Dense Optical Flow Algorithm

June 14, 2012 Wang Zhibin

Motivation

In the field of Computer Vision, motion detection and classification represents a central problem in the understanding of the environment. Motion within subsequent frames of an image sequence can be displayed as a dense vector field describing the movements of certain illumination patterns relative to the image plane, which corresponds to a projection of displacements performed by objects captured. Those vector fields are called *Optical Flow Fields*, or *Optical Flow*.

Optical Flow Constraint

Goal is the computation of the optic flow vector $\mathbf{w}(\mathbf{x}) := (u(\mathbf{x}), v(\mathbf{x}), h_t)$ between two successive images of an image sequence $\mathrm{I}: (\Omega \subset R^3) \to R$, where $\mathbf{x} = (x, y, t)$ is now a point in the spatio-temporal domain with t denoting the time axis. The grid size in temporal direction is set to be $h_t = 1$. In order to estimate this optical flow vector, it is necessary to define a constancy assumption. Assuming that it stays constant between the pixel (x, y) in the image at time t and the shifted pixel $(x+u, y+v)$ in a successive image at time $t+1$ yields the constraint $\mathrm{I(x, y, t)} = I(x+u, y+v, t+1)$.

This is a nonlinear equation in u and v. In order to resolve for u and v the equation is mostly linearized by a first order Taylor expansion. This leads to the linearized gray value constancy assumption, the so-called *optic flow constraint (OFC)*:

$$I_x u + I_y v + I_t = 0$$

The subscripts denote partial derivatives.

Variational Model

- **Protocol A: Original Horn and Schunck Model (OHS)**

Horn and Schunck combine the rate of change of image brightness and the measure of the departure from smoothness in the velocity flow. The energy functional is as follows:

$$E(u,v)=\int_{\Omega}\left(\underbrace{(I_x u+I_y v+I_t)^2}_{Data}+\underbrace{\alpha(|\nabla u|^2+|\nabla v|^2)}_{Smooth}\right)dxdy\ .$$

Minimize the energy functional with the Euler-Lagrange equations

$$(I_x u+I_y v+I_t)I_x-\alpha\Delta u=0$$

$$(I_x u+I_y v+I_t)I_y-\alpha\Delta v=0$$

The original Horn and Schunck use the Laplacian Approximate $\Delta u=\bar{u}-u$ and $\Delta v=\bar{v}-v$. Here is the way to compute the average. We use the follow template to convolve.

1/12	1/6	1/12
1/6	-1	1/6
1/12	1/6	1/12

Then, we use the Gauss-Seidel to solve the Euler-Lagrange Equations. We can compute a new set of velocity estimates (u^{n+1},v^{n+1}) from the estimated derivatives and the average of the previous velocity estimates (u^n,v^n) by

$$u^{n+1}=u^n-\frac{I_x\left(I_x\bar{u}^n+I_y\bar{v}^n+I_t\right)}{\alpha+I_x^2+I_y^2}$$

$$v^{n+1}=v^n-\frac{I_y\left(I_x\bar{u}^n+I_y\bar{v}^n+I_t\right)}{\alpha+I_x^2+I_y^2}$$

Protocol B: The Modification of Horn and Schunck Model (MHS)

The Laplacian of Approximate is the modification part. Substitute for divergence $\Delta u=u_{xx}+u_{yy}$ and $\Delta v=v_{xx}+v_{yy}$ the appropriate spatial discretization.

$$u_{xx}^{i,j}=u^{i+1,j}-u^{i,j}+u^{i-1,j}-u^{i,j}$$

$$u_{yy}^{i,j}=u^{i,j+1}-u^{i,j}+u^{i,j-1}-u^{i,j}$$

$$v_{xx}^{i,j}=v^{i+1,j}-v^{i,j}+v^{i-1,j}-v^{i,j}$$

$$u_{yy}^{i,j}=v^{i,j+1}-v^{i,j}+v^{i,j-1}-v^{i,j}$$

Where i denotes the row and j denotes the column.

So the Euler-Lagrange Equations at pixel i (column wise indexed)

$$(I_x^2)_i u_i+(I_xI_y)_i v_i+(I_xI_z)_i-\alpha\left(\sum_{j\in N_i}u_j-\sum_{j\in N_i}u_i\right)=0$$

$$(I_xI_y)_i u_i+(I_y^2)_i v_i+(I_yI_z)_i-\alpha\left(\sum_{j\in N_i}v_j-\sum_{j\in N_i}v_i\right)=0$$

Where $N_i=\{i+1,i-1,i+h,i-h\}$ denotes the four neighbors of pixel I, it may contain less than four elements.

The sparse linear system can be solved quite efficiently by successive over relaxation (SOR) method. Starting with an initialization$(u^0, v^0) = 0$, the SOR iteration scheme for minimizing the energy reads:

$$u_i^{k+1} = (1-w)u_i^k + w\frac{\sum_{j\in N^-(i)} u_j^{k+1} + \sum_{j\in N^+(i)} u_j^k - \frac{1}{\alpha}\left(\left(I_x I_y\right)_i v_i^k + (I_x I_z)_i\right)}{\sum_{j\in N^-(i)\cup j\in N^+(i)} 1 + \frac{1}{\alpha}(I_x^2)_i}$$

$$v_i^{k+1} = (1-w)v_i^k + w\frac{\sum_{j\in N^-(i)} v_j^{k+1} + \sum_{j\in N^+(i)} v_j^k - \frac{1}{\alpha}\left(\left(I_x I_y\right)_i v_i^{k+1} + \left(I_y I_z\right)_i\right)}{\sum_{j\in N^-(i)\cup j\in N^+(i)} 1 + \frac{1}{\alpha}(I_y^2)_i}$$

Here we can also use Symmetric Successive Over Relaxation (SSOR) and Red-Black SOR. SSOR combines a forward SOR sweep and a backward SOR sweep. Red-Black SOR combines odd pixel SOR sweep and even pixel SOR sweep.

- **Protocol C: Non-quadratic Horn and Schunck (NQ-HS)**

The quadratic penalize gives too much influence to outliers. To allow discontinuities in the optical flow field (object boundaries), we need to penalize them less severely. We choose a robust function like $\Psi(s^2) = \sqrt{(s^2)}$ which corresponds to TV regularization (minimizing the total variation of the field)

$$E(u,v) = \int_\Omega \left(\underbrace{\Psi(\,(I_x u + I_y v + I_t)^2)}_{Data} + \underbrace{\alpha\Psi(|\nabla u|^2 + |\nabla v|^2)}_{Smooth} \right) dxdy$$

Such a model contains the assumption of a piecewise smooth flow field. Choosing a convex and continuous Ψ would lead to a convex objective with unique optimum. So instead of $\Psi(s^2) = \sqrt{(s^2)}$, we choose $\Psi(s^2) = \sqrt{s^2 + \epsilon^2}$ with $\epsilon = 0.001$ which is continuous and convex. So the Euler-Lagrange equations for the NQ-HS functional are

$$\Psi'\left(\left(I_x u + I_y v + I_t\right)^2\right)\left(I_x u + I_y v + I_t\right)I_x - \alpha div(\Psi'(|\nabla u|^2 + |\nabla v|^2)\nabla u) = 0$$

$$\Psi'\left(\left(I_x u + I_y v + I_t\right)^2\right)\left(I_x u + I_y v + I_t\right)I_y - \alpha div(\Psi'(|\nabla u|^2 + |\nabla v|^2)\nabla v) = 0$$

We expand these equations.

$$(\Psi_D')_i((I_x^2)_i u_i + (I_x I_y)_i v_i + (I_x I_z)_i) - \alpha\left(\sum_{j\in N_i}(\Psi_S')_{i\sim j}u_j - \sum_{j\in N_i}(\Psi_S')_{i\sim j}u_i\right) = 0$$

$$(\Psi_D')_i((I_x I_y)_i u_i + (I_y^2)_i v_i + (I_y I_z)_i) - \alpha\left(\sum_{j\in N_i}(\Psi_S')_{i\sim j}v_j - \sum_{j\in N_i}(\Psi_S')_{i\sim j}v_i\right) = 0$$

Where

$$\Psi_D' = \Psi'((I_x u + I_y v + I_t)^2)$$

$$\Psi_S' = \Psi'(|\nabla u|^2 + |\nabla v|^2)$$

Note that when we compute Ψ'_S, we use two-point difference scheme suggested in Brox's thesis. For every pixel *i*, we compute the diffusivity of its four neighbors, which is

$$south: \quad (\Psi'_S)_{i\sim i+1} = \Psi'(\left|\nabla\left(u_{ii+\frac{1}{2},jj}\right)\right|^2 + \left|\nabla\left(v_{ii+\frac{1}{2},jj}\right)\right|^2)$$

$$north: \quad (\Psi'_S)_{i\sim i-1} = \Psi'(\left|\nabla\left(u_{ii-\frac{1}{2},jj}\right)\right|^2 + \left|\nabla\left(v_{ii-\frac{1}{2},jj}\right)\right|^2)$$

$$east: \quad (\Psi'_S)_{i\sim i+h} = \Psi'(\left|\nabla\left(u_{ii,jj+\frac{1}{2}}\right)\right|^2 + \left|\nabla\left(v_{ii,jj+\frac{1}{2}}\right)\right|^2)$$

$$west: \quad (\Psi'_S)_{i\sim i-h} = \Psi'(\left|\nabla\left(u_{ii,jj-\frac{1}{2}}\right)\right|^2 + \left|\nabla\left(v_{ii,jj+\frac{1}{2}}\right)\right|^2)$$

Then we use central differences

$$\nabla u_{ii+\frac{1}{2},jj} = \sqrt{\left(u_{ii+1,jj} - u_{ii,jj}\right)^2 + \left(\frac{1}{2}\left(\frac{u_{ii+1,jj+1} - u_{ii+1,jj-1}}{2} + \frac{u_{ii,jj+1} - u_{ii,jj-1}}{2}\right)\right)^2}$$

$$\nabla u_{ii-\frac{1}{2},jj} = \sqrt{(u_{ii,jj} - u_{ii-1,jj})^2 + (\frac{1}{2}(\frac{u_{ii-1,jj+1} - u_{ii-1,jj-1}}{2} + \frac{u_{ii,jj+1} - u_{ii,jj-1}}{2}))^2}$$

$$\nabla u_{ii,jj+\frac{1}{2}} = \sqrt{(u_{ii,jj+1} - u_{ii,jj})^2 + (\frac{1}{2}(\frac{u_{ii+1,jj+1} - u_{ii-1,jj+1}}{2} + \frac{u_{ii+1,jj} - u_{ii-1,jj}}{2}))^2}$$

$$\nabla u_{ii,jj-\frac{1}{2}} = \sqrt{(u_{ii,jj} - u_{ii,jj-1})^2 + (\frac{1}{2}(\frac{u_{ii+1,jj-1} - u_{ii-1,jj-1}}{2} + \frac{u_{ii+1,jj} - u_{ii-1,jj}}{2}))^2}$$

Where ii and jj are the row and column index of pixel, then h and w are the height and width of the image, while $\mathrm{N_i} = \{\mathrm{i}+1, \mathrm{i}-1, \mathrm{i}+\mathrm{h}, \mathrm{i}-\mathrm{h}\}$ denotes the four neighbor pixels at pixel i. Actually, the south and north, the east and west are equivalent respectively. So during the implementation, we only compute south and east neighbor, then shift them down and right respectively to get north and west.

Then let l denote the iteration index for the SOR iterations, then the iteration scheme for solving the linear system is:

$$u_i^{k,l+1} = (1-w)u_i^{k,l} + w\frac{\sum_{j\in N^-(i)}(\Psi'_S)_{i\sim j}^k u_j^{k,l+1} + \sum_{j\in N^+(i)}(\Psi'_S)_{i\sim j}^k u_j^{k,l} - \frac{(\Psi'_D)_i^k}{\alpha}((I_xI_y)_i v_i^{k,l} + (I_xI_z)_i)}{\sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^k + \frac{(\Psi'_D)_i^k}{\alpha}(I_x^2)_i}$$

$$v_i^{k,l+1} = (1-w)v_i^{k,l} + w\frac{\sum_{j\in N^-(i)}(\Psi'_S)_{i\sim j}^k v_j^{k,l+1} + \sum_{j\in N^+(i)}(\Psi'_S)_{i\sim j}^k v_j^{k,l} - \frac{(\Psi'_D)_i^k}{\alpha}((I_xI_y)_i u_i^{k,l+1} + (I_yI_z)_i)}{\sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^k + \frac{(\Psi'_D)_i^k}{\alpha}(I_y^2)_i}$$

Here we can also use Symmetric Successive Over Relaxation (SSOR) and Red-Black SOR. SSOR combines a forward SOR sweep and a backward SOR sweep. Red-Black SOR combines odd pixel SOR sweep and even pixel SOR sweep.

- **Protocol D: Combine Local and Global Method (CLG)**

It should combine the robustness of local methods with the density of global approaches. This should be done next. We start with spatial formulations. It becomes evident that the Lucas-Kanade method minimizes the quadratic form

$$E_{LK}(u,v) = \frac{1}{2} K_\rho * \left(I_x u + I_y v + I_t\right)^2.$$

While the Horn-Schunck technique minimize the functional

$$E_{HS}(u,v) = \int_\Omega \left(\left(I_x u + I_y v + I_t\right)^2 + \alpha(|\nabla u|^2 + |\nabla v|^2)\right) dxdy.$$

We simply combine LK and HS. Thus, we propose to minimize the functional

$$E_{CLG}(u,v) = \int_\Omega \left(K_\rho * \left(I_x u + I_y v + I_t\right)^2 + \alpha(|\nabla u|^2 + |\nabla v|^2)\right) dxdy\ .$$

We can get the Euler-Lagrange Equations:

$$K_\rho * (I_x u + I_y v + I_t) I_x - \alpha \Delta u = 0$$
$$K_\rho * (I_x u + I_y v + I_t) I_y - \alpha \Delta v = 0$$

Then we expand these equations.

$$K_\rho * (I_x^2)_i u_i + K_\rho * (I_x I_y)_i v_i + K_\rho * (I_x I_z)_i - \alpha\left(\sum_{j\in N_i} u_j - \sum_{j\in N_i} u_i\right) = 0$$

$$K_\rho * (I_x I_y)_i u_i + K_\rho * (I_y^2)_i v_i + K_\rho * (I_y I_z)_i - \alpha\left(\sum_{j\in N_i} v_j - \sum_{j\in N_i} v_i\right) = 0$$

We can solve these equations the same as the above. So we obtain the SOR iterations.

$$u_i^{k+1} = (1-w)u_i^k + w\frac{\sum_{j\in N^-(i)} u_j^{k+1} + \sum_{j\in N^+(i)} u_j^k - \frac{1}{\alpha}\left(\left(K_\rho * I_x I_y\right)_i v_i^k + \left(K_\rho * I_x I_z\right)_i\right)}{\sum_{j\in N^-(i)\cup j\in N^+(i)} 1 + \frac{1}{\alpha}(K_\rho * I_x^2)_i}$$

$$v_i^{k+1} = (1-w)v_i^k + w\frac{\sum_{j\in N^-(i)} v_j^{k+1} + \sum_{j\in N^+(i)} v_j^k - \frac{1}{\alpha}\left(\left(K_\rho * I_x I_y\right)_i v_i^{k+1} + \left(K_\rho * I_y I_z\right)_i\right)}{\sum_{j\in N^-(i)\cup j\in N^+(i)} 1 + \frac{1}{\alpha}(K_\rho * I_y^2)_i}$$

Here we can also use Symmetric Successive Over Relaxation (SSOR) and Red-Black SOR. SSOR combines a forward SOR sweep and a backward SOR sweep. Red-Black SOR combines odd pixel SOR sweep and even pixel SOR sweep.

Here the least square template is

0.0625	0.25	0.375	0.25	0.0625

So we can obtain a 5*5 matrix

0.0625*0.0625	0.0625*0.25	0.0625*0.375	0.0625*0.25	0.0625*0.0625
0.25*0.0625	0.25*0.25	0.25*0.375	0.25*0.25	0.25*0.0625
0.375*0.0625	0.375*0.25	0.375*0.375	0.375*0.25	0.375*0.0625
0.25*0.0625	0.25*0.25	0.25*0.375	0.25*0.25	0.25*0.0625
0.0625*0.0625	0.0625*0.25	0.0625*0.375	0.0625*0.25	0.0625*0.0625

- **Protocol E: Non Quadratic Combine Local and Global Method (NQ-CLG)**

According to the CLG, we can obtain the non-quadratic CLG. The energy functional is as follows.

$$E(u,v) = \int_\Omega \left(\underbrace{\Psi_D(K_\rho * (I_x u + I_y v + I_t)^2)}_{Data} + \underbrace{\alpha\Psi_S(|\nabla u|^2 + |\nabla v|^2)}_{Smooth}\right) dxdy$$

We can easily obtain the Euler-Lagrange Equations.

$$\Psi_D'\left(K_\rho * \left(I_x u + I_y v + I_t\right)^2\right)\left(K_\rho * \left(I_x u + I_y v + I_t\right)I_x\right) - \alpha div(\Psi_S'(|\nabla u|^2 + |\nabla v|^2)\nabla u) = 0$$

$$\Psi_D'\left(K_\rho * \left(I_x u + I_y v + I_t\right)^2\right)\left(K_\rho * \left(I_x u + I_y v + I_t\right)I_y\right) - \alpha div(\Psi_S'(|\nabla u|^2 + |\nabla v|^2)\nabla v) = 0$$

Then we expand these equations.

$$(\Psi_D')_i((K_\rho * I_x^2)_i u_i + (K_\rho * I_x I_y)_i v_i + (K_\rho * I_x I_z)_i) - \alpha\left(\sum_{j\in N_i}(\Psi_S')_{i\sim j}u_j - \sum_{j\in N_i}(\Psi_S')_{i\sim j}u_i\right) = 0$$

$$(\Psi_D')_i((K_\rho * I_x I_y)_i u_i + (K_\rho * I_y^2)_i v_i + (K_\rho * I_y I_z)_i) - \alpha\left(\sum_{j\in N_i}(\Psi_S')_{i\sim j}v_j - \sum_{j\in N_i}(\Psi_S')_{i\sim j}v_i\right) = 0$$

Where

$$\Psi_D'\left(K_\rho * \left(I_x u + I_y v + I_t\right)^2\right) = \frac{1}{\sqrt{1 + \frac{K_\rho * \left(I_x u + I_y v + I_t\right)^2}{\beta_D^2}}}$$

$$\Psi_S'(|\nabla u|^2 + |\nabla v|^2) = \frac{1}{\sqrt{1 + \frac{|\nabla u|^2 + |\nabla v|^2}{\beta_S^2}}}$$

Then let l denote the iteration index for the SOR iterations, then the iteration scheme for solving the linear system is:

$$u_i^{k,l+1} = (1-w)u_i^{k,l} + w\frac{\sum_{j\in N^-(i)}({\Psi'}_S)_{i\sim j}^k u_j^{k,l+1} + \sum_{j\in N^+(i)}({\Psi'}_S)_{i\sim j}^k u_j^{k,l} - \frac{({\Psi'}_D)_i^k}{\alpha}\left(\left(K_\rho * I_x I_y\right)_i v_i^{k,l} + \left(K_\rho * I_x I_z\right)_i\right)}{\sum_{j\in N^-(i)\cup j\in N^+(i)}({\Psi'}_S)_{i\sim j}^k + \frac{({\Psi'}_D)_i^k}{\alpha}(K_\rho * I_x^2)_i}$$

$$v_i^{k,l+1} = (1-w)v_i^{k,l} + w\frac{\sum_{j\in N^-(i)}({\Psi'}_S)_{i\sim j}^k v_j^{k,l+1} + \sum_{j\in N^+(i)}({\Psi'}_S)_{i\sim j}^k v_j^{k,l} - \frac{({\Psi'}_D)_i^k}{\alpha}\left(\left(K_\rho * I_x I_y\right)_i u_i^{k,l+1} + \left(K_\rho * I_y I_z\right)_i\right)}{\sum_{j\in N^-(i)\cup j\in N^+(i)}({\Psi'}_S)_{i\sim j}^k + \frac{({\Psi'}_D)_i^k}{\alpha}(K_\rho * I_y^2)_i}$$

Here we can also use Symmetric Successive Over Relaxation (SSOR) and Red-Black SOR. SSOR combines a forward SOR sweep and a backward SOR sweep. Red-Black SOR combines odd pixel SOR sweep and even pixel SOR sweep.

- **Protocol F: Nonlinear Non-quadratic Combine Local and Global Method (N-NQ-CLG)**

The quadratic penalize gives too much influence to outliers. To allow discontinuities in the optic flow field (object boundaries); we need to penalize them less severely. So the non-quadratic model reduces to

$$E(\mathbf{u}) = \int_\Omega \Psi(K_\rho * \left(\boldsymbol{I}(\mathbf{x}+\mathbf{u}) - \boldsymbol{I}(\mathbf{x})\right)^2)d\mathbf{x} + \alpha\int_\Omega \Psi(|\nabla\mathbf{u}|^2)d\mathbf{x}$$

The corresponding Euler-Lagrange equations that have to be satisfied by a function **u** minimizing the energy can be written as

$$\Psi'(I_z^2)\cdot K_\rho * I_z I_{x_1} - \alpha\mathrm{div}(\Psi'(|\nabla u_1|^2 + |\nabla u_2|^2)\nabla u_1) = 0$$

$$\Psi'(I_z^2)\cdot K_\rho * I_z I_{x_2} - \alpha\mathrm{div}(\Psi'(|\nabla u_1|^2 + |\nabla u_2|^2)\nabla u_2) = 0$$

Where

$$I_z = I(\mathbf{x}+\mathbf{u}) - I(\mathbf{x})$$
$$I_{x_1} = \frac{\partial\, I(\mathbf{x}+\mathbf{u})}{\partial x_1} \qquad\qquad I_{x_2} = \frac{\partial\, I(\mathbf{x}+\mathbf{u})}{\partial x_2}.$$

Note that I_z is not a temporal derivative, but a difference is sought to be minimized. One way to handle the nonlinear equations is to derive a fixed point scheme and to determine the solution iteratively. The actual iteration scheme as well as the proper choice of the initialization is equally important. Here is his Euler-Lagrange Equations.

$$\Psi'\left(K_\rho * I_z^2\right) \cdot K_\rho * I_z I_{x_1} - \alpha \mathrm{div}(\Psi'(|\nabla u_1|^2 + |\nabla u_2|^2)\nabla u_1) = 0$$

$$\Psi'\left(K_\rho * I_z^2\right) \cdot K_\rho * I_z I_{x_2} - \alpha \mathrm{div}(\Psi'(|\nabla u_1|^2 + |\nabla u_2|^2)\nabla u_2) = 0$$

Iteration Scheme

Let $\mathbf{u}^k = ({u_1}^k, {u_2}^k, 1)^T$ and let ${I_*}^k$ denote abbreviations defined in above but with the iteration variable $\mathbf{u}^k$ instead of **u,** then $\mathbf{u}^{k+1}$ can be obtained as the solution of

$$\Psi'(K_\rho * (I_z^2)^{k+1}) \cdot K_\rho * {I_z}^{k+1}{I_{x_1}}^k - \alpha \mathrm{div}(\Psi'(|\nabla {u_1}^{k+1}|^2 + |\nabla {u_2}^{k+1}|^2)\nabla {u_1}^{k+1}) = 0$$

$$\Psi'(K_\rho * (I_z^2)^{k+1}) \cdot K_\rho * {I_z}^{k+1}{I_{x_2}}^k - \alpha \mathrm{div}(\Psi'(|\nabla {u_1}^{k+1}|^2 + |\nabla {u_2}^{k+1}|^2)\nabla {u_2}^{k+1}) = 0$$

