Pattern Recognition 50 (2016) 223–232

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Weighted local intensity fusion method for variational optical flow estimation ☆

Zhigang Tu *, Ronald Poppe, Remco C. Veltkamp

Department of Information and Computing Sciences, Utrecht University Princetonplein 5, Utrecht, Netherlands

ARTICLE INFO

Article history:
Received 3 January 2015
Received in revised form
30 June 2015
Accepted 1 September 2015
Available online 14 September 2015

Keywords:
Optical flow
Weighted local intensity fusion
Large displacement
Smoothness parameter estimation
Corrected weighted median filter

ABSTRACT

Estimating a dense motion field of successive video frames is a fundamental problem in image processing. The multi-scale variational optical flow method is a critical technique that addresses this issue. Despite the considerable progress over the past decades, there are still some challenges such as dealing with large displacements and estimating the smoothness parameter. We present a local intensity fusion (LIF) method to tackle these difficulties. By evaluating the local interpolation error in terms of L1 block match on the corresponding set of images, we fuse flow proposals which are obtained from different methods and from different parameter settings integrally under a unified LIF. This approach has two benefits: (1) the incorporated matching information is helpful to recover large displacements; and (2) the obtained optimal fusion solution gives a tradeoff between the data term and the smoothness term. In addition, a selective gradient based weight is introduced to improve the performance of the LIF. Finally, we propose a corrected weighted median filter (CWMF), which applies the motion information to correct errors of the color distance weight to denoise the intermediate flow fields during optimization. Experiments demonstrate the effectiveness of our method.

1. Introduction

Motion in the form of optical flow is one of the most significant bottom-up cues for various computer vision and image processing tasks. Horn and Schunck (HS) [1] proposed a variational method to compute optical flow based on a brightness constancy assumption (BCA) constraint. This single constraint is insufficient to determine two unknown components of the motion field, which is termed the *aperture problem*. In particular, this problem denotes the ambiguity in motion perception, i.e. the direction of the motion is ambiguous. For a given pixel one first image, there could be many correspondences in the subsequent image. To solve this problem, Horn and Schunck introduced a smoothness term which integrates information from the entire image. A parameter λ was used to control the balance of these two terms.

Estimating dense flow fields in realistic videos presents many challenges, which causes the variational method to suffer from a significant deterioration [2]. For example, multiple moving objects could occlude each other and the HS model lacks robustness to handle occlusions and motion discontinuities. Second, illumination changes and motion blur seriously violate the BCA. Third, when displacements are larger than the object structure, matching typically fails due to the limitations of Taylor expansion. Finally, different types of video frames might have different qualities and a fixed smoothness parameter is therefore an unsuitable factor to balance the data and smoothness terms. The selection of an appropriate λ is important but difficult [3,4]. In this paper, we address these two problems and present a weighted local intensity fusion (WLIF) method to handle both in a unified way.

Multi-scale variational optical flow methods that employ a coarse-to-fine strategy [5,6] have become the predominant way to estimate a dense motion fields. These methods can deal with large displacements but have difficulty estimating motion details in the presence of large displacements [7]. The basic sub-sample manner reduces the size of the image structures and the displacements, which causes some motion details to be removed. In the refinement process, the inaccurate initialization propagated from the coarse level leads to incorrect motion estimation at finer levels. Therefore, the motion of small structures with large displacements cannot be estimated accurately at the final scale. Recently, matching information [8,9] is introduced to integrate with the variational method to handle this problem. However, nearly all of the work suffers from difficulties with small displacements. Inspired by the FusionFlow method [10], the local data fidelity method [4] and the weighted root mean square (WRMS) method

☆This publication was supported by the Dutch national program COMMIT
* Corresponding author.
E-mail addresses: z.tu@uu.nl (Z. Tu), r.w.poppe@uu.nl (R. Poppe), r.c.veltkamp@uu.nl (R.C. Veltkamp).

http://dx.doi.org/10.1016/j.patcog.2015.09.002
0031-3203/

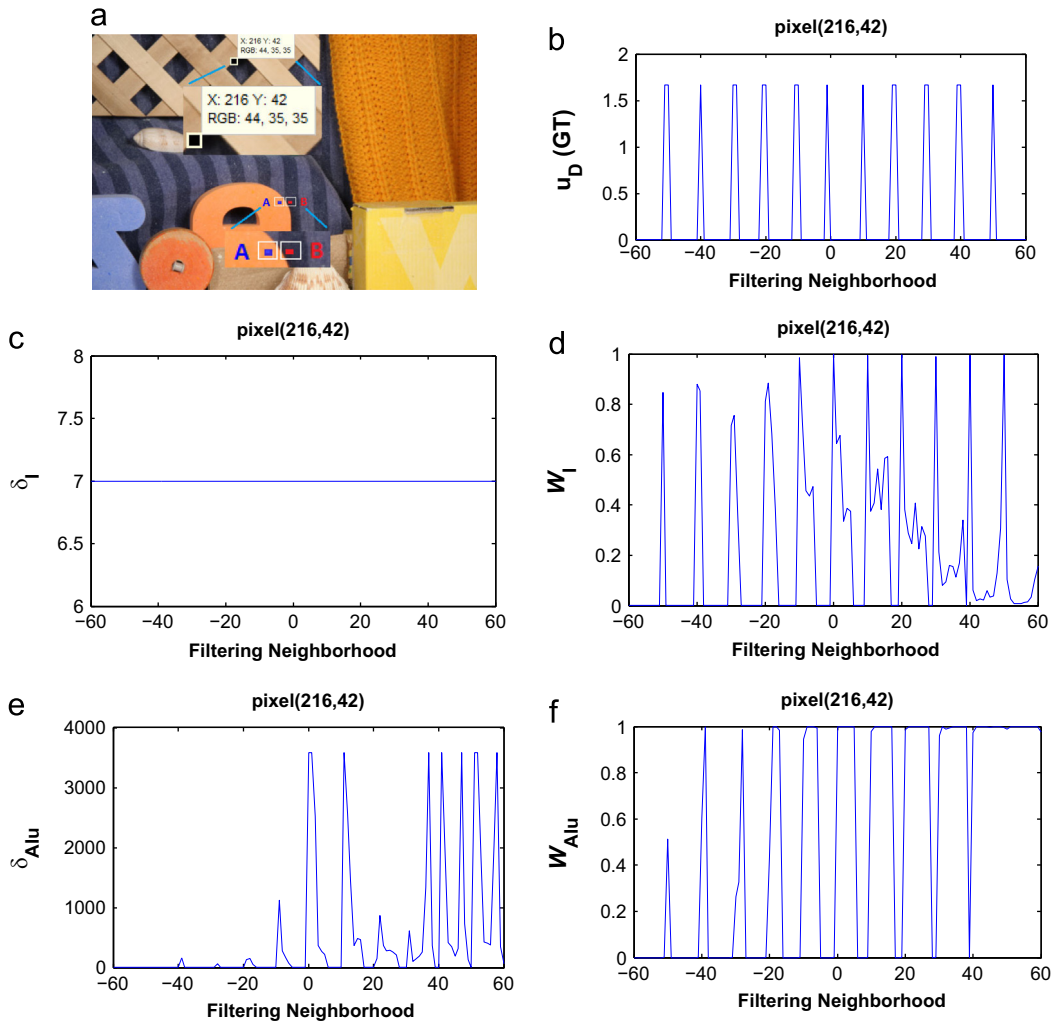

Fig. 1. Comparison of our CWMF with the traditional WMF. (a) Selected edge pixel (216,42). (b) Horizontal GT motion difference u_D in its filtering window. (c) Standard deviation σ_I of WMF. (d) Color weight w_I of WMF. (e) Our adaptive standard deviation σ_{AI}. (f) Our corrected color weight w_{AI}.

[11], we use the WLIF to fuse multiple flows of different λ to form a single, improved flow estimation.

Applying a median filter (MF) [12,13] to intermediate flow values during optimization is helpful to improve the accuracy of the estimated flow field. However, the MF over-smoothes some details, especially small structures and corners. Sun et al. [14] proposed a weighted version to handle the over-smoothing problem. Unfortunately, the weighted median filter (WMF) has one crucial drawback: the color measure does not always coincide with the motion measure (see Fig. 1). Accordingly, the color measure deteriorates the performance of the WMF. To address this, we use the motion information and present a corrected weighted median filter (CWMF).

We make the following two contributions in this work:

1. *Weighted local intensity fusion* (*WLIF*): To handle large displacements and estimate the smoothness parameter simultaneously, we present a local intensity fusion (LIF) method to deal with both in a unified framework. To further improve the fusion performance, we use selective gradient magnitude as the weight to LIF. The WLIF reduces errors of LIF that are caused by outliers and occlusions.
2. *Corrected weighted median filter* (*CWMF*): To reduce the error of the color distance weight, we first find the points whose color similarity is in conflict with their motion similarity in the filtering window. Then, the motion of these points is used to help correct their color weights.

The remainder of the paper is organized as follows. We first review related work on dense optical flow estimation. Section 3 describes our approach. We then evaluate our method qualitatively and quantitatively and conclude in Section 5.

2. Related work

The variational method of Horn and Schunck [1] combined with a coarse-to-fine warping strategy [6] has become the predominant approach to compute optical flow. Although advances have been made in recent years, the variational framework has difficulty to obtain good results in challenging situations such as those with occlusions [15], illumination changes [16–18], motion discontinuities [19,20], large displacements [21,22] and real-time computation [23–25]. As the image resolution of consumer video cameras is consistently increasing, large displacement is an important factor in dense motion estimation.

To address the large displacement problem, virtually all variational methods adopt a coarse-to-fine warping strategy. This strategy has difficulty in recovering the motion of structures whose scale is smaller than their displacement [26]. One reason is that linearization of the data term with the Taylor expansion is only accurate for small motions. Alvarez et al. [27] addressed this problem by applying a linear scale-space focusing scheme to avoid converging to incorrect local minima. This method only partially

treats the large displacement issue as it still depends on sub-sample warping. Steinbruecker et al. [28] presented a scheme to avoid both linearization and warping. Their algorithm requires an exhaustive search for pixel-level candidate matching, which is too computational expensive to make the estimation intractable. Rhemann et al. [29] proposed a fast edge-preserving filter to smooth the label costs to ensure that flow field contain both fine structures and large displacements. The method does not achieve sub-pixel accuracy.

Recently, there has been an interest in using feature matching to assist the variational methods to handle large displacements. The SIFT-flow method [30] uses SIFT [31] descriptors to compute a dense scene flow field. Since this method fully depends on SIFT descriptors, it performs poorly at small-motion regions [9]. Brox et al. [32] modified [30] by only considering the best matches among the detected range of features. This method has some issues. First, the local descriptors are reliable only at salient locations and are locally rigid. Second, there can be false matches. Stoll et al. [33] presented an adaptive integration strategy for feature matches to reduce these false point matches. Weinzaepfel [9] proposed DeepFlow which involved dense and deformable matching, to gain performance for fast motion. However, these matching methods still sustain the problem of false matches and low precision. The fusion method [10,7] is an alternative way to deal with large displacements. Lempitsky et al. [10] proposed to fuse flow candidates which are obtained from different flow methods or one flow method with different parameter settings. This produces a superior flow field. Derived from the fusion idea, [7] used the selected sparse SIFT matching and the PatchMatch [34] to find candidates, then fuse them with the flow proposal. In contrast, [8] utilized approximate nearest neighbor fields (NNF) to compute an initial motion field which contains different types of correspondence information, and then fuses it with the variational flow candidate for refinement. However, their fusion method (quadratic pseudo-boolean optimization, QPBO [35,36]) has some drawbacks: its submodularity constraint imposed on the pairwise terms is hard to fully meet, and the constraint is not suitable for variational optical flow models.

Estimating the smoothness parameter λ to obtain an optimal trade-off between the data and smoothness terms is of crucial importance. Inspired by the Optimal Prediction Principle (OPP) technique [3] which automatically determines the optimal smoothness parameter, [4] presented a local data fidelity method to fuse optical flow estimates of different smoothness. The local evaluation in terms of the best data fit on the gradient images is easily affected by illumination changes, and is sensitive to noise and discontinuities. Tu et al. [11] proposed a weighted root mean square (WRMS) error method to handle this problem but this is a global measure. The obtained "optimal" smoothness λ is not always the best for each point.

Applying a median filter (MF, [12]) to denoise the flow field is a good way to improve accuracy although over-smoothing typically occurs for edges and corners. To prevent this, [14] proposed a modified weighted median filter (WMF) which relies on the spatial distance, the color distance and the occlusion state. Unfortunately, the color distance weight would be harmful when the color similarity is inconsistent with the motion similarity.

Our work is related to [7] and [8] in terms of fusing flow proposals obtained with different algorithms to handle both large and small displacements. The difference is that we utilize the WLIF method instead of the QPBO method for fusion. Our work is related to [4] in terms of fusing flow proposals obtained with different smoothness parameters to get an optimal balance between the data term and the smoothness term. Again, WLIF is used, this time to replace to local data fidelity term on the gradient images. We adapt the work of [12,14] by using CWMF to filter the intermediate flow fields.

3. Weighted local intensity fusion

In this section, we introduce the parts of our algorithm. First, we describe the NN-field algorithm [8], which formulates the motion estimation as a motion segmentation. Then, we introduce the modified Classic+CPF algorithm [37]. The edge-preserving regularization and the occlusion-aware refinement are utilized to compute a sub-pixel motion field. Third, we describe the WLIF method to fuse the obtained NN-field flow and the sub-pixel flow for continuous flow refinement. WLIF is also used to fuse flow candidates computed with different smoothness parameters λ. Finally, we present CWMF, a method to denoise the intermediate flow fields during optimization.

3.1. The NN-field algorithm

To handle large displacements, PatchMatch [34] correspondences are introduced and the motion patterns are computed from a noisy nearest neighbor field (NN-field). We then perform motion segmentation [38] with dominant motion patterns to clean up the noise. Specifically, the dense motion estimation is formulated as a labeling problem [8]:

$$E(u,v)=\sum_{\mathbf{x}}\rho_D(I_2(\mathbf{x}+\mathbf{w}(\mathbf{x}))-I_1(\mathbf{x}))+\sum_{(\mathbf{x},\mathbf{x}')|\,\mathbf{x}\in N_c(\mathbf{x}')}\rho_S(\mathbf{w}(\mathbf{x})-\mathbf{w}(\mathbf{x}'))$$
$$s.t.\quad \mathbf{w}(\mathbf{x})\in\{\Omega(\mathbf{w}_1),\ldots,\Omega(P_1\circ\mathbf{x})\} \tag{1}$$

where $\mathbf{x}=(x,y)$ denotes a point in the image domain Ω. $P=\{P_1,\ldots,P_J\}$ are the dominant projection matrices. $\mathbf{w}(\mathbf{x})=(u_0,v_0)$ is the motion pattern. I_1 and I_2 are the input images. $N_c(\mathbf{x})$ denotes the 4-connected neighbors of $\mathbf{x}$. ρ_D and ρ_S are robust functions, where $\rho_S(\mathbf{x})=\rho_D(\mathbf{x})=\rho(\mathbf{x})=(\mathbf{x}^2+\xi^2)^{\alpha}, \alpha=0.45, \xi=0.001$ [6]. To better preserve edges, the advanced edge preserving motion smoothness [8,39] is applied:

$$\rho_S(\mathbf{w}(\mathbf{x})-\mathbf{w}(\mathbf{x}'))=\omega(\mathbf{x})\rho(\mathbf{w}(\mathbf{x})-\mathbf{w}(\mathbf{x}')) \tag{2}$$

where $\omega(\mathbf{x})$ is the simple structure adaptive map that maintains motion discontinuity [7]:

$$\omega(\mathbf{x})=\exp(-\|\nabla I_1\|^k) \tag{3}$$

where $k=0.8$. The NN-field $\mathbf{w}(\mathbf{x})$ is computed by optimizing Eq. (1) using [8]. The $\mathbf{w}(\mathbf{x})$ will be combined with the variational flow $\mathbf{v}(\mathbf{x})$ as the refined initial flow in our WLIF method. This will be done at each image level before the first warping during the coarse-to-fine optimization. The $\mathbf{w}(\mathbf{x})$ supplies accurate dense matching information between large and small distance patches and the refined fusion flow contains reliable motion details which can serve as a initialization for continuous refinement. Hence, the NN-field is helpful in recovering the motion of structures with displacements larger than their size.

3.2. The modified Classic+CPF algorithm

We apply the Classic+CPF algorithm [37] to estimate a sub-pixel flow field. The CPF (combined post-filtering) is a combined filtering technique that uses a weighted median filter (WMF), a bilateral filter (BF) and a fast MF [12] to smooth the detected edges, occlusions, and flat regions of the intermediate flow field, respectively. Its energy functional is defined as

$$\begin{aligned}E(u,v,\overline{u},\overline{v})=&\sum_{\mathbf{x}}\Psi_D(\|I_2(\mathbf{x}+\mathbf{v}(\mathbf{x}))-I_1(\mathbf{x})\|)+\lambda\Psi_S(\|\nabla\mathbf{v}(\mathbf{x})\|)\\&+\lambda_2(\|u-\overline{u}\|+\|v-\overline{v}\|)+\sum_{x_E,y_E x'_E,y'_E\in N_{x_E,y_E}}w_{|(x_E,y_E,x'_E,y'_E)}(|\overline{u}_{x,y}-\overline{u}_{x',y'}|\\&+|\overline{v}_{x,y}-\overline{v}_{x',y'}|)+\sum_{x_O,y_O x'_O,y'_O\in N_{x_O,y_O}}w_{x_O,y_O,x'_O,y'_O}(|\overline{u}_{x,y}-\|u_{x',y'}|+|\|v_{x,y}\\&-\|v_{x',y'}|)+\sum_{x_F,y_F x'_F,y'_F\in N_{x_F,y_F}}median(|\overline{u}_{x,y}-\overline{u}_{x',y'}|+|\overline{v}_{x,y}-\overline{v}_{x',y'}|)\end{aligned} \tag{4}$$

Table 1
AAE/EPE for the tested sequences from the Middlebury benchmark (no CWMF operations) and comparison with WLIF with different window sizes.

Window	Urban2	Urban3	Grove2	Grove3	RubW.	Venus	Dime.	Hydra.	Avg.Ratio
3×3	1.835/0.229	2.748/0.496	1.269/0.090	4.227/0.420	2.132/0.067	2.972/0.219	2.764/0.144	1.895/0.155	1.004/1.006
5×5	1.837/0.230	2.702/0.489	1.269/0.090	4.224/0.421	2.129/0.067	2.959/0.218	2.758/0.140	1.890/0.155	1.000/1.000
7×7	2.113/0.235	2.808/0.500	1.265/0.090	4.261/0.423	2.171/0.068	2.967/0.219	2.781/0.141	1.892/0.155	1.025/1.012
11×11	1.860/0.239	2.676/0.490	1.272/0.090	4.229/0.421	2.132/0.067	2.974/0.219	2.778/0.142	1.905/0.156	1.003/1.007

Table 2
AAE/EPE for the tested sequences from the Middlebury benchmark with four different fusion methods (no λ fusion and no CWMF operations) and the comparison with WLIF.

Method	Urban2	Urban3	Grove2	Grove3	RubW.	Venus	Dime.	Hydra.	Avg.Ratio
Raket	1.898/0.243	2.939/0.450	1.249/0.088	4.240/0.419	2.090/0.066	2.959/0.220	2.632/0.136	1.885/0.154	1.025/1.032
QPBO	1.880/0.250	3.024/0.558	1.250/0.088	4.285/0.419	2.215/0.069	2.932/0.215	2.613/0.136	1.888/0.155	1.035/1.098
LIF	1.926/0.247	3.012/0.453	1.246/0.089	4.202/0.415	2.121/0.067	2.983/0.221	2.669/0.136	1.881/0.154	1.033/1.035
WLIF	1.908/0.240	2.603/0.416	1.214/0.085	4.202/0.412	2.101/0.065	2.904/0.218	2.605/0.133	1.875/0.152	1.000/1.000

where $\mathbf{v}(\mathbf{x})=(u,v)$ denotes the continuous flow field, and *median* refers to fast MF calculation. Compared to [37], we use the Matlab built-in Sobel function to faster extract flow edges. We also replace the smoothness term of [37] with an advanced edge-preserving smoothness term [8,39]. To handle occlusions, the occlusion-aware refinement strategy of [7] is employed. It uses a mapping uniqueness criterion to detect occlusion.

3.3. Weighted local intensity fusion

The Classic+CPF algorithm usually performs well for optical flow computation. However, it fails to recover the motion of structures with displacements larger than their own scales. The primary reason for this problem is that the coarse-to-fine based method initializes with zero motion. Inspired by the idea of FusionFlow [10], the reliable motion that may derive from the sparse matching [10], patch matching [7] or dense correspondence matching [8] is integrated into the continuous flow field to overcome these problems using fusion.

The QPBO [35] fusion is an efficient optimization method to minimize submodular, binary pairwise MRFs but it has some drawbacks. First, not all nodes can be labeled using the non-submodular function. As a consequence, QPBO only partly obtains an optimal solution. Second, as discussed in [4], to select the optimal flow among flow candidates, the measure that relies on the variational model (a composition of a brightness constant data term and a pairwise smoothness term) is not sufficient. The data term is the intrinsic characteristic of the optical flow method while the smoothness term is an added constraint. Therefore, the data term is more suitable than the pairwise variational model for flow candidate selection. However, QPBO is constrained to be valid only for binary pairwise models.

In line with the local fusion idea of [4], we propose a WLIF method for fusing:

$$\mathrm{WLIF}(\mathbf{u}(\mathbf{x}_i))=\sum_{\mathbf{x}\in N(\mathbf{x}_i)}\left|(\|\nabla I_{\mathrm{Error}}(\mathbf{x})\|)(I_2(\mathbf{x}+\mathbf{c}(\mathbf{x}_i))-I_1(\mathbf{x}))\right| \tag{5}$$

where $\mathbf{x}_i=(x_i,y_i)$ denotes a pixel location, $\mathbf{u}(\mathbf{x})$ is the fused flow field, $\mathbf{c}(\mathbf{x})$ refers to motion candidates (i.e. the NN-field $\mathbf{w}(\mathbf{x})$ or the variational flow field $\mathbf{v}(\mathbf{x})$), $N(\mathbf{x}_i)$ denotes a local neighborhood of $\mathbf{x}_i$ (we set the neighborhood to 5×5, see Table 1 for comparison). I_{Error} is the absolute difference between the warped image $I_2(\mathbf{x}+\mathbf{c}(\mathbf{x}))$ and I_1:

$$I_{\mathrm{Error}}(\mathbf{x})=\sqrt{(I_2(\mathbf{x}+\mathbf{c}(\mathbf{x}))-I_1(\mathbf{x}))^2+\xi} \tag{6}$$

where $\|\nabla I_{\mathrm{Error}}(\mathbf{x})\|$ is the gradient magnitude of I_{Error}. $\xi=0.001$ is a regularization parameter. To reduce the derivative error due to the numerical calculation, we introduce the selective gradient technique of [11] to compute $\nabla I_{\mathrm{Error}}$. The selective gradient based magnitude weight $\|\nabla I_{\mathrm{Error}}(\mathbf{x})\|$ gives higher weights to incorrect flow vectors and lower weights to correct ones.

The proposed WLIF method is simple. For different flow candidates, we compute the WLIF at each pixel location $\mathbf{x}_i$, and assign the flow vector from the candidates with the lowest value at $\mathbf{x}_i$ as the final fused flow. During the coarse-to-fine optimization, the flow values that are propagated from the coarse level will be corrected, because the useful matching information from the NN-field is introduced through the WLIF. Due to this improvement, the motion of structures with large displacements as well as fine structures with small displacements can be accurately recovered.

The local data fidelity measure of the WLIF method is the foundation of the optical flow technique, so it can be used to fuse flow candidates computed in different ways. In this paper, we also apply the WLIF method to blend flow candidates obtained with different smoothness parameters into a single flow. Different types of image sequences as well as different image regions have different properties and a fixed λ is inappropriate. Therefore, the data term and the smoothness term should be weighted adjustable. Table 3 and Fig. 2 demonstrate that estimation of λ is very necessary.

3.4. Corrected weighted median filter

Filtering the intermediate flow fields during incremental estimation and warping significantly improves the estimation accuracy [12,37]. However, the WMF method [14] has one severe defect: the color similarity can conflict with the motion similarity. This causes the color weight to negatively influence the final weights. To address this, we propose a CWMF method.

The weights of WMF are defined based on the spatial distance, the color-value distance, and the occlusion state:

$$w_{\mathrm{P}}(x,y,x',y')=\exp\left(-\frac{|x-x'|^2+|y-y'|^2}{2\sigma_P^2}\right) \tag{7}$$

$$w_{\mathrm{I}}(x,y,x',y')=\exp\left(-\frac{|I(x,y)-I(x',y')|^2}{2\sigma_I^2}\right) \tag{8}$$

$$w_{\mathrm{O}}(x,y,x',y')=-\frac{O(x',y')}{O(x,y)} \tag{9}$$

$$w=w_{\mathrm{P}}\cdot w_{\mathrm{I}}\cdot w_{\mathrm{O}} \tag{10}$$

where (x',y') represents the position of the neighbors of pixel (x,y) in a local neighborhood. $I(x,y)$ and $I(x',y')$ are the color vectors of (x,y) and (x',y') in the Lab space, respectively. $\sigma_P=7$ and $\sigma_I=7$ are the standard deviations. $O(x,y)$ is the occlusion state that is computed following [37].

Table 3
AAE/EPE for the tested sequences from the Middlebury benchmark (no NNF matching fusion and no CWMF operations) and the comparison with WLIF.

Method	Urban2	Urban3	Grove2	Grove3	RubW.	Venus	Dime.	Hydra.	Avg.Ratio
NoFuse	1.850/0.210	2.455/0.407	1.255/0.094	4.342/0.433	2.159/0.070	3.156/0.230	2.605/0.130	1.896/0.157	1.045/1.052
Raket	1.853/0.208	2.572/0.408	1.284/0.093	4.202/0.412	2.124/0.067	3.150/0.224	2.552/0.133	1.846/0.153	1.013/1.016
QPBO	1.882/0.208	2.443/0.412	1.280/0.092	4.186/0.413	2.175/0.068	3.193/0.227	2.724/0.139	1.891/0.156	1.082/1.026
LIF	1.875/0.210	2.495/0.410	1.293/0.093	4.203/0.415	2.148/0.068	3.147/0.224	2.586/0.140	1.865 /0.163	1.014/1.040
WLIF	1.833/0.207	2.451/0.402	1.280/0.090	4.193/0.403	2.099/0.065	3.142/0.220	2.543/0.133	1.833/0.151	1.000/1.000

Fig. 2. Visual comparison of Army [40] sample frame with different smoothness parameter λ settings. (a) Estimated flow field of $\lambda=0.5$. (b) Fused flow field of $\lambda=0.5, 1, 3, 5, 7$. (c) Estimated flow field of $\lambda=7$. (d) The corresponding λ-field of the fused flow field of $\lambda=0.5$ and $\lambda=7$, dark represents $\lambda=0.5$ and light represents $\lambda=7$.

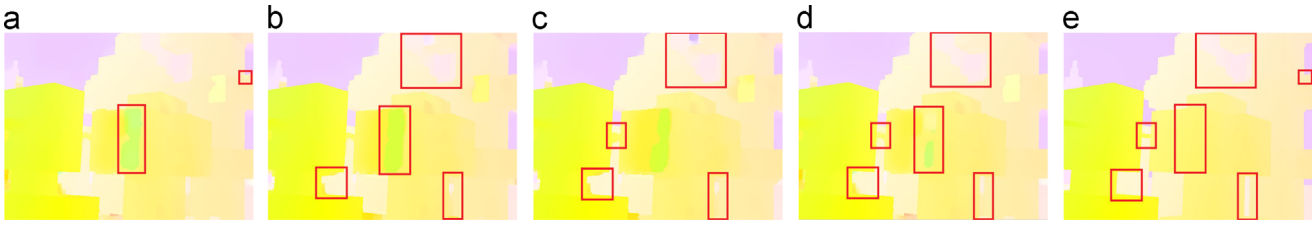

Fig. 3. Visual comparison with four different fusion methods on an Urban3 [40] sample frame (no λ fusion and no CWMF operations). (a)Raket[4], (b)QPBO, (c)LIF, (d)WLIF, and (e)GT.

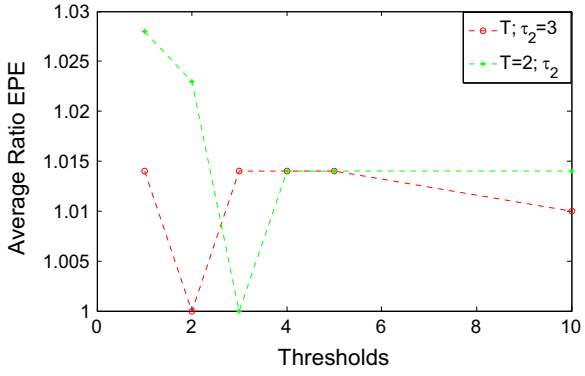

Fig. 4. Optical flow accuracy on the training set of the Middlebury benchmark [40], measured as the average ratio of the EPE using different thresholds τ_1 ($\tau_1=1/T$) and τ_2.

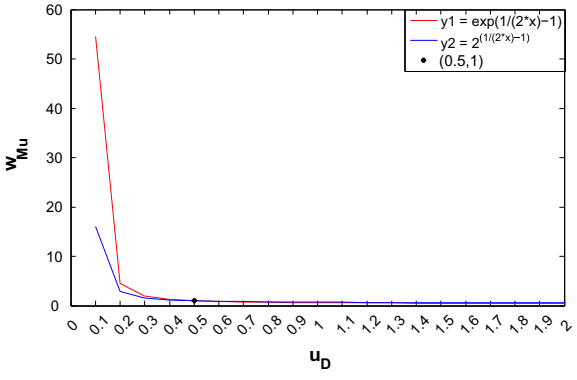

Fig. 5. Decreasing weight functions with different bases.

To correct the errors of the color weight, we modify the original formulation with a two-step strategy:

(1) *Labeling the problematic pixels*: For some motion vectors in the filtering window, the color similarity is inconsistent with the motion similarity. We label a pixel as problematic ($\mathbf{x}_l$) if it meets the following conditions:

$$u_D \le \tau_1 \vee u_D < \frac{\text{mean}(N(u_D))}{\tau_2} \vee I_D > \tau_2 \cdot \text{mean}(N(I_D)) \tag{11}$$

where $u_D = |u-u'|$ is the motion difference between the central pixel (x,y) and its neighbor (x',y'). $I_D = |I(x,y)-I(x',y')|$ is the corresponding color-value difference. N denotes the local filtering neighborhood. $u_D \le \tau_1$ is the essential condition of Eq. (11). τ_1 and τ_2 are thresholds, we set $\tau_1 = 0.5$ based on the notion that if the motion difference is smaller than 0.5 (half a pixel), their motion is similar. We empirically set $\tau_2 = 3$. Fig. 4 shows the accuracy of the baseline method Classic+NL [14] using the proposed CWMF with a fixed parameter set and varying thresholds τ_1 and τ_2.

(2) *Correcting the color weight of the labeled pixels*: The color weight of the labeled pixels is related to their motion similarity, which implies that motion information might be used to modify the incorrect components of the color weight. Essentially, the color weight relies on two elements: σ_l and the color-value distance. Since the color-value distance is fixed, improving σ_l is the only valid way to correct the color weight. We introduce motion information to adaptively change σ_l to correct the color weight. A weight w_{Mu} in terms of the motion difference u_D is added to σ_l for the labeled problematic pixels, and it is expressed as

$$w_{\text{Mu}}(\mathbf{x}) = \begin{cases} 2^{(1/2\cdot u_D(\mathbf{x}_l)-1)}, & \mathbf{x}\in\mathbf{x}_l \\ 1 & \text{otherwise} \end{cases} \tag{12}$$

For one $\mathbf{x}_l$ in a filtering window, the monotonic decreasing function w_{Mu} gives extra weight to its corresponding σ_l adaptively (see Fig. 5). To

Fig. 6. Median filtering comparison. (a) Result of the WMF. (b) Result of our CWMF. (c) GT. (For interpretation of the references to color in the text, the reader is referred to the web version of this paper.)

avoid w_{Mu} to go to infinity when u_{D} equals 0, we set a constraint to u_{D}, where $u_{\mathrm{D}} = \max(u_{\mathrm{D}}, \tau)$. The threshold τ is empirically set to 0.005.

The corrected color weight standard deviation is defined as

$$\sigma_{AI} = \sigma_I \cdot w_{\mathrm{Mu}} \tag{13}$$

The corrected color weight (CCW) is defined as

$$w_{\mathrm{AI}}(x, y, x', y') = \exp\left(-\frac{|I(x,y) - I(x',y')|^2}{2\sigma_{AI}^2}\right) \tag{14}$$

Due to the contribution of σ_{AI}, the weight of the pixel with a large color difference and low motion difference can be appropriately improved compared to [14] (see Fig. 1). The final weight of CWMF is rewritten as

$$w = w_{\mathrm{P}} \cdot w_{\mathrm{AI}} \cdot w_{\mathrm{O}} \tag{15}$$

Algorithm 1. WLIF-Flow.

Input: Images I_1 and I_2
Output: Flow field (u, v)

Pre-compute a NNF $(\hat{u}, \hat{v})$ from I_1 and I_2
for $l = 1$ to *max_level* **do**
  Compute pyramid images ${}^{l}I_1$ and ${}^{l}I_2$
  Compute pyramid NNF $({}^{l}\hat{u}, {}^{l}\hat{v})$
end for

for $l = L$ to 1 (set $L =$ *max_level*) **do**
  /*__Initialization__*/
  (1) *Initialize the continuous flow field:*
  if $l = L$
    $({}^{l}u, {}^{l}v) = 0$
  else
    $({}^{l}u, {}^{l}v) = resample({}^{l+1}u, {}^{l+1}v)$
  end if
  (2) *Refine the continuous flow field with NNF by fusion:*
  $({}^{l}u, {}^{l}v) =$ **WLIF**$\{({}^{l}u, {}^{l}v), ({}^{l}\hat{u}, {}^{l}\hat{v})\}$

  /*__Flowcomputationwithmethod__*/[37]
  for $k = 1$ to *warp_iteration* **do**
    Initialize $(du, dv)^{l,k} = 0$
    Compute $(du, dv)^{l,k}$ by solving Eq. 4)
    Update: $(u, v)_0^{l,k} = (u, v)^{l,k-1} + (du, dv)^{l,k}$
    CWMF $(u, v)_0^{l,k}$
    Recompute $(du, dv)^{l,k}$ after **CWMF**,
    where $(du, dv)_{new}^{l,k} = (u, v)_0^{l,k} - (u, v)^{l,k-1}$
    Reupdate: $(u, v)^{l,k} = (u, v)^{l,k-1} + (du, dv)_{new}^{l,k}$
    end for
end for

4. Experiments

To evaluate the performance of our proposed WLIF-Flow algorithm, we perform experiments on two public benchmarks datasets: Middlebury [40] and MPI-Sintel [41]. In addition, we demonstrate our approach on challenging sequences [42] to evaluate the performance in controlling the balance between the data and smoothness terms in the flow estimation when handling both large and small displacements.

For sequences where ground truth (GT) is available, we present quantitative comparisons with state-of-the-art algorithms in terms of two error measures: average angular error (AAE, [43]) and average endpoint error (EPE, [44]). All experiments are run in MATLAB on a laptop with an Intel Core i5-2410M 2.30 GHz processor and 4 GB memory. Our WLIF-Flow implementation is publicly available.[1] Algorithm 1 outlines the general framework of this method (without λ fusion). In our current CPU implementation, the whole program takes 380 s to compute a high quality flow field for an image pair with resolution 640×480, for instance, the Urban sequence.

4.1. Evaluation of the WLIF approach

We first test the fusion performance of our WLIF approach on combining flow candidates from different methods. We compare our WLIF approach with 3 other fusion approaches: the LIF, the QPBO [35] and the approach of [4]. Table 2 shows the AAE and EPE achieved by the fusion methods on the Middlebury training set. From the last column, we see that our WLIF approach yields better results overall, which means our WLIF approach incorporates flow candidates more effectively. This is further demonstrated in Fig. 3. By comparing the labeled areas of the estimated fields, it becomes clear that our flow field approximates the GT better. Some small motion details that are missed with other approaches are recovered with the proposed WLIF method. Additionally, discontinuities are better preserved in our flow field.

We also test the fusion performance of our WLIF approach on combining flow candidates from different smoothness parameter settings. The quantitative comparison on Middlebury training sequences with λ settings of [0.5, 1, 3] is shown in Table 3. The error statistics display that fusion improves the accuracy of the estimated flow fields. Our WLIF approach outperforms other approaches. We show the benefits of our WLIF approach to fuse flows of different smoothness in Fig. 2. If the selected λ is not suitable or no fusion is performed, the estimated flow field is inaccurate. For example, the motion of the small front palm of the frog is not recovered in Fig. 2(a), and is over-smoothed in Fig. 2(c). With the application of our WLIF approach, these two problems are overcome (Fig. 2(b)). Fig. 2(d) reveals that at edge and texture areas, the flow candidates of small λ are chosen.

Table 4 shows the advantage of our CWMF compared to WMF [14] by evaluating the sequences from the Middlebury benchmark quantitatively. Changing the base of the decreasing weight exponential function (see Fig. 5) does not produce large differences. The weight function $2^{(1/2x-1)}$ performs slightly better, because the variable x changes relatively smoothly when it is smaller than 0.5.

[1] www.projects.science.uu.nl/opticalflow/WLIF-Flow/

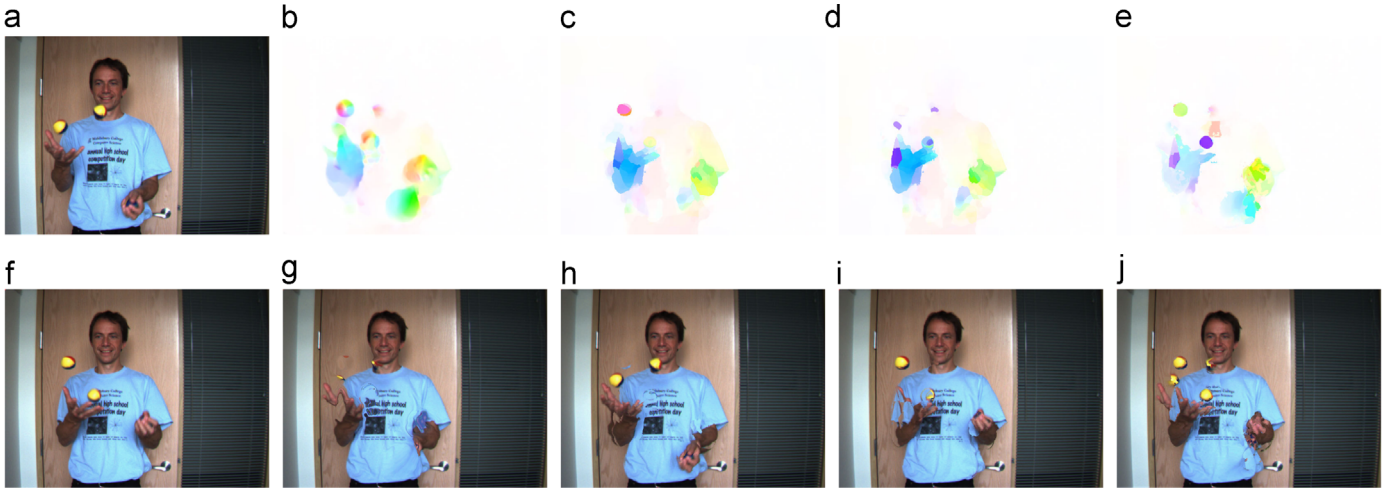

Fig. 7. Evaluation of the Beanbags sequence [40] including small objects with large displacements. (a) Frame 11. (b)–(e) Estimated flow fields. (f) Frame 09. (g)–(j) The corresponding backward warping results based on the flow estimates in (b)–(e), respectively.

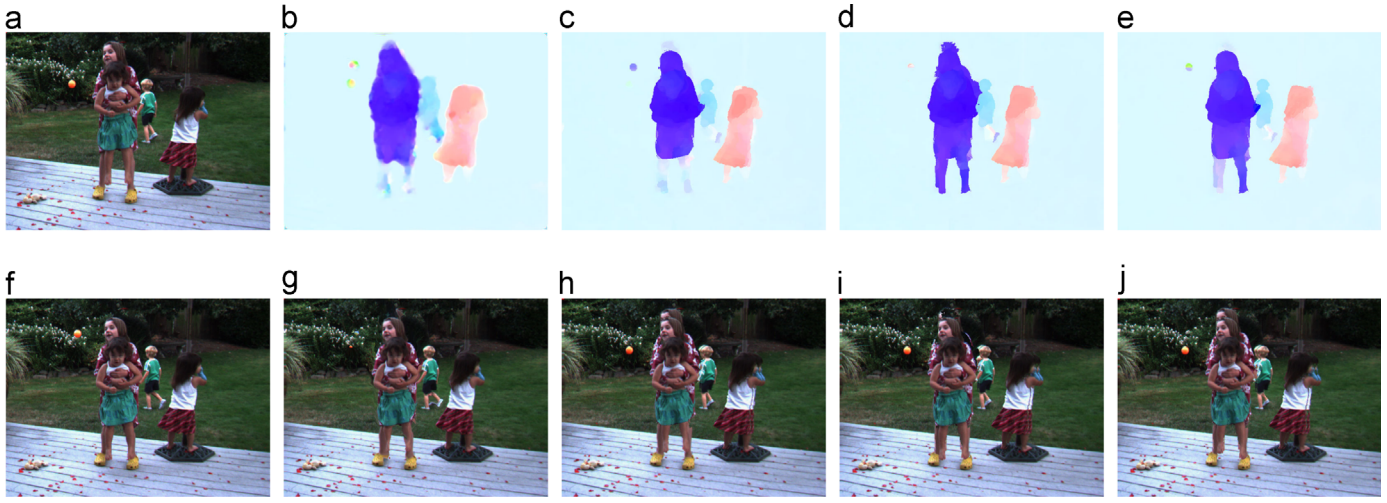

Fig. 8. Evaluation of the Backyard sequence [40] including small objects with large displacements and occlusions. (a) Frame 11. (b)–(e) Estimated flow fields. (f) Frame 09. (g)–(j) The corresponding backward warping results based on the flow estimates in (b)–(e), respectively.

Table 4
AAE/EPE for the tested sequences from the Middlebury benchmark with two different filtering approaches: WMF and CWMF with different monotonic decreasing exponential functions (no fusion operation).

Method	Urban2	Urban3	Grove2	Grove3	RubW.	Venus	Dime.	Hydra.	Avg.Ratio
WMF	1.840/0.210	2.445/0.407	1.255/0.090	4.342/0.433	2.159/0.070	3.156/0.226	2.514/0.130	1.896/0.157	1.017/1.026
CWMF(e)	1.844/0.209	2.378/0.391	1.245/0.088	4.397/0.434	2.138/0.067	3.109/0.224	2.497/0.129	1.888/0.155	1.006/1.008
CWMF(2)	1.842/0.208	2.377/0.390	1.243/0.088	4.301/0.427	2.135/0.066	3.109/0.224	2.493/0.127	1.879/0.154	1.000/1.000

From Fig. 1 we can see why the CWMF is superior to the WMF. When the color similarity is inconsistent with the motion similarity, the adaptive standard deviation σ_{AI} (Fig. 1(e)) will automatically correct the color distance weight (Fig. 1(f)) according to amend the standard deviation σ_I (Fig. 1(c)). One edge pixel of RubberWhale at position $(x=216, y=42)$ is selected for analysis. Fig. 1(b) shows its horizontal GT motion difference u_D in its filtering window (5×5). By comparing Fig. 1(b) with Fig. 1(d), it can be observed that the color weight w_I of the WMF contains many errors. For some neighbors, such as $(x'=218, y'=45)$, the motion difference is very small. This means that they have similar motion, and the filtering weight between them should be large. But in fact, their color weight is very small (Fig. 1(d)). This problem is solved in Fig. 1(f), the corrected color weight w_{AI} now accurately reflects the weight relationship between the central pixel and its neighbors. Fig. 6 shows the final filtering results of RubberWhale. CWMF (Fig. 6(b)) corrects some boundary errors produced by the WMF (Fig. 6(a)). For example, the boundaries of the light green semicircle are closer to the GT (Fig. 6(c)) than the WMF.

To evaluate the performance of our WLIF-Flow method on handling large displacements, we validate it from three aspects:

(1) *Large displacement of small objects*: Fig. 7 shows the results of the Beanbags [40] sequence produced by three state-of-the-art large displacement handling methods: LDOF [32], Classic+NLP [45] and NN-field [8]. As shown in Fig. 7(e) and Fig. 7(j), our WLIF-Flow method successfully captures the large displacements of the small, textureless balls. In contrast, LDOF [32] (Fig. 7(b) and (g)) fails to recover the motion of the balls, because the algorithm relies on descriptor matching and it is

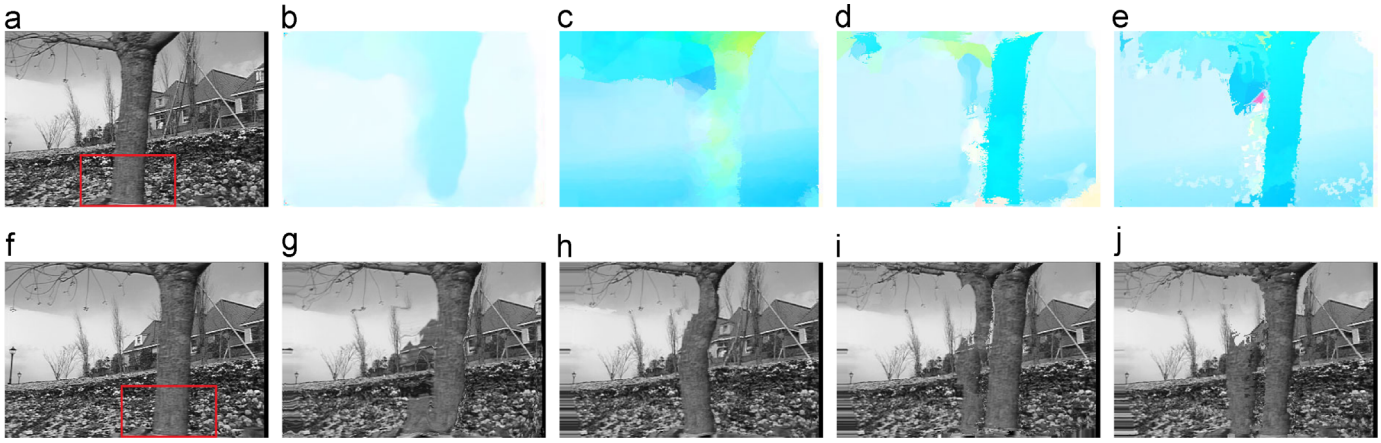

Fig. 9. Evaluation of the FlowerGarden sequence including large objects with large displacements. (a) Frame 11. (b)–(e) Estimated flow fields. (f) Frame 01. (g)–(j) The corresponding backward warping results based on the flow estimates in (b)–(e), respectively.

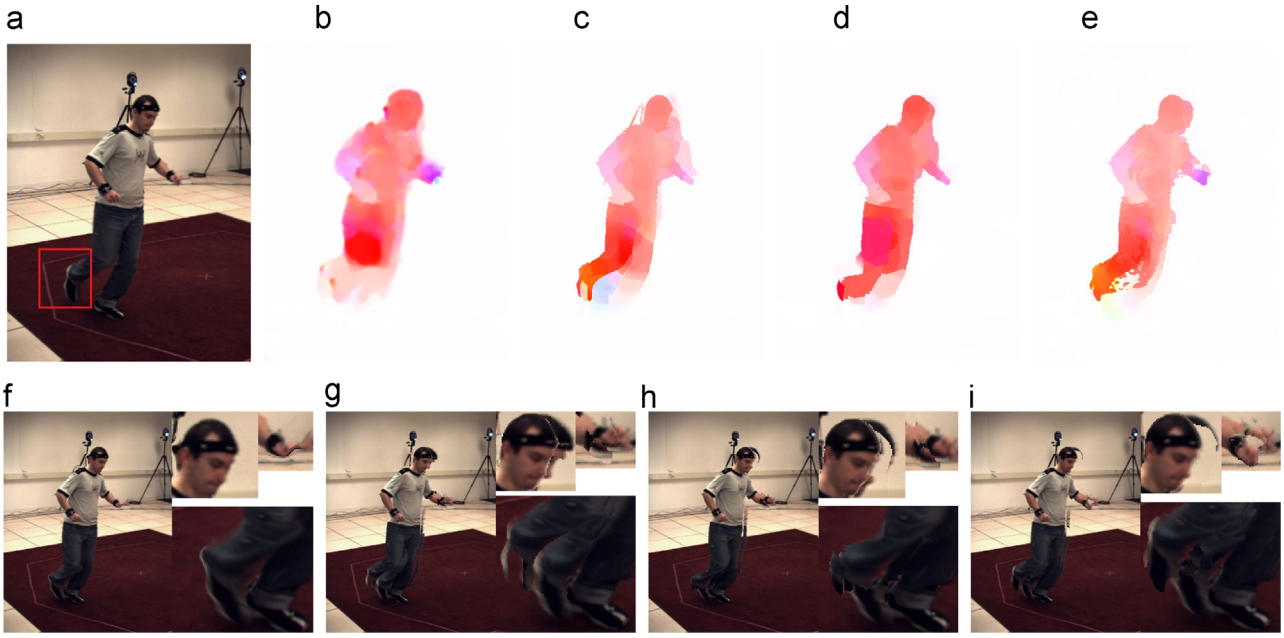

Fig. 10. Visual comparison of an HumanEva-II sequence [42] with challenging situations. (a) Frame 546. Four flow estimations and the corresponding backward warping results are shown in (b)–(i).

hard to extract descriptors of the textureless balls. In Fig. 7(c), the Classic+NLP [45] only recovers the motion of one ball. Moreover, from its corresponding warped image (Fig. 7(h)), we can see that the estimated flow of the balls is wrong: the locations of the recovered balls are quite different from the original input image (Fig. 7(a)). Since the QPBO fusion of the NN-field [8] is inaccurate, the captured motion of the balls is not precise and the motion of the right down ball is too dim to identify (see Fig. 7(d) and (i)).

(2) *Large displacement of small objects and occlusions*: Fig. 8 shows another example of small scale objects with large displacements. Additionally, the frames include serious occlusions. With the application of our WLIF and CWMF approaches, as well as the occlusion-aware refinement strategy, our WLIF-Flow method improves the performance of the NN-field [8]. Not only the large displacements of small objects are well recovered, but also the occlusions are appropriately tackled in Fig. 8(e) and (j). LDOF cannot estimate the motion of the ball, and the motion of the legs of the young girl who is embraced is lost in Classic+NLP (see Fig. 8(h)).

(3) *Large displacement of large objects*: Fig. 9 compares our result with other methods on the FlowerGarden sequence. The LDOF and NN-field fail to capture motions of the tree. In particular, the motion of the stump is lost in LDOF. Furthermore, the edges (e.g. the trunk and the branches) are poorly preserved. In Classic+NLP, due to the intrinsic limitation of the coarse-to-fine strategy, many large displacements are incorrectly estimated. As shown in Fig. 9(c), it is too dim to distinguish whether the captured flow contains a tree. In contrast, by combining the continuous flow field and the discrete NN-field properly in terms of the proposed WLIF, our WLIF-Flow method performs better. Apart from some errors around the branches, large displacements are well estimated, discontinuities are well preserved and the redundant error motion is reduced.

4.2. Evaluation of both large and small displacement handling

Fig. 10 gives an example of the challenge HumanEva-II sequence [42], which contains large displacements of large objects (e.g. right foot), large displacements of small objects (e.g.

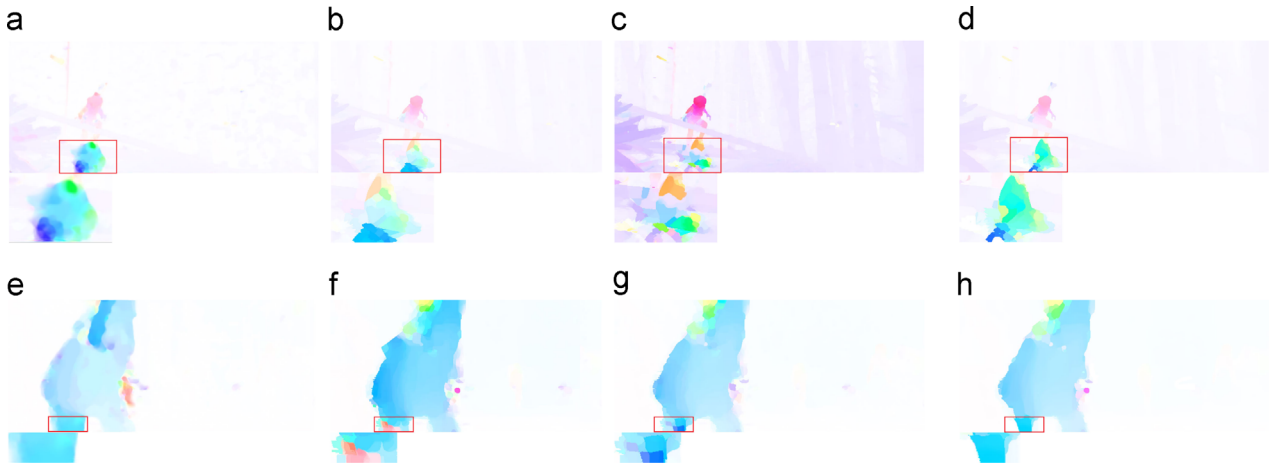

Fig. 11. Two examples on the MPI-Sintel benchmark. Results of training sequences bamboo_2 (top) and market_2 (bottom), frames 10 and 11. From left to right: results of LDOF [32], Classic+NLP [45], NN-field [8] and WLIF.

Table 5
Average end-point error (EPE) on the MPI-Sintel test set.

Method	WLIF-Flow (Ours)	MDP-Flow2 [7]	MLDP-OF	LDOF [32]	Classic+NLP [45]	Classic+NL [14]
EPE(clean)	**5.734**	5.837	7.297	7.563	6.731	7.961
EPE(final)	**8.049**	8.445	8.287	9.116	8.291	9.153

left hand) and motion blur and occlusions. Fig. 10 (b) and (f) shows that the LDOF [32] can capture large displacements of the right foot but fails to capture large displacement of the left hand. The NN-field [8] is poor at recovering large displacements of the left foot and also misses some details of the right hand in Fig. 10 (d) and (h). The Classic+NLP [45] well estimates small displacements (e.g. the right hand), and due to the contribution of the WMF, motion boundaries are well preserved. However, it performs poorly when faced with large displacements. For example, in Fig. 10(g), the warped right foot is far from the ground truth (Fig. 10(a)).

Moreover, two sequences bamboo_2 and market_2 on the MPI-Sintel benchmark [41] are selected to test whether the proposed WLIF-Flow method can both handle large displacements and preserve motion details. Fig. 11 shows that the motion boundaries of LDOF [32] are too blurry to identify the contents. The Classic+NLP [45] is good at computing small displacements. For instance, the motion of the girl is well estimated. However, for large displacements such as the butterfly (Fig. 11(b)) and the foreleg (Fig. 11(f)), its performance degrades. Although the NN-field [8] contains a large displacement handling mechanisms, it also fails to capture the motion of the butterfly (Fig. 11(c)) and the foreleg (Fig. 11(g)) because of the drawbacks of the QPBO fusion. In comparison, by integrating both advantages of the Classic+NLP [45] and the NN-field [8], our results are much more accurate and the recovered motion structures are nearly as the same as the actual objects.

Finally, we evaluate our WLIF-Flow method on both the Middlebury and the MPI-Sintel benchmarks quantitatively. At the time of submission, among the published papers, it is ranked 5th on EPE and 8th on AAE of the Middlebury benchmark, and it is ranked 3th on the clean pass and 5th on the final pass of the MPI-Sintel benchmark. Some methods only perform well on one benchmark, such as the MDP-Flow2 [7] performs top on the Middlebury while poorly on the MPI-Sintel benchmark, and the DeepFlow [9] performs much better on the MPI-Sintel than on the Middlebury benchmark. Our method performs equally well on both of benchmarks. Specifically, our method outperforms related state-of-the-art methods MLDP-OF, Classic+NLP and LDOF on the MPI-Sintel (see Table 5).

5. Conclusion

We have introduced WLIF (weighted local intensity fusion), a method to fuse flow candidates with different algorithms and different smoothness parameter settings. By combining the useful matching information into the variational optimization framework, the motion of structures with displacements larger than their own size can be recovered accurately. Additionally, some motion details which may be ignored in the traditional coarse-to-fine refinement can actually be estimated. Furthermore, the blended flow that comes from different smoothness parameters produced flows achieves an excellent balance between the data term and smoothness terms. To improve the performance on median filtering the intermediate flow field, corrected weighted median filter (CWMF) is proposed. The adaptive color weight standard deviation is employed to correct the errors of WMF. Future work includes improving the matching technique to handle occlusions and to further improve the fusion performance to integrate flow candidates. We are also exploring ways to reduce the computational load of the algorithm.

Conflict of interest

None declared.

References

[1] B. Horn, B. Schunck, Determining optical flow, Artif. Intell. 17 (1–3) (1981) 185–203.
[2] G. Botella, U. Meyer-Baese, A. García, M. Rodríguez-Álvarez, Quantization analysis and enhancement of a VLSI gradient-based motion estimation architecture, Digit. Signal Process. 22 (6) (2012) 1174–1187.
[3] H. Zimmer, A. Bruhn, J. Weickert, Optic flow in harmony, Int. J. Comput. Vis. 93 (3) (2011) 368–388.
[4] L.L. Raket, Local smoothness for global optical flow, in: Proceedings of International Conference on Image Processing, 2012, pp. 1–4.
[5] B. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, in: Proceedings of International Joint Conference on Artificial Intelligence, 198, pp. 1 674–679.

[6] N. Papenberg, A. Bruhn, T. Brox, S. Didas, J. Weickert, Highly accurate optic flow computation with theoretically justified warping, Int. J. Comput. Vis. 67 (2) (2006) 141–158.
[7] L. Xu, J. Jia, Y. Matsushita, Motion detail preserving optical flow estimation, IEEE Trans. Pattern Anal. Mach. Intell. 16 (9) (2012) 1744–1757.
[8] Z. Chen, H. Jin, Z. Lin, S. Cohen, Y. Wu, Large displacement optical flow from nearest neighbor fields, in: Proceeding of Computer Vision and Pattern Recognition, 2013, pp. 2443–2450.
[9] P. Weinzaepfel, J. Revaud, Z. Harchaoui, C. Schmid, Deep flow: large displacement optical flow with deep matching, in: Proceedings of International Conference on Computer Vision, 2013, pp. 1385–1392.
[10] V. Lempitsky, S. Roth, C. Rother, Fusion flow: discrete continuous optimization for optical flow estimation, in: Proceeding of Computer Vision and Pattern Recognition, 2008, pp. 1–8.
[11] Z. Tu, W. Xie, W. Hurst, S. Xiong, Q. Qin, Weighted root mean square approach to select the optimal smoothness parameter of the variational optical flow algorithms, Opt. Eng. 51 (2012) 037202–037209.
[12] A. Wedel, T. Pock, C. Zach, H. Bischof, D. Cremers, An improved algorithm for TV-L1 optical flow computation, in: Proceedings of Dagstuhl Visual Motion Analysis Workshop, 2008.
[13] Z. Tu, C. van Gemeren, R.C. Veltkamp, Improved color patch similarity measure based weighted median filter, in: Proceedings of Asian Conference on Computer Vision, 2015, pp. 1–15.
[14] D. Sun, S. Roth, M. J. Black, Secrets of optical flow estimation and their principles, in: Proceeