1-1-2010

Motion estimation with adaptive regularization and neighborhood dependent constraint

Muhammad Wasim Nawaz
University of Wollongong, mwn831@uow.edu.au

Abdesselam Bouzerdoum
University of Wollongong, bouzer@uow.edu.au

Son Lam Phung
University of Wollongong, phung@uow.edu.au

Recommended Citation

Nawaz, Muhammad Wasim; Bouzerdoum, Abdesselam; and Phung, Son Lam: Motion estimation with adaptive regularization and neighborhood dependent constraint 2010, 387-392.
https://ro.uow.edu.au/infopapers/827

Motion estimation with adaptive regularization and neighborhood dependent constraint

Abstract

Modern variational motion estimation techniques use total variation regularization along with the L1 norm in constant brightness data term. An algorithm based on such homogeneous regularization is unable to preserve sharp edges and leads to increased estimation errors. A better solution is to modify regularizer along strong intensity variations and occluded areas. In addition, using neighborhood information with data constraint can better identify correspondence between image pairs than using only a pointwise data constraint. In this work, we present a novel motion estimation method that uses neighborhood dependent data constraint to better characterize local image structure. The method also uses structure adaptive regularization to handle occlusions. The proposed algorithm has been evaluated on Middlebury's benchmark image sequence dataset and compared to state-of-the-art algorithms. Experiments show that proposed method can give better performance under noisy conditions.

Keywords

dependent, neighborhood, constraint, regularization, motion, adaptive, estimation

Disciplines

Physical Sciences and Mathematics

Publication Details

Nawaz, M. Wasim., Bouzerdoum, A. & Phung, S. (2010). Motion estimation with adaptive regularization and neighborhood dependent constraint. 2010 Digital Image Computing: Techniques and Applications (DICTA 2010) (pp. 387-392). USA: IEEE.

2010 Digital Image Computing: Techniques and Applications

Motion Estimation with Adaptive Regularization and Neighborhood Dependent Constraint

Muhammad Wasim Nawaz, Abdesselam Bouzerdoum, Son Lam Phung
ICT Research Institute, University of Wollongong
Northfields Avenue, NSW 2522, Australia
mwn831,bouzer,phung@uow.edu.au

Abstract

Modern variational motion estimation techniques use total variation regularization along with the $\ell 1$ norm in constant brightness data term. An algorithm based on such homogeneous regularization is unable to preserve sharp edges and leads to increased estimation errors. A better solution is to modify regularizer along strong intensity variations and occluded areas. In addition, using neighborhood information with data constraint can better identify correspondence between image pairs than using only a pointwise data constraint. In this work, we present a novel motion estimation method that uses neighborhood dependent data constraint to better characterize local image structure. The method also uses structure adaptive regularization to handle occlusions. The proposed algorithm has been evaluated on Middlebury's benchmark image sequence dataset and compared to state-of-the-art algorithms. Experiments show that proposed method can give better performance under noisy conditions.

1. Introduction

Motion estimation is one of the basic tasks in computer vision, which can further be used for more complex vision tasks such as object recognition and scene analysis in image sequences. The two-dimensional motion field, which is the projection of the three-dimensional motion onto the image plane of a camera or any other imaging device, is known as *optical flow* [6]. Methods to estimate optical flow can be categorized into four categories: variational (gradient based) [13, 14, 15, 8, 7, 22, 21], energy based [1, 12], phase based [10] and region matching based [2]. Current research in the field focuses on gradient based methods due to their high accuracy [5].

Horn and Schunck [13] are the first to formulate optical flow estimation as a variational problem and minimize data and regularization energy of the form

$$E(v) = \lambda E_D(v) + E_R(v), \tag{1}$$

this formulation uses well-known brightness constancy assumption as data term E_D (weighted by λ), and a smoothness prior E_R is added to regularize the motion field. The purpose of the regularizer term is to impose smoothness upon the motion field. The regularizer also updates the regions that has insufficient gradient, which yields a dense flow field. Horn and Schunck's variational model uses quadratic penalization of both data and smoothness constraints. Since then, many improvements have been made in optical flow estimation. Some improved algorithms replace quadratic regularization with discontinuity preserving regularization [15, 16, 19], others penalize data term using the $\ell 1$ norm [22, 20, 21].

Traditional optical flow estimation methods use a linearized version of brightness constancy constraint. If the pixel displacement from one frame to another is large, linearized model can not perform well. Large displacements can be estimated by constructing Laplacian pyramid of images and using coarse-to-fine warping techniques [2, 6].

The data term in variational model only enforces pointwise similarity of intensity across an image pair; it calculates a normal flow field if a regularizer is not used [11]. Although a regularization term helps identify correspondence between image pairs, more sophisticated data terms can improve the performance of underlying variational methods [18]. If neighborhood information of pointwise data term is integrated, local image structure can be better characterized.

In this paper, we present optical flow estimation method that uses a neighborhood dependent constraint and structure adaptive regularization. We show that using intensity information from neighboring pixels makes this method more robust against noise and outliers. The remainder of the paper is organized as follows. We summarize neighborhood dependent optical flow estimation model in Section 2. Advantages of using structure adaptive regularization are discussed in Section 3. Experimental results, performance

978-0-7695-4271-3/10 $26.00 © 2010 IEEE
DOI 10.1109/DICTA.2010.72

evaluation and comparison of the proposed method are presented in Section 4. The paper ends with concluding remarks in Section 5.

2. Optical Flow Estimation with Neighborhood Dependent Constraint

In most optical flow estimation techniques, it is assumed that change in image intensity is only due to discrete displacements of pixels from one frame to another. This assumption is also known as brightness constancy; it is used as data term in most state-of-the-art optical flow estimation techniques. The brightness constancy assumption is given by

$$E_D(v) = \rho(|I_1(\vec{s}) - I_2(\vec{s} + \vec{v})|), \tag{2}$$

where $\vec{v} = [v_1 \; v_2]$ is the combination of horizontal and vertical velocity components and $\vec{s} = [x \; y]$ is spatial variable. For simplicity, we omit temporal variable t. Optical flow estimation algorithms may use either quadratic data term $\rho(z) = z^2$ or more robust data term of the form $\rho(z) = |z|$.

Because pointwise constraint gives no information about image structure, it can only calculate motion component normal to some image structure or edge (parallel to the intensity gradient)- known as *aperture problem*. To estimate optical flow, a regularizer such as smoothness prior is used. However if local image structure is better characterized, then using a regularizer can better estimate the optical flow. Integrating information over a neighborhood can determine the image structure

$$E_D(v) = \sum_{\vec{s} \in R} \mathbf{W}(\rho(|I_1(\vec{s}) - I_2(\vec{s} + \vec{v})|), \tag{3}$$

where $\mathbf{W}$ is a window or region of interest with appropriate weights which shows explicit dependency upon neighborhood. Note that an appropriate neighborhood size is first selected and then $\mathbf{W}$ is computed for every point in data term. We can also consider data term as $\ell 1$ distance of brightness constancy constraint, computed over a weighted window. To simplify the minimization stage and derive Euler-Lagrange equations, we assume herein $\mathbf{W}$ to be a linear operator.

Interestingly, if a linearized version of quadratic data term $|I_1(\vec{s}, t) - I_2(\vec{s} + \vec{v}, t + 1)|^2 \approx \nabla I(\vec{s}, t).\vec{v}$ is used, along with the average taken over neighborhood $\mathbf{W}$, we get Lucas and Kanade's local model [14], which can be solved by applying least square estimation. Alternatively we can also use Gaussian or even more sophisticated weights. In case of a Gaussian kernel, the algorithm's robustness under noisy conditions is improved. Bruhn et al. attempted to use Gaussian weights with Horn and Schuncks's variational model in a combined local and global (CLG) mothed [8]. They combine advantages from both local and global optical flow estimation methods. Although their model works well in noisy situations, it applies quadratic penalty terms for both data and regularizer terms.

3. Structure Adaptive Regularization

Recently, it has become clear that using a regularizer involving $\ell 1$ norm is very suitable for image processing tasks. This type of regularizer can preserve sharp edges, a task which the $\ell 2$ norm cannot achieve. Preserving sharp edges is desirable in optical flow estimation. Therefore it is better to use total variation as smoothness prior in neighborhood dependent optical flow estimation model

$$E(v) = \int_R \lambda \mathbf{W} \rho(|I_1(\vec{s}) - I_2(\vec{s} + \vec{v})|) + |\nabla \vec{v}| d\vec{s}, \tag{4}$$

where $|\nabla \vec{v}| = |\nabla v_1| + |\nabla v_2|$ is the total variation norm. Although total variation norm gives better results compared to quadratic regularizer, it is homogeneous in the sense that it imposes the same amount of regularization everywhere, resulting in unnecessary smoothing of edges and strong intensity areas.

Discontinuities in the flow field usually occur across occlusions and strong intensity edges; therefore making regularizer adaptive to image intensity can be favorable in preserving discontinuities. Structure and brightness adaptive regularizer has been used by [15] which uses anisotropic and inhomogeneous regularization. Here, we add a relatively simple function with total variation regularization mentioned in [19]. This function helps to control the amount of regularization across occluded areas

$$E(v) = \int_R \lambda \mathbf{W} \rho(|I_1(\vec{s}) - I_2(\vec{s} + \vec{v})|) + e^{(-\gamma |\nabla I|^{\beta})} |\nabla \vec{v}| d\vec{s}, \tag{5}$$

with γ and β are positive constants. The term $F = e^{(-\gamma |\nabla I|^{\beta})}$ is a decreasing function that gives regularizer appropriate weights according to edge strengths. Note that F has an inverse relation with edge intensity. As a consequence, the regularizer effect is decreased along strong image edges preserving sharp edges. In low textured areas, flow field are propagated by strong effect of regularizer. We name the model given in Equation 5 as neighborhood dependent constraint with adaptive total variation regularization (Ad-TV-NDC).

4. Minimization of Energy Functional

Minimization of energy involving $\ell 1$ and total variation norm is not a straightforward task because these norms are non-differentiable at zero. In this paper, we apply the algorithm presented in [9], which uses dual formulation of Rudin, Osher and Fatemi's total variation model [17], and minimizes it by projected gradient descent method. First, we modify our model given in Equation (5) and introduce a coupling variable $\vec{w} = [w_1 \; w_2]$ [3] as follows

$$\int_R \sum_{i=1}^{2} \left[\lambda \mathbf{W} \rho(|I_1(\vec{s}) - I_2(\vec{s}+\vec{w})|) + \frac{(v_i - w_i)^2}{2\alpha} + F|\nabla \vec{v}| \right] d\vec{s}, \tag{6}$$

where the sum over i represents minimization problem in more than one dimensions. Note that $i = 2$ for two-dimensional optical flow estimation. In the above setting, α is a small constant; thus, the estimates v_i and w_i are almost equal. Now we solve for v_i and w_i separately:

$$\int_R \sum_{i=1}^{2} \frac{(v_i - w_i)^2}{2\alpha} + F|\nabla \vec{v}| \, d\vec{s}, \tag{7}$$

and

$$\int_R \sum_{i=1}^{2} \lambda \mathbf{W} \rho(|I_1(\vec{s}) - I_2(\vec{s} + \vec{w})|) + \frac{(v_i - w_i)^2}{2\alpha} \, d\vec{s}. \tag{8}$$

To solve (7) we apply the algorithm given in [9]

$$\sum_{i=1}^{2} v_i = w_i + \alpha \mathbf{div}(F \mathbf{p}_i) \tag{9}$$

where $\mathbf{p_i} = [p_{i,1} \; p_{i,2}]$ is a dual variable. It is introduced for every dimension i, and each $\mathbf{p_i}$ follows $\nabla(\alpha \mathbf{div} \mathbf{p_i} - w_i) = |\nabla(\alpha \mathbf{div} \mathbf{p_i} - w_i)|\mathbf{p_i}$. The divergence operator is given as

$$\mathbf{div}(\mathbf{p_i}) = \nabla . \mathbf{p_i} = \frac{\partial p_{i,1}}{\partial x} + \frac{\partial p_{i,2}}{\partial y}. \tag{10}$$

The solution to $\mathbf{p_i}$ can be given by following gradient descent scheme [9]:

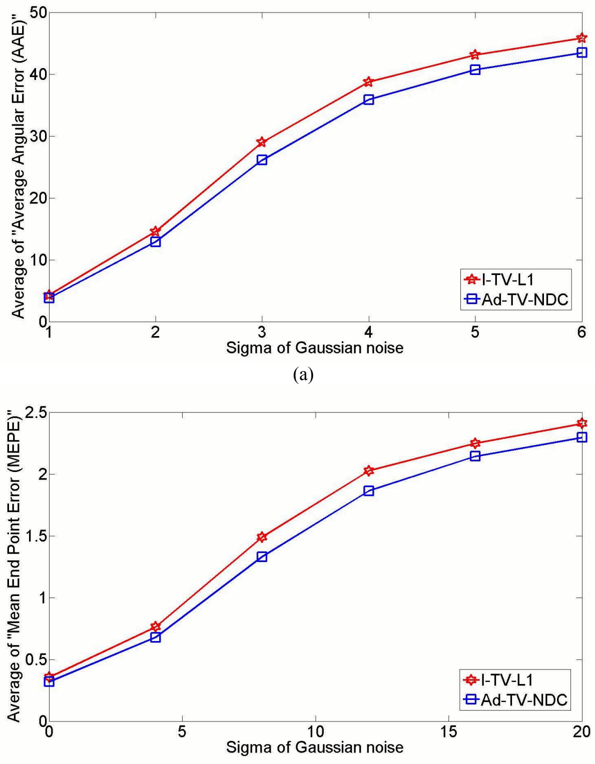
$$\sum_{i=1}^{2} (p_{i,1})^{n+1} = \sum_{i=1}^{2} \frac{(p_{i,1})^n + \frac{\tau}{\alpha}(F \nabla^x w_i)}{\max\{1, \frac{\tau}{\alpha}(F|\nabla^x w_i|)\}} \tag{11}$$

$$\sum_{i=1}^{2} (p_{i,2})^{n+1} = \sum_{i=1}^{2} \frac{(p_{i,2})^n + \frac{\tau}{\alpha}(F \nabla^y w_i)}{\max\{1, \frac{\tau}{\alpha}(F|\nabla^y w_i|)\}} \tag{12}$$

The above dual variable based update scheme converges for step size τ less than 0.25 [9]. To minimize w_i in the second part of the model in (8), we derive associated Euler-Lagrange equations with appropriate thresholding scheme

$$\vec{w}^{n+1} = \vec{v}^{n+1} + \begin{cases} -\lambda\theta \mathbf{W} \nabla I_1 & \rho(\vec{v}) > -\lambda\theta \mathbf{W}|\nabla I_1|^2 \\ \lambda\theta \mathbf{W} \nabla I_1 & \rho(\vec{v}) < -\lambda\theta \mathbf{W}|\nabla I_1|^2 . \\ -\rho(\vec{v}) \frac{\mathbf{W}\nabla I_1}{\mathbf{W}|\nabla I_1|^2} & |\rho(\vec{v})| \leq -\lambda\theta \mathbf{W}|\nabla I_1|^2 \end{cases} \tag{13}$$

Upon inspection of Equations (9) and (13), it becomes clear that this repetitive scheme alternatively updates $\vec{v}$ or $\vec{w}$ in every iteration.



(a)

(b)

Figure 1. (a) Average AAE and (b) Average MEPE over all video sequences plotted against sigma of Gaussian noise. Note that Ad-TV-NDC outperform improved-TV-L1 under noisy conditions.

5. Experimental Results

For evaluation of our algorithm, we have used the Middlebury video sequence database [4] and computed optical flow from frame 10 to 11 of several video sequences. In all these experiments, we used a step size $\tau = 1/6$, $\lambda = 100$ and $\alpha = 0.35$. The mean end point error (MEPE) and average angular error (AAE) are computed as performance measures for all algorithms:

$$MEPE = \sqrt{(v_1 - v_{1GT})^2 + (v_2 - v_{2GT})^2},$$

$$AAE = \arccos \left[\frac{1 + v_1 v_{1GT} + v_2 v_{2GT}}{\sqrt{1 + v_1^2 + v_2^2} \sqrt{1 + v_{1GT}^2 + v_{2GT}^2}} \right],$$

where $[v_{1GT} \; v_{2GT}]$ is the ground-truth flow. For neighborhood dependent data term, we have selected the Gaussian kernel of size 5×5 and standard deviation 0.5. To accurately estimate larger displacements, we used Laplacian pyramid with a downsampling factor of 0.7; the maximum number of pyramid levels is restricted to 30 and 5 warps are used per level. In this way, coarser estimates are passed on to finer stages to obtain final optical flow estimation. The gradients are estimated using the forward difference operators. Furthermore, the discrete divergence at image indices

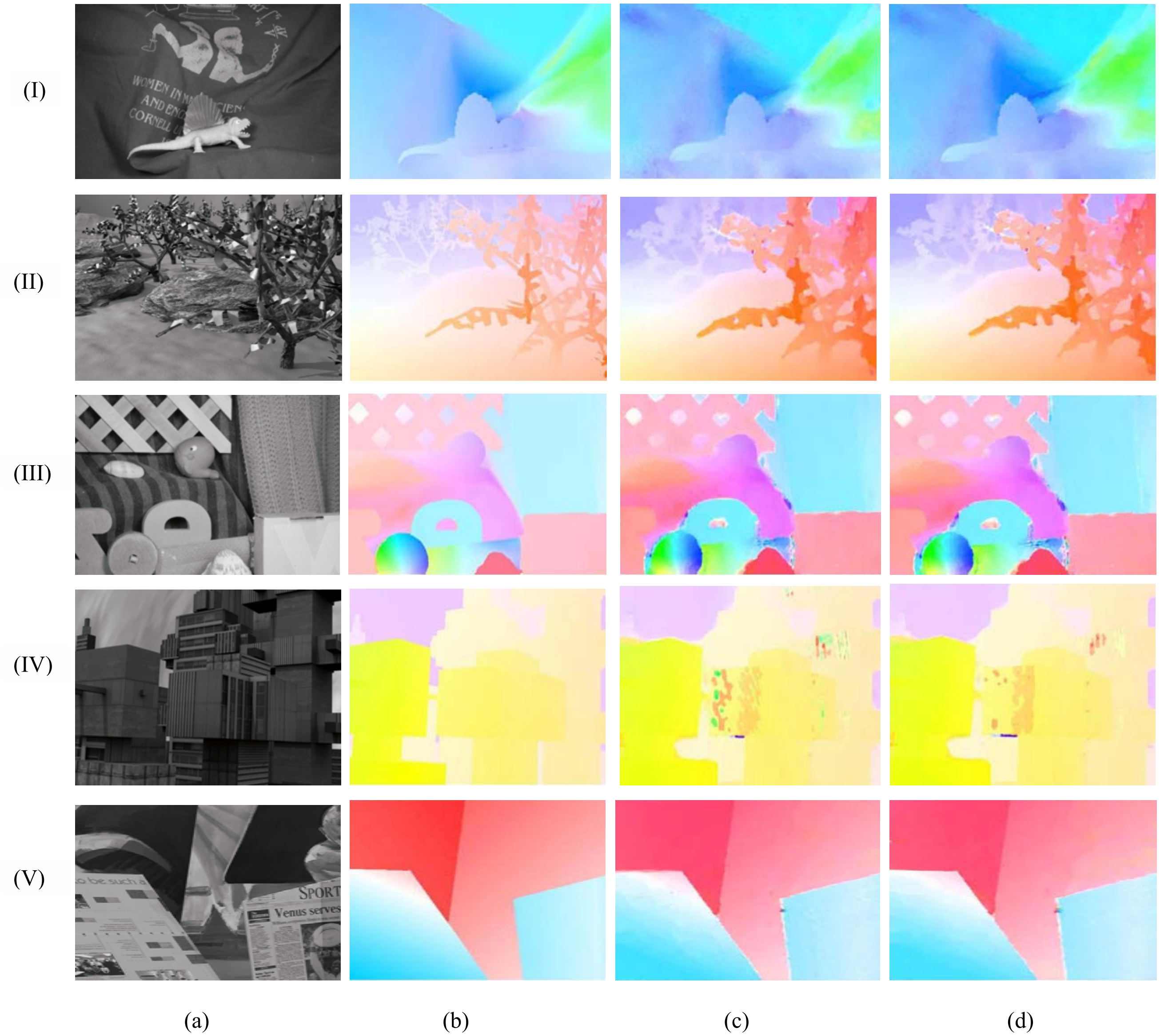
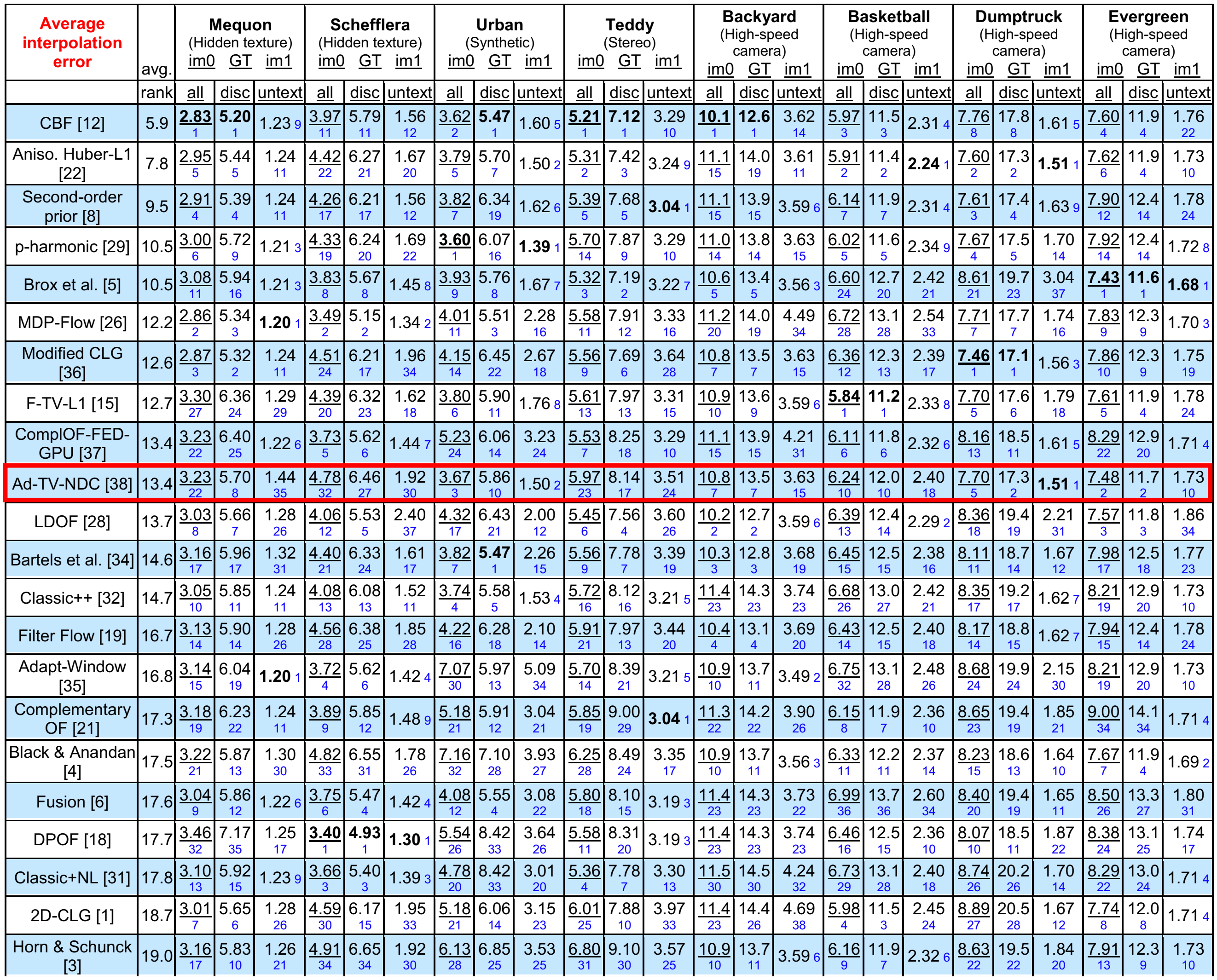


Figure 2. Flow computed on (I) Dimetrodon, (II) Grove3, (III) Rubberwhale, (IV) Urban3 and (V) Venus video sequences respectively (Each row corresponds to one video sequence). Column (a) shows frame 10 of video sequences. (b) Ground truth flow. (c) Flow computed by using I-TV-$\ell 1$. (d) Flow computed by the proposed method Ad-TV-NDC. Note that edges are better preserved by Ad-TV-NDC in the last video sequence (Venus).

Table 1. Average Angular Error from different algorithms applied on 8 image sequences of Middlebury database.

Algorithm	Dimetrodon	Grove2	Grove3	Hydrangea	Rubberwhale	Urban2	Urban3	Venus
I-TV-L1	4.1511	2.7113	6.2494	2.8037	4.7117	3.3129	6.0434	4.1717
Ad-TV-L1	4.1370	2.7049	6.1369	2.9244	**4.5241**	3.2494	5.8009	4.1072
TV-NDC	3.4116	2.3195	6.0721	**2.6820**	4.8273	3.0659	5.3522	3.8662
Ad-TV-NDC	**3.3758**	**2.2024**	**5.8027**	2.7493	4.5483	**2.9713**	**5.0359**	**3.7050**

I-TV-$\ell 1$=Improved TV-$\ell 1$,Ad-TV-$\ell 1$= TV-$\ell 1$ with Adaptive Regularization, TV-NDC= Neighborhood Dependent Constraint with Total Variation, Ad-TV-NDC (Neighborhood Dependent Constraint with Adaptive Total Variation Regularization).



Average interpolation error	avg.	Mequon (Hidden texture) im0 GT im1			Schefflera (Hidden texture) im0 GT im1			Urban (Synthetic) im0 GT im1			Teddy (Stereo) im0 GT im1			Backyard (High-speed camera) im0 GT im1			Basketball (High-speed camera) im0 GT im1			Dumptruck (High-speed camera) im0 GT im1			Evergreen (High-speed camera) im0 GT im1		
	rank	all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext	all	disc	untext
CBF [12]	5.9	**2.83** 1	**5.20** 1	1.23 9	3.97 11	5.79 11	1.56 12	3.62 2	**5.47** 1	1.60 5	**5.21** 1	**7.12** 1	3.29 10	**10.1** 1	**12.6** 1	3.62 14	5.97 3	11.5 3	2.31 4	7.76 8	17.8 8	1.61 5	7.60 4	11.9 4	1.76 22
Aniso. Huber-L1 [22]	7.8	2.95 5	5.44 5	1.24 11	4.42 22	6.27 21	1.67 20	3.79 5	5.70 7	1.50 2	5.31 2	7.42 3	3.24 9	11.1 15	14.0 19	3.61 11	5.91 2	11.4 2	**2.24** 1	7.60 2	17.3 2	**1.51** 1	7.62 6	11.9 4	1.73 10
Second-order prior [8]	9.5	2.91 4	5.39 4	1.24 11	4.26 17	6.21 17	1.56 12	3.82 7	6.34 19	1.62 6	5.39 5	7.68 5	**3.04** 1	11.1 15	13.9 15	3.59 6	6.14 7	11.9 7	2.31 4	7.61 3	17.4 4	1.63 9	7.90 12	12.4 14	1.78 24
p-harmonic [29]	10.5	3.00 6	5.72 9	1.21 3	4.33 19	6.24 20	1.69 22	**3.60** 1	6.07 16	**1.39** 1	5.70 14	7.87 9	3.29 10	11.0 14	13.8 14	3.63 15	6.02 5	11.6 5	2.34 9	7.67 4	17.5 5	1.70 14	7.92 14	12.4 14	1.72 8
Brox et al. [5]	10.5	3.08 11	5.94 16	1.21 3	3.83 8	5.67 8	1.45 8	3.93 9	5.76 8	1.67 7	5.32 3	7.19 2	3.22 7	10.6 5	13.4 5	3.56 3	6.60 24	12.7 20	2.42 21	8.61 21	19.7 23	3.04 37	**7.43** 1	**11.6** 1	**1.68** 1
MDP-Flow [26]	12.2	2.86 2	5.34 3	**1.20** 1	3.49 2	5.15 2	1.34 2	4.01 11	5.51 3	2.28 16	5.58 11	7.91 12	3.33 16	11.2 20	14.0 19	4.49 34	6.72 28	13.1 28	2.54 33	7.71 7	17.7 7	1.74 16	7.83 9	12.3 9	1.70 3
Modified CLG [36]	12.6	2.87 3	5.32 2	1.24 11	4.51 24	6.21 17	1.96 34	4.15 14	6.45 22	2.67 18	5.56 9	7.69 6	3.64 28	10.8 7	13.5 7	3.63 15	6.36 12	12.3 13	2.39 17	**7.46** 1	**17.1** 1	1.56 3	7.86 10	12.3 9	1.75 19
F-TV-L1 [15]	12.7	3.30 27	6.36 24	1.29 29	4.39 20	6.32 23	1.62 18	3.80 6	5.90 11	1.76 8	5.61 13	7.97 13	3.31 15	10.9 10	13.6 9	3.59 6	**5.84** 1	**11.2** 1	2.33 8	7.70 5	17.6 6	1.79 18	7.61 5	11.9 4	1.78 24
ComplOF-FED-GPU [37]	13.4	3.23 22	6.40 25	1.22 6	3.73 5	5.62 6	1.44 7	5.23 24	6.06 14	3.23 24	5.53 7	8.25 18	3.29 10	11.1 15	13.9 15	4.21 31	6.11 6	11.8 6	2.32 6	8.16 13	18.5 11	1.61 5	8.29 22	12.9 20	1.71 4
Ad-TV-NDC [38]	13.4	3.23 22	5.70 8	1.44 35	4.78 32	6.46 27	1.92 30	3.67 3	5.86 10	1.50 2	5.97 23	8.14 17	3.51 24	10.8 7	13.5 7	3.63 15	6.24 10	12.0 10	2.40 18	7.70 5	17.3 2	**1.51** 1	7.48 2	11.7 2	1.73 10
LDOF [28]	13.7	3.03 8	5.66 7	1.28 26	4.06 12	5.53 5	2.40 37	4.32 17	6.43 21	2.00 12	5.45 6	7.56 4	3.60 26	10.2 2	12.7 2	3.59 6	6.39 13	12.4 14	2.29 2	8.36 18	19.4 19	2.21 31	7.57 3	11.8 3	1.86 34
Bartels et al. [34]	14.6	3.16 17	5.96 17	1.32 31	4.40 21	6.33 24	1.61 17	3.82 7	**5.47** 1	2.26 15	5.56 9	7.78 7	3.39 19	10.3 3	12.8 3	3.68 19	6.45 15	12.5 15	2.38 16	8.11 11	18.7 14	1.67 12	7.98 17	12.5 18	1.77 23
Classic++ [32]	14.7	3.05 10	5.85 11	1.24 11	4.08 13	6.08 13	1.52 11	3.74 4	5.58 5	1.53 4	5.72 16	8.12 16	3.21 5	11.4 23	14.3 23	3.74 23	6.68 26	13.0 27	2.42 21	8.35 17	19.2 17	1.62 7	8.21 19	12.9 20	1.73 10
Filter Flow [19]	16.7	3.13 14	5.90 14	1.28 26	4.56 28	6.38 25	1.85 28	4.22 16	6.28 18	2.10 14	5.91 21	7.97 13	3.44 20	10.4 4	13.1 4	3.69 20	6.43 14	12.5 15	2.40 18	8.17 14	18.8 15	1.62 7	7.94 15	12.4 14	1.78 24
Adapt-Window [35]	16.8	3.14 15	6.04 19	**1.20** 1	3.72 4	5.62 6	1.42 4	7.07 30	5.97 13	5.09 34	5.70 14	8.39 21	3.21 5	10.9 10	13.7 11	3.49 2	6.75 32	13.1 28	2.48 26	8.68 24	19.9 24	2.15 30	8.21 19	12.9 20	1.73 10
Complementary OF [21]	17.3	3.18 19	6.23 22	1.24 11	3.89 9	5.85 12	1.48 9	5.18 21	5.91 12	3.04 21	5.85 19	9.00 29	**3.04** 1	11.3 22	14.2 22	3.90 26	6.15 8	11.9 7	2.36 10	8.65 23	19.4 19	1.85 21	9.00 34	14.1 34	1.71 4
Black & Anandan [4]	17.5	3.22 21	5.87 13	1.30 30	4.82 33	6.55 31	1.78 26	7.16 32	7.10 28	3.93 27	6.25 28	8.49 24	3.35 17	10.9 10	13.7 11	3.56 3	6.33 11	12.2 11	2.37 14	8.23 15	18.6 13	1.64 10	7.67 7	11.9 4	1.69 2
Fusion [6]	17.6	3.04 9	5.86 12	1.22 6	3.75 6	5.47 4	1.42 4	4.08 12	5.55 4	3.08 22	5.80 18	8.10 15	3.19 3	11.4 23	14.3 23	3.73 22	6.99 36	13.7 36	2.60 34	8.40 20	19.4 19	1.65 11	8.50 26	13.3 27	1.80 31
DPOF [18]	17.7	3.46 32	7.17 35	1.25 17	**3.40** 1	**4.93** 1	**1.30** 1	5.54 26	8.42 33	3.64 26	5.58 11	8.31 20	3.19 3	11.4 23	14.3 23	3.74 23	6.46 16	12.5 15	2.36 10	8.07 10	18.5 11	1.87 22	8.38 24	13.1 25	1.74 17
Classic+NL [31]	17.8	3.10 13	5.92 15	1.23 9	3.66 3	5.40 3	1.39 3	4.78 20	8.42 33	3.01 20	5.36 4	7.78 7	3.30 13	11.5 30	14.5 30	4.24 32	6.73 29	13.1 28	2.40 18	8.74 26	20.2 26	1.70 14	8.29 22	13.0 24	1.71 4
2D-CLG [1]	18.7	3.01 7	5.65 6	1.28 26	4.59 30	6.17 15	1.95 33	5.18 21	6.06 14	3.15 23	6.01 25	7.88 10	3.97 33	11.4 23	14.4 26	4.69 38	5.98 4	11.5 3	2.45 24	8.89 27	20.5 28	1.67 12	7.74 8	12.0 8	1.71 4
Horn & Schunck [3]	19.0	3.16 17	5.83 10	1.26 21	4.91 34	6.65 34	1.92 30	6.13 28	6.85 25	3.53 25	6.80 31	9.10 30	3.57 25	10.9 10	13.7 11	3.59 6	6.16 9	11.9 7	2.32 6	8.63 22	19.5 22	1.84 20	7.91 13	12.3 9	1.73 10

Figure 3. Optical flow evaluation results from Middlebury's benchmark, note that proposed method ranks $8th$ in *average* interpolation error calculation, among 38 methods evaluated on the same benchmark. For details, refer to http://vision.middlebury.edu/flow/.

Table 2. Mean End Point Error from different algorithms applied on 8 image sequences of Middlebury database.

Algorithm	Dimetrodon	Grove2	Grove3	Hydrangea	Rubberwhale	Urban2	Urban3	Venus
I-TV-L1	0.2070	0.1952	0.6048	0.2358	0.1482	0.4586	0.7093	0.2764
Ad-TV-L1	0.2065	0.1943	0.5875	0.2420	**0.1438**	0.4509	0.7002	0.2884
TV-NDC	0.1762	0.1616	0.6084	**0.2282**	0.1514	**0.4174**	0.6217	0.2578
Ad-TV-NDC	**0.1743**	**0.1531**	**0.5746**	0.2308	0.1441	0.4239	**0.5968**	**0.2480**

I-TV-$\ell1$=Improved TV-$\ell1$,Ad-TV-$\ell1$= TV-$\ell1$ with Adaptive Regularization, TV-NDC= Neighborhood Dependent Constraint with Total Variation, Ad-TV-NDC (Neighborhood Dependent Constraint with Adaptive Total Variation Regularization).

i and j is estimated as

$$\mathbf{div}(p_{i,j}) = \begin{cases} p^1_{i,j} + p^2_{i,j} & \text{if } i,j = 1 \\ -p^1_{i-1,j} - p^2_{i,j-1} & i = M, j = N \\ p^1_{i,j} - p^1_{i-1,j} + p^2_{i,j} - p^2_{i,j-1} & \text{otherwise} \end{cases}$$

where M and N are the height and width of the image respectively.

We have also implemented the improved TV-$\ell1$ algorithm with same parameters and compared the performance of our method with this algorithm. It is clear from the results in Tables 1 and 2 that using neighborhood dependent constraint improves on performance, as compared to the improved TV-$\ell1$ algorithm. Quality is further improved by using structure adaptive total variation regularization. The algorithm also performs well in the presence of additive Gaussian noise as depicted in Figure 1. The improved quality of the method can be seen in high textured region located in the middle of Urban3 sequence presented in Figure 2. Moreover, edges are better preserved in Venus sequence compared to improved TV-$\ell1$ algorithm. We have also implemented the proposed method on Middlebury's evaluation video datasets. For comparison with existing optical flow estimation methods, we have taken a screenshot from http://vision.middlebury.edu/flow/eval/results/results-i1.php, refer to Figure 3. Ad-TV-NDC (proposed method) outperforms all other methods (refer to numbers under error results in Figure 3) for high speed camera image sequence Dumptruck and is ranked second for synthetic image sequence Urban. However, the performance of our method decreases for image sequences containing hidden textures e.g., Mequon and Schefflera. When an *average* is taken over interpolation error calculation of all image sequences, the proposed method ranks $8th$, among 38 methods evaluated on the same benchmark.

6. Conclusion

In this paper, we presented an optical flow estimation method that uses neighborhood dependent data constraint in a variational framework. To better handle occlusions, we have used inhomogeneous regularization. We have shown that a neighborhood dependent constraint can better match intensities across image pairs than traditional data term used in modern techniques. Experimental results are presented which show that the proposed method can better identify correspondence between image pairs compared to methods based on traditional data constraint and can better preserve strong edges by using inhomogeneous regularization. It will be interesting to use data terms with more sophisticated neighborhood dependency in future work.

References

[1] E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of motion. *Journal of Optical Society of America A*, 2(2):284–299, Feb. 1985. 1

[2] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2:283–310, 1989. 1

[3] J. F. Aujol, G. Gilboa, T. Chan, and S. J. Osher. Structure-texture image decomposition: Modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67(1):111–136, 2006. 3

[4] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. Technical report, Microsoft Research Microsoft Corporation, One Microsoft Way, Redmond, WA 98052, December 2009. 3

[5] J. L. Barron, D. J. Fleet, and S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12:43–77, Sep 1994. 1

[6] S. Beauchemin and J.L.Barron. The computation of optical flow. *ACM computing Survey*, 27(3):433–467, Sep. 1995. 1

[7] M. J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Int. Conf. on Computer Vision, ICCV -93, Berlin*, pages 231–236, May 1993. 1

[8] A. Bruhn, J. Weickert, and C. Schnorr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International Journal of Computer Vision*, 61(3):211–231, 2005. 1, 2

[9] A. Chambolle. *Energy Minimization Methods in Computer Vision and Pattern Recognition*, chapter Total Variation Minimization and a Class of Binary MRF Models, pages 136–152. Springer Berlin / Heidelberg, 2005. 2, 3

[10] D. Fleet and A. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5:77–104, 1990. 1

[11] D. J. Fleet and Y. Weiss. *Mathematical Models in Computer Vision: The Handbook*, chapter Optical Flow Estimation, pages 239–258. Springer, 2005. 1

[12] D. Heeger. Optical flow using spatiotemporal filters. *International Journal of Computer Vision*, 1:279–302, 1988. 1

[13] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, vol. 17(8):185–203, August 1981. 1

[14] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. International Joint Conference on Artificial Intelligence*, page 674679, Aug 1981. 1, 2

[15] H. H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(5):565–593, Sep 1986. 1, 2

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

[17] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992. 2

[18] F. Steinbrucker, T. Pock, and D. Cremers. Advanced data terms for variational optic flow estimation. In *VMV*, 2009. 1

[19] A. Wedel, D. Cremers, T. Pock, and H. Bischof. Structure and motion adaptive regularization for high accuracy optic flow. In *ICCV*, 2009. 1, 2

[20] A. Wedel, T. Pock, C. Zach, H. Bischof, and D. Cremers. An improved algorithm for tv-l1 optical flow. In *Dagstuhl Motion Workshop*, volume 4713, pages 214–223, 2008. 1

[21] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, and H. Bischof. Anisotropic huber l1 optical flow. In *BMVC*, 2009. 1

[22] C. Zach, T. Pock, and H. Bischof. A duality based approach for real time tv-l1 optical flow. *Pattern Recognition*, 4713:214–223, 2007. 1