow result between frame $t = 53$ and $t = 54$, where the central fluid was expanding and the boundary fluid was contracting, forming a new ring-shaped band. Our NS-flow accurately estimates the flow field on both sides of the band whereas the BA model estimates uniformly expanding flow. We have further used the estimated flow between frame $t = 53$ and $t = 54$ to propagate the wavefront. The top row of Fig. 6 shows that our propagated wavefront at time instance $t = 58$ is consistent with the ground truth. Our NS potential flow algorithm is also highly efficient. On a Dell workstation with Intel(R) Pentium(R) D CPU 3.00GHz, it takes 0.37 second to com-

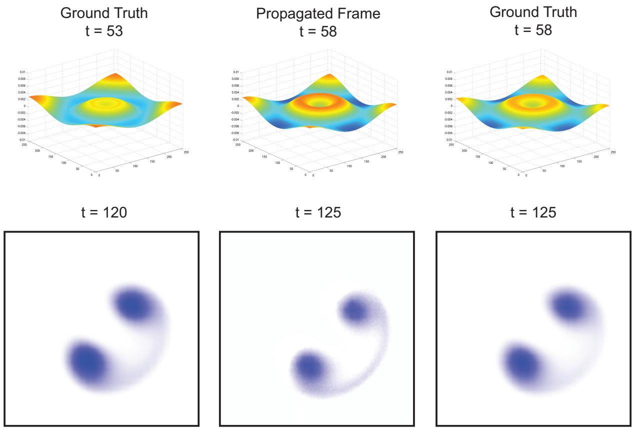

Figure 6. Wavefront Propagation Using the NS Potential Flow. Top row: Our synthesized frame $t = 58$ using the estimated flow between frame $t = 53$ and $t = 54$ on a Gaussian wave. Bottom row: Our synthesized new frame $t = 125$ using the flow between $t = 120$ and $t = 121$ on a gas flow.

pute the flow at a 256x256 resolution and 14.92 seconds to propagate the wavefront to the following frame.

4.2. Gas, Clouds, and Typhoon Images

Gas Flow. To test the robustness of our algorithm, we synthesize an irregularly shaped gas flow at a 512x512 resolution as shown in Fig. 5. We use the NVidia SDK 9.5 [18] for real-time Stable Fluids [22]. Synthetic blue dyes were added to better illustrate the flow motion. We first invert the intensity of the images and use the blue component as the wavefront height in our algorithm. The bottom row of Fig. 4 shows the estimated flows. Our NS potential flow method is able to capture both the global translational motion towards the bottom right and the local expanding motion of the dyes. The BA method only captures the global motion as shown in Fig. 4. Using the estimated motion between frame $t = 120$ and $t = 121$, we further propagate the image to frame $t = 125$. The bottom row of Fig. 6 compares our propagation result and the ground truth.

Clouds. We obtain the clouds image sequences (Fig. 7) from [14]. We first extract the alpha matte of the foreground clouds using blue screen matting. We then map the alpha matte to a height field. Notice that, although our NS constraints (Eqs. (NS.2)–(NS.4)) were initially derived for fluid surface height, they can also be used to model the density, and hence, the alpha matte.

The top row of Fig. 7 shows the alpha mattes of five consecutive frames from the clouds sequence. We apply our NS potential flow method to estimate the motion flow between frame $t = 13$ and $t = 14$. We include the preceding and following frames to validate our result. The BA model only detects translational flow inside the red area whereas our model estimates strong contracting motion. The image sequence confirms that the clouds get condensed over time inside the red region. Similarly, inside the yellow region, the BA model estimates a strong rightward flow whereas our estimated result reveals an expanding flow, as verified by the image sequence.

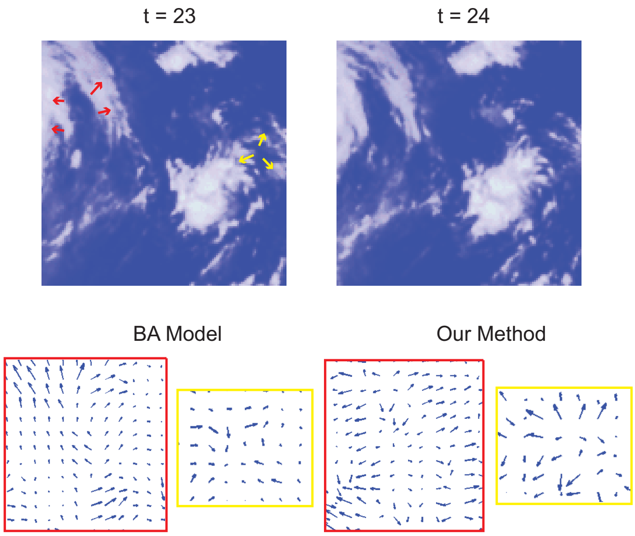

Figure 8. NS Potential Flow Estimation on a Typhoon Image Sequence. Top row: Two consecutive frames of the Typhoon satellite image. Bottom row: The estimated optical flow by the BA model [4] and our NS potential flow.

Infrared Satellite Images. Fig. 8 shows the optical flow results on Typhoon Haitang (0505) when it passed by the Taiwan Strait. The image sequence was captured by a Multi-functional Transport Satellite-1R (MTSAT-1R) of Japan Meteorological Agency (JMA) at one frame per hour. Similar to the clouds example, we first apply blue screen matting to extract the alpha mattes for each frame and estimate the motion flows between the alpha mattes. For clarity, we highlight the estimated optical flows inside two regions. The BA model detects an underestimated flow in the red and yellow regions whereas our NS potential flow method reveals an expanding flow in both regions.

4.3. Error Analysis

Notice that the motion flows of real images are usually complex. Therefore, we further present a scheme to measure the quality of the flow. Recall that our NS potential flow formulation allows us to both estimate the flow and to propagate the wavefront. Therefore, once we estimate the flow $\mathbf{u_1}$ (or ξ_1) between two frames η_1 and η_2, we can use it to synthesize η_2' and compare it with the ground-truth η_2. For the BA model, we can also warp η_1 to η_2'' using its estimated optical flow. Fig. 9 compares the ground truth η_2, our propagated η_2', the BA model warped η_2'' for the clouds and Typhoon examples. Our propagation results are more consistent with the ground truth.

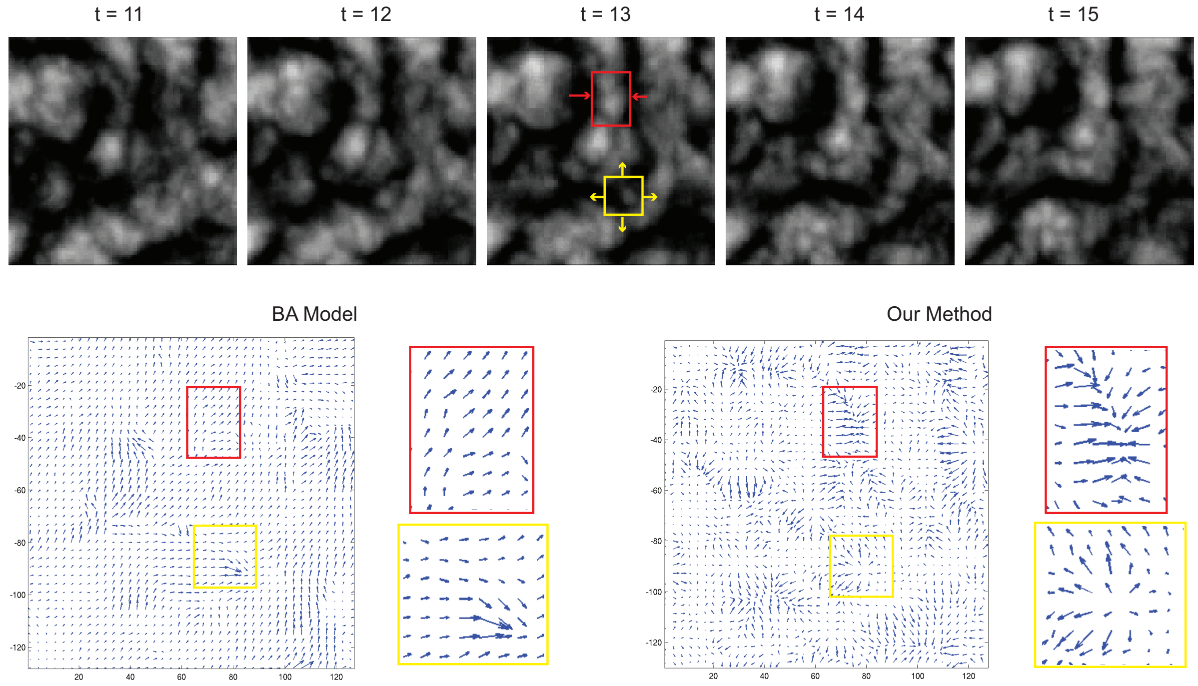

Figure 7. Motion Flow Estimation on a Clouds Sequence. The top row shows five consecutive frames. Bottom left: The motion flow estimated between frame $t = 13$ and $t = 14$ by the BA model [4]. Bottom right: Our NS potential flow results.

We can further quantitatively measure the errors between the ground truth and the synthesized frames by computing the difference image. Since fluid-type images contain large areas of uniform colors, a meaningful scheme should exclude such areas. Therefore, we only use the top 10% pixels with the largest values in the difference image and compute their averaged error. Table 1 compares the errors using the BA model and our method for the examples shown in this paper. Our method surpasses the BA model in all cases.

Notice that our matching errors are significantly larger on real images than on synthetic ones for several reasons. First, we only approximate $G(\eta)$ using the first two terms in its Taylor expansion and apply the fast Fourier transform to compute $G(\eta)^{-1}$. Thus, our approximation method works better on periodic and smooth synthetic data than on real images of inhomogeneous flows. Second, the real image sequences that we obtain are of poor quality and contain severe compression artifacts. Therefore, the estimated spatial derivatives are noisy. Finally, for real images, the transitions between the frames are not smooth, as shown in the Typhoon example (Fig. 8). This violates the basic optical flow assumption that brightness variations $\partial_t \eta$ can be approximated using finite differences.

5. Conclusion and Future work

We have presented a new Navier-Stokes (NS) potential flow method that explicitly models brightness variations in fluid-type motions. Our NS potential flow aims to recover the 3D velocity potential whose gradient describes the actual motion flow. We have derived a *general brightness constraint* that models brightness variations in terms of fluid dynamics of the velocity potential. To solve for the velocity potential, we have presented a Dirichlet-Neumann Operator (DNO) that simplifies the 3D volumetric flow to 2D

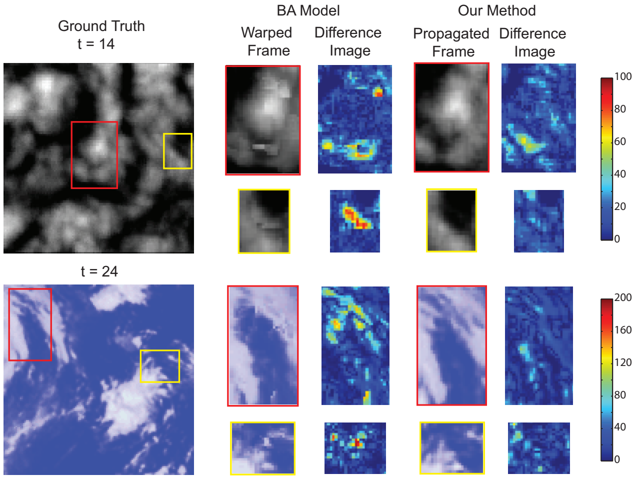

Figure 9. Error Analysis on the Warped/Propagated Results. Left: The ground truth second frame used in computing the optical flow. Middle: The warped second frame using the optical flow estimated by the BA model. Right: The propagated second frame using our estimated optical flow.

	Gaussian1	Gaussian2	Gas	Clouds	Satellites
BA model	2.30282	6.31217	4.29739	38.14344	74.23475
our method	0.93173	0.85458	2.50324	35.90740	61.01021

Table 1. Quantitative Error on the Warped/Propagated Results. The maximum wavefront height was normalized to 255 in all examples.

surface flow. We have shown that the DNO can be effectively approximated via Taylor expansions and computed in the Fourier space. The DNO can be directly applied to estimate the velocity potential from images and to propagate the wavefront.

Our NS-flow formulation has several advantages compared with existing fluid-based optical flow techniques. First, our method does not use complex optimization schemes for flow estimation whereas many existing methods need either optimize over a large number of parameters [21] or solve over-constrained systems [1, 6]. Second, our fluid dynamics model is numerically accurate, efficient, and stable. Finally, since our general brightness constraint separately treats potential flow dynamics and brightness variations, one can replace the fluid dynamics model with other physics models and reuse the same solution process.

For future work, we plan to integrate our NS potential flow model with fluid surface acquisition. Most existing fluid acquisition methods have been focused on reconstructing individual frames and very little has been done to enforce temporal coherence of reconstructed surface. Since our NS potential flow provides effective means to both estimate the flow and to propagate the wavefront, it may be directly applied to measure and possibly improve the quality of the acquired surface, e.g., by checking if it obeys fluid dynamics. Another class of important future work is to extend our potential flow framework to accommodate vorticity and viscosity, as well as to model turbulence-type flows.

Acknowledgement

The authors would like to thank Dr. H. Sakaino for his invaluable suggestions. This work was supported in part by NSF grant NSF-MSPA-MCS-0625931 and IIS-CAREER-0845268.

References

[1] E. Arnaud, E. Memin, R. Sosa, and G. Artana. A fluid motion estimator for schlieren image velocimetry. In *ECCV06*, pages I: 198–210, 2006. 1, 4, 8

[2] G. Aubert, R. Deriche, and P. Kornprobst. Computing optical flow via variational techniques. *SIAM J. Appl. Math.*, 60(1):156–182, 1999. 1

[3] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, and R. Szeliski. A database and evaluation methodology for optical flow. In *ICCV07*, pages 1–8, Oct. 2007. 1

[4] M. J. Black and P. Anandan. The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. *Comput. Vis. Image Underst.*, 63(1):75–104, 1996. 1, 5, 6, 7

[5] N. Cornelius and T. Kanade. Learning for optical flow using stochastic optimization. In *Proc. ACM SIGGRAPH/SIGART Interdisciplinary Workshop Motion: Representation and Perception*, Apr 1983. 1

[6] T. Corpetti, E. Mémin, and P. Pérez. Dense estimation of fluid flows. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(3):365–380, 2002. 1, 4, 8

[7] W. Craig and C. Sulem. Numerical simulation of gravity waves. *J. Comp. Phys.*, 108:73–83, 1993. 4

[8] E. De Castro and C. Morandi. Registration of translated and rotated images using finite fourier transforms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(5):700–703, 1987. 1

[9] M. A. Gennert and S. Negahdaripour. Relaxing the brightness constancy assumption in computing optical flow. Technical Report A.I. Memo No.975, MIT, 1987. 1

[10] G. D. Hager and P. N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(10):1025–1039, 1998. 1

[11] H. W. Haussecker and D. J. Fleet. Computing optical flow with physical models of brightness variation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(6):661–673, 2001. 1

[12] C. Hirsch. *Numerical Computation of Internal and External Flows, Volume 1*. Butterworth-Heinemann; 2 edition, 2007. 5

[13] K. Horikawa. *An introduction to ocean engineering*. Tokyo Univ. Press, 2004. 1

[14] V. Kwatra, A. Schödl, I. Essa, G. Turk, and A. Bobick. Graphcut textures: image and video synthesis using graph cuts. *ACM Trans. Graph.*, 22(3), 2003. 6

[15] N. Mukawa. Estimation of shape, reflection coefficients and illuminant direction from image sequences. In *ICCV90*, pages 507–512, 1990. 1

[16] H.-H. Nagel. On a constraint equation for the estimation of displacement rates in image sequences. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(1):13–30, 1989. 1

[17] Y. Nakajima, H. Inomata, H. Nogawa, Y. Sato, S. Tamura, K. Okazaki, and S. Torii. Physics-based flow estimation of fluids. *Pattern Recognition*, 36(5):1203 – 1212, 2003. 1

[18] Nvidia. Nvidia SDK 9.5. *http://developer.download.nvidia.com/SDK/9.5/Samples/samples.html*. 6

[19] N. Papenberg, A. Bruhn, T. Brox, S. Didas, and J. Weickert. Highly accurate optic flow computation with theoretically justified warping. *Int. J. Comput. Vision*, 67(2):141–158, 2006. 1

[20] S. Roth and M. J. Black. On the spatial statistics of optical flow. *Int. J. Comput. Vision*, 74(1):33–50, 2007. 1

[21] H. Sakaino. Fluid motion estimation method based on physical properties of waves. In *CVPR08*, June 2008. 2, 3, 4, 8

[22] J. Stam. Stable fluids. In *SIGGRAPH '99*, 1999. 6

[23] R. P. Wildes, M. J. Amabile, A.-M. Lanzillotto, and T.-S. Leu. Recovering estimates of fluid flow from image sequence data. *Comput. Vis. Image Underst.*, 80(2):246–266, 2000. 1

[24] L. Xu and P. Guyenne. Numerical simulation of three-dimensional nonlinear water waves. *J. Comp. Phys.*, 228:8446–8466, 2009. 2, 3, 4

[25] V. E. Zakharov. Stability of periodic waves of finite amplitude on the surface of a deep fluid. *J. Appl. Mech. Tech. Phys.*, 9:190–194, 1968. 4