Where Ψ' denotes the derivative of Ψ with respect to its argument s^2. This fixed point iteration is nonlinear in **u** due to the structure of the components $I(\mathbf{x}+\mathbf{u}^{k+1})$ and $\Psi'(\cdot)$. Hence for further simplification towards linear equations, additional steps are necessary. First the terms of the form $I(\mathbf{x}+\mathbf{u}^{k+1})$ are linearized via Taylor expansion

$${I_z}^{k+1} = I(\mathbf{x}+\mathbf{u}^{k+1}) - I(\mathbf{x}) \approx I(\mathbf{x}+\mathbf{u}^k) + {I_{x_1}}^k d{u_1}^k + {I_{x_2}}^k d{u_2}^k - I(\mathbf{x}) =$$

$${I_{x_1}}^k d{u_1}^k + {I_{x_2}}^k d{u_2}^k + {I_z}^k.$$

We split the unknown iteration variable $\mathbf{u}^{k+1}$ into the known variable $\mathbf{u}^k$ and an unknown update $\boldsymbol{du}^k = (d{u_1}^k, d{u_2}^k, 0)^T$. This leads to a system

$$(\Psi')^k_{data} \cdot K_\rho * ({I_{x_1}}^k d{u_1}^k + {I_{x_2}}^k d{u_2}^k + {I_z}^k){I_{x_1}}^k - \alpha \mathrm{div}\left((\Psi')^k_{smooth}\nabla({u_1}^k + d{u_1}^k)\right) = 0$$

$$(\Psi')^k_{data} \cdot K_\rho * ({I_{x_1}}^k d{u_1}^k + {I_{x_2}}^k d{u_2}^k + {I_z}^k){I_{x_2}}^k - \alpha \mathrm{div}\left((\Psi')^k_{smooth}\nabla({u_2}^k + d{u_2}^k)\right) = 0$$

with the abbreviations

$$(\Psi')^k_{data} \coloneqq \Psi'(K_\rho * \left({I_{x_1}}^k d{u_1}^k + {I_{x_2}}^k d{u_2}^k + {I_z}^k\right)^2)$$
$$(\Psi')^k_{smooth} \coloneqq \Psi'(|\nabla({u_1}^k + d{u_1}^k)|^2 + |\nabla({u_2}^k + d{u_2}^k)|^2).$$

The flow $\mathbf{u}^k$ is known from the previous iteration step. Note that the system of the equations is still nonlinear with respect to $\boldsymbol{du}^k$**,** but now this nonlinearity stems only from the terms of the form Ψ'. We cope with this nonlinearity through an approach similar to the one above and we introduce a second, inner, fixed point iteration that will allow determining the increment $\boldsymbol{du}^k$ used in outer iteration.

To this end, let $\boldsymbol{du}^{k,0} = 0$ our initialization and let $\boldsymbol{du}^{k,l}$ denote the iteration variables at step *l*. Further, let $(\Psi')^{k,l}_{data}$ and $(\Psi')^{k,l}_{smooth}$ stand for the robustness factor and the diffusivity. Then $\boldsymbol{du}^{k,l}$ is the solution of the linear system

$$(\Psi')^{k,l}_{data} \cdot K_\rho * ({I_{x_1}}^k {du_1}^{k,l} + {I_{x_2}}^k {du_2}^{k,l} + {I_z}^k){I_{x_1}}^k - \alpha \mathrm{div}\big((\Psi')^{k}_{smooth} \nabla({u_1}^k + {du_1}^{k,l})\big) = 0$$

$$(\Psi')^{k,l}_{data} \cdot K_\rho * ({I_{x_1}}^k {du_1}^{k,l} + {I_{x_2}}^k {du_2}^{k,l} + {I_z}^k){I_{x_2}}^k - \alpha \mathrm{div}\big((\Psi')^{k}_{smooth} \nabla({u_2}^k + {du_2}^{k,l})\big) = 0$$

with the abbreviations

$$(\Psi')^{k,l}_{data} \coloneqq \Psi'\left(K_\rho * \left({I_{x_1}}^k {du_1}^{k,l} + {I_{x_2}}^k {du_2}^{k,l} + {I_z}^k\right)^2\right)$$

$$(\Psi')^{k}_{smooth} \coloneqq \Psi'(|\nabla({u_1}^k + {du_1}^{k,l})|^2 + |\nabla({u_2}^k + {du_2}^{k,l})|^2).$$

The fixed point of this system is employed to increment the outer iteration. After computation of the new outer iteration variable $\mathbf{u}^{k+1}$, we are able to obtain terms of $I(\mathbf{x} + \mathbf{u}^k)$ via interpolation. Such a calculation has to be performance only once just at the beginning of the inner iteration loop.

Relation to Warping

The energy $E(\mathbf{u})$ to be minimized is nonlinear and nonconvex. Hence the solution obtained by the proposed iteration scheme heavily depends on the initialization. In order to avoid getting trapped in local minima, we suggest employing a coarse-to-fine warping techniques.

For this purpose a complete image pyramid is generated where the image is successively down sampled by an arbitrary but fixed constant $\eta \in (0,1)$. For usual pyramids $\eta = 0.5$ is taken. Choosing larger η results in smoother transitions between levels of the pyramid and potentially leads to better results.

Expansion Equations at pixel *i* in every pyramid level

Here, we introduce a multi-resolution strategy to approximate a better global optimum. Gaussian pyramid with $\eta = 0.95$ is used. In every pyramid level *l*, the equations at pixel *i* are expanded as follows.

$$({\Psi'}_D)^{k,l}_i \cdot K_\rho * ({I_{x_1}}^k {I_{x_1}}^k)_i {du_i}^{k,l} + ({\Psi'}_D)^{k,l}_i \cdot K_\rho * ({I_{x_1}}^k {I_{x_2}}^k)_i {dv_i}^{k,l} + ({\Psi'}_D)^{k,l}_i \cdot K_\rho * \left({I_{x_1}}^k {I_z}^k\right)_i$$
$$- \alpha\Big(\sum_{j \in N_i} ({\Psi'}_S)^k_{i\sim j}\left({u_j}^k + {du_j}^{k,l}\right) - \sum_{j \in N_i} ({\Psi'}_S)^k_{i\sim j}({u_i}^k + {du_i}^{k,l})\Big) = 0$$

$$({\Psi'}_D)^{k,l}_i \cdot K_\rho * ({I_{x_1}}^k {I_{x_2}}^k)_i {du_i}^{k,l} + ({\Psi'}_D)^{k,l}_i \cdot K_\rho * ({I_{x_2}}^k {I_{x_2}}^k)_i {dv_i}^{k,l} + ({\Psi'}_D)^{k,l}_i \cdot K_\rho * \left({I_{x_1}}^k {I_z}^k\right)_i$$
$$- \alpha\Big(\sum_{j \in N_i} ({\Psi'}_S)^k_{i\sim j}\left({v_j}^k + {dv_j}^{k,l}\right) - \sum_{j \in N_i} ({\Psi'}_S)^k_{i\sim j}({v_i}^k + {dv_i}^{k,l})\Big) = 0$$

Where

$$({\Psi'}_D)^{k,l}_i \coloneqq \Psi'\left(K_\rho * \left({I_{x_1}}^k {du_i}^{k,l} + {I_{x_2}}^k {dv_i}^{k,l} + {I_z}^k\right)^2\right)$$

$${\Psi'}_S \coloneqq \Psi'(|\nabla({u_i}^k + {du_i}^{k,l})|^2 + |\nabla({v_i}^k + {du_i}^{k,l})|^2)$$

Note that when we compute ${\Psi'}_S$, we use two-point difference scheme suggested in Brox's thesis.

For every pixel *i*, we compute the diffusivity of its four neighbors, which is

$$south: \quad ({\Psi'}_S)_{i\sim i+1}^{k} = \Psi'(\left|\nabla\left({u_{ii+\frac{1}{2},jj}}^{k} + {du_{ii+\frac{1}{2},jj}}^{k,l}\right)\right|^2 + \left|\nabla\left({v_{ii+\frac{1}{2},jj}}^{k} + {dv_{ii+\frac{1}{2},jj}}^{k,l}\right)\right|^2)$$

$$north: \quad ({\Psi'}_S)_{i\sim i-1}^{k} = \Psi'(\left|\nabla\left({u_{ii-\frac{1}{2},jj}}^{k} + {du_{ii-\frac{1}{2},jj}}^{k,l}\right)\right|^2 + \left|\nabla\left({v_{ii-\frac{1}{2},jj}}^{k} + {dv_{ii-\frac{1}{2},jj}}^{k,l}\right)\right|^2)$$

$$east: \quad ({\Psi'}_S)_{i\sim i+h}^{k} = \Psi'(\left|\nabla\left({u_{ii,jj+\frac{1}{2}}}^{k} + {du_{ii,jj+\frac{1}{2}}}^{k,l}\right)\right|^2 + \left|\nabla\left({v_{ii,jj+\frac{1}{2}}}^{k} + {dv_{ii,jj+\frac{1}{2}}}^{k,l}\right)\right|^2)$$

$$west: \quad ({\Psi'}_S)_{i\sim i-h}^{k} = \Psi'(\left|\nabla\left({u_{ii,jj-\frac{1}{2}}}^{k} + {du_{ii,jj-\frac{1}{2}}}^{k,l}\right)\right|^2 + \left|\nabla\left({v_{ii,jj+\frac{1}{2}}}^{k} + {dv_{ii,jj-\frac{1}{2}}}^{k,l}\right)\right|^2)$$

Then we use central differences

$$\nabla u_{ii+\frac{1}{2},jj} = \sqrt{\left(u_{ii+1,jj} - u_{ii,jj}\right)^2 + \left(\frac{1}{2}\left(\frac{u_{ii+1,jj+1} - u_{ii+1,jj-1}}{2} + \frac{u_{ii,jj+1} - u_{ii,jj-1}}{2}\right)\right)^2}$$

$$\nabla u_{ii-\frac{1}{2},jj} = \sqrt{(u_{ii,jj} - u_{ii-1,jj})^2 + (\frac{1}{2}(\frac{u_{ii-1,jj+1} - u_{ii-1,jj-1}}{2} + \frac{u_{ii,jj+1} - u_{ii,jj-1}}{2}))^2}$$

$$\nabla u_{ii,jj+\frac{1}{2}} = \sqrt{(u_{ii,jj+1} - u_{ii,jj})^2 + (\frac{1}{2}(\frac{u_{ii+1,jj+1} - u_{ii-1,jj+1}}{2} + \frac{u_{ii+1,jj} - u_{ii-1,jj}}{2}))^2}$$

$$\nabla u_{ii,jj-\frac{1}{2}} = \sqrt{(u_{ii,jj} - u_{ii,jj-1})^2 + (\frac{1}{2}(\frac{u_{ii+1,jj-1} - u_{ii-1,jj-1}}{2} + \frac{u_{ii+1,jj} - u_{ii-1,jj}}{2}))^2}$$

Where *ii* and *jj* are the row and column index of pixel, then *h* and *w* are the height and width of the image, while $N_i = \{i+1, i-1, i+h, i-h\}$ denotes the four neighbor pixels at pixel *i.* Actually, equation south and north, equation east and west are equivalent respectively. So during the implementation, we only compute south and east neighbor, then shift them down and right respectively to get north and west.

Between every pyramid level, *u* and *v* are updated as

$$u^{k+1} = u^k + du^k$$
$$v^{k+1} = v^k + dv^k$$

They stay the same during the fix point iteration inside each pyramid level. Between levels, bilinear interpolation is used to change the scale of *u* and *v.*

- **Protocol G: Non-quadratic Combine Brightness and Gradient Method (NQ-CBG)**

In the case the brightness does not remain constant during motion (for example cloud motion) we add in the data term gradient constancy assumption:

$$\nabla I(x+u, y+v, z+1) - \nabla I(x,y,z) = 0$$

After Taylor expansion we get the constraints:

$$I_{xx}u + I_{xy}v + I_{xz} = 0$$
$$I_{xy}u + I_{yy}v + I_{yz} = 0$$

Which involve second order derivatives and we expect higher sensitivity to noise. So the data term becomes:

$$E_{data} = \left(I_x u + I_y v + I_t\right)^2 + \gamma(\left(I_{xx}u + I_{xy}v + I_{xt}\right)^2 + \left(I_{xy}u + I_{yy}v + I_{yt}\right)^2)$$

And we get the energy functional:

$$E_{CBG}(u,v) = \int_\Omega \left(\Psi(E_{data}) + \alpha\Psi(|\nabla u|^2 + |\nabla v|^2)\right) dxdy$$

We can easily obtain the Euler-Lagrange Equations.

$$\Psi'(E_{data})\left(\left(I_x u + I_y v + I_t\right)I_x + \gamma\left(\left(I_{xx}u + I_{xy}v + I_{xt}\right)I_{xx} + \left(I_{xy}u + I_{yy}v + I_{yt}\right)I_{xy}\right)\right) - \alpha\, \mathrm{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2)\nabla u) = 0$$

$$\Psi'(E_{data})\left(\left(I_x u + I_y v + I_t\right)I_y + \gamma\left(\left(I_{xx}u + I_{xy}v + I_{xt}\right)I_{xy} + \left(I_{xy}u + I_{yy}v + I_{yt}\right)I_{yy}\right)\right) - \alpha\, \mathrm{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2)\nabla v) = 0$$

Then we expand these equations.

$$(\Psi'_D)_i((I_x^2)_i u_i + (I_x I_y)_i v_i + (I_x I_t)_i + \gamma\left((I_{xx}^2 + I_{xy}^2)_i u_i + (I_{xx}I_{xy} + I_{yy}I_{xy})_i v_i + (I_{xx}I_{xt} + I_{xy}I_{yt})_i\right)) - \alpha\left(\sum_{j\in N_i}(\Psi'_S)_{i\sim j}u_j - \sum_{j\in N_i}(\Psi'_S)_{i\sim j}u_i\right) = 0$$

$$(\Psi'_D)_i((I_x I_y)_i u_i + (I_y^2)_i v_i + (I_y I_z)_i + \gamma\left((I_{xx}I_{xy} + I_{yy}I_{xy})_i u_i + (I_{xy}^2 + I_{yy}^2)_i v_i + (I_{xy}I_{xt} + I_{yy}I_{yt})_i\right)) - \alpha\left(\sum_{j\in N_i}(\Psi'_S)_{i\sim j}u_j - \sum_{j\in N_i}(\Psi'_S)_{i\sim j}u_i\right) = 0$$

And the update equations with SOR are:

$$u_i^{k,l+1} = (1-w)u_i^{k,l} + w\frac{\sum_{j\in N^-(i)}(\Psi'_S)_{i\sim j}^k u_j^{k,l+1} + \sum_{j\in N^+(i)}(\Psi'_S)_{i\sim j}^k u_j^{k,l}}{\sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^k + \frac{(\Psi'_D)_i^k}{\alpha}((I_x^2)_i + (I_{xy})_i^2 + (I_{xx})_i^2)}$$

$$-\frac{\frac{(\Psi'_D)_i^k}{\alpha}((I_x I_y)_i v_i^{k,l} + (I_x I_z)_i + \gamma(\left((I_{xy})_i v_i^{k,l} + (I_{xt})_i\right)(I_{xx})_i + \left((I_{yy})_i v_i^{k,l} + (I_{yt})_i\right)(I_{xy})_i)}{\sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^k + \frac{(\Psi'_D)_i^k}{\alpha}((I_x^2)_i + (I_{xy})_i^2 + (I_{xx})_i^2)}$$

$$v_i^{k,l+1} = (1-w)v_i^{k,l} + w\frac{\sum_{j\in N^-(i)}(\Psi'_S)_{i\sim j}^k v_j^{k,l+1} + \sum_{j\in N^+(i)}(\Psi'_S)_{i\sim j}^k v_j^{k,l}}{\sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^k + \frac{(\Psi'_D)_i^k}{\alpha}((I_y^2)_i + (I_{xy})_i^2 + (I_{yy})_i^2)}$$

$$-\frac{\frac{(\Psi'_D)_i^k}{\alpha}((I_x I_y)_i u_i^{k,l} + (I_y I_z)_i + \gamma(((I_{xx})_i v_i^{k,l} + (I_{xt})_i)(I_{xy})_i + \left((I_{xy})_i u_i^{k,l+1} + (I_{yt})_i\right)(I_{yy})_i)}{\sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^k + \frac{(\Psi'_D)_i^k}{\alpha}((I_y^2)_i + (I_{xy})_i^2 + (I_{yy})_i^2)}$$

- **Protocol H: Nonlinear Non-quadratic Combine Brightness and Gradient Method (N-NQ-CBG)**

We choose not to use Taylor expansion and to have a nonlinear but accurate function in the data tem:

$$\mathrm{E(u,v)} = \int_\Omega (\Psi(|I(x+w) - I(x)|^2 + \gamma|\nabla I(x+w) - \nabla I(x)|^2) + \alpha\Psi(|\nabla u|^2 + |\nabla v|^2))dx$$

The nonlinear data term has as a consequence the above functional to have multiple local optima. Each local optimum has to satisfy the Euler-Lagrange equations:

$$\Psi'\left(I_t^2 + \gamma\left(I_{xt}^2 + I_{yt}^2\right)\right)\left(I_x I_z + \gamma\left(I_{xx}I_{xz} + I_{xy}I_{yz}\right)\right) - \alpha\,\mathrm{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2)\nabla u) = 0$$

$$\Psi'\left(I_t^2 + \gamma\left(I_{xt}^2 + I_{yt}^2\right)\right)\left(I_x I_z + \gamma\left(I_{xx}I_{xz} + I_{xy}I_{yz}\right)\right) - \alpha\,\mathrm{div}(\Psi'(|\nabla u|^2 + |\nabla v|^2)\nabla u) = 0$$

Where

$$I_x = \frac{\partial I(x+w)}{\partial x}$$
$$I_y = \frac{\partial I(x+w)}{\partial y}$$
$$I_t = I(x+w) - I(x)$$
$$I_{xx} = \frac{\partial^2 I(x+w)}{\partial x^2}$$
$$I_{yy} = \frac{\partial^2 I(x+w)}{\partial y^2}$$
$$I_{xy} = \frac{\partial^2 I(x+w)}{\partial x \partial y}$$
$$I_{xt} = \frac{\partial I(x+w)}{\partial x} - \frac{\partial I(x)}{\partial x}$$
$$I_{yt} = \frac{\partial I(x+w)}{\partial y} - \frac{\partial I(x)}{\partial y}$$

Here, we introduce a multi-resolution strategy to approximate a better global optimum. Gaussian Pyramid with $\eta = 0.95$ is used. In every pyramid level l, the equations at pixel i are expanded as follows. We omit the superscript l here.

$$(\Psi_D')_i((I_x^2)_i du_i + (I_x I_y)_i dv_i + (I_x I_t)_i + \gamma\big((I_{xx}^2 + I_{xy}^2)_i du_i + (I_{xx} I_{xy} + I_{yy} I_{xy})_i dv_i + (I_{xx} I_{xt} + I_{xy} I_{yt})_i\ )\big) - \alpha\left(\sum_{j \in N_i} (\Psi_S')_{i\sim j}(u_j + du_j) - \sum_{j \in N_i} (\Psi_S')_{i\sim j}(u_i + du_i)\right) = 0$$

$$(\Psi_D')_i((I_x I_y)_i \mathrm{d}u_i + (I_y^2)_i dv_i + (I_y I_z)_i + \gamma\big((I_{xx} I_{xy} + I_{yy} I_{xy})_i du_i + (I_{xy}^2 + I_{yy}^2)_i dv_i + (I_{xy} I_{xt} + I_{yy} I_{yt})_i\ )\big) - \alpha\left(\sum_{j \in N_i} (\Psi_S')_{i\sim j}(u_j + du_j) - \sum_{j \in N_i} (\Psi_S')_{i\sim j}(u_i + du_i)\right) = 0$$

Where

$$\Psi_D' = \Psi'((I_x du + I_y dv + I_t\ )^2 + \gamma((I_{xx} du + I_{xy} dv + I_{xt}\ )^2 + (I_{xy} du + I_{yy} dv + I_{yt}\ )^2))$$
$$\Psi_S' = \Psi'(|\nabla(u + du)|^2 + |\nabla(v + dv)|^2)$$

Between every pyramid level, u, v are updated as

$$u^l = u^{l-1} + du^{l-1}$$
$$v^l = v^{l-1} + dv^{l-1}$$

These stay the same during the fix point iteration inside each pyramid level. Between levels, bilinear interpolation is used to change the scale of u, v.

Discretization yields a linear system of equations, which can be solved by SOR. Let m denote the iteration index for the SOR iterations, then the iteration scheme for solving the linear system is:

$$du_i^{k,l,m+1} = (1-w)du_i^{k,l,m} + w\frac{\sum_{j\in N^-(i)}(\Psi'_S)_{i\sim j}^{k,l}(u_j^k + du_j^{k,l,m+1}) + \sum_{j\in N^+(i)}(\Psi'_S)_{i\sim j}^{k,l}(u_j^k + du_j^{k,l,m}) - \sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^{k,l} u_j^k}{\sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^{k,l} + \frac{(\Psi'_D)_i^{k,l}}{\alpha}(((I_x)_i^k)^2 + \left((I_{xy})_i^k\right)^2 + \left((I_{xx})_i^k\right)^2)}$$
$$- \frac{\frac{(\Psi'_D)_i^{k,l}}{\alpha}\left(\left(I_x I_y\right)_i^k dv_i^{k,l,m} + (I_x I_z)_i^k + \gamma\left(\left(\left(I_{xy}\right)_i^k dv_i^{k,l,m} + (I_{xt})_i^k\right)(I_{xx})_i^k + \left(\left(I_{yy}\right)_i^k dv_i^{k,l,m} + \left(I_{yt}\right)_i^k\right)\left(I_{xy}\right)_i^k\right)\right)}{\sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^{k,l} + \frac{(\Psi'_D)_i^{k,l}}{\alpha}(((I_x)_i^k)^2 + \left((I_{xy})_i^k\right)^2 + \left((I_{xx})_i^k\right)^2)}$$

$$
\begin{aligned}
& dv_i^{k,l,m+1} \\
&= (1-w)dv_i^{k,l,m} \\
&+ w\frac{\sum_{j\in N^-(i)}(\Psi'_S)_{i\sim j}^{k,l}(v_j^k + dv_j^{k,l,m+1}) + \sum_{j\in N^+(i)}(\Psi'_S)_{i\sim j}^{k,l}(v_j^k + dv_j^{k,l,m}) - \sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^{k,l}v_j^k}{\sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^{k,l} + \frac{(\Psi'_D)_i^{k,l}}{\alpha}(((I_y)_i^k)^2 + \left((I_{xy})_i^k\right)^2 + \left((I_{yy})_i^k\right)^2)} \\
&- \frac{\frac{(\Psi'_D)_i^{k,l}}{\alpha}(\left(I_xI_y\right)_i^k du_i^{k,l,m+1} + \left(I_yI_z\right)_i^k + \gamma(((I_{xx})_i^k du_i^{k,l,m+1} + (I_{xt})_i^k)\left(I_{xy}\right)_i^k + \left(\left(I_{xy}\right)_i^k du_i^{k,l,m+1} + \left(I_{yt}\right)_i^k\right)\left(I_{yy}\right)_i^k)}{\sum_{j\in N^-(i)\cup j\in N^+(i)}(\Psi'_S)_{i\sim j}^{k,l} + \frac{(\Psi'_D)_i^{k,l}}{\alpha}(((I_y)_i^k)^2 + \left((I_{xy})_i^k\right)^2 + \left((I_{yy})_i^k\right)^2)}
\end{aligned}
$$

Here we can also use Symmetric Successive Over Relaxation (SSOR) and Red-Black SOR. SSOR combines a forward SOR sweep and a backward SOR sweep. Red-Black SOR combines odd pixel SOR sweep and even pixel SOR sweep.

Summary

The above dense optical flow algorithm is based on Variational model. Firstly, we choose some suitable assumptions according to the difference motions, and we can obtain Variational Energy Functional. Secondly, we can get Euler-Lagrange Equations in Euler-Lagrange method. Thirdly, we solve these equations by some ways, such as Gauss-Seidel, Successive Over Relaxation and Symmetric Successive Over Relaxation. At last, we can obtain the optical flow.

Reference

1. Horn, B. and Schunck, B. 1981. Determining optical flow. Artificial Intelligence, 17:185–203.
2. Thomas Brox. 2005. From Pixels to Regions: Partial Differential Equations in Image Analysis. PhD Thesis.
3. Andres Bruhn. 2006. Variational Optic Flow Computation Accurate Modelling and Efficient Numerics. PhD Thesis.
4. Barron, J.L., Fleet, D.J., and Beauchemin, S.S. 1994. Performance of optical flow techniques. International Journal of Computer Vision,12(1):43–77.
5. A. Bruhn, J. Weickert, and C. Schn¨orr. Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods. International Journal of Computer Vision, 61(3):211–231, 2005.
6. T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, “High Accuracy Optical Flow Estimation Based on a Theory for Warping,” Proc.Eighth European Conf. Computer Vision, T. Pajdla and J. Matas, eds.,pp. 25-36, May 2004.

Tips

The document is about dense optical flow algorithm part and written by WangZhibin on June 19, 2012. If you have some questions or find some errors, please contact me. My E-mail address is zhibinwang@yahoo.cn or stepzhibin@gmail.com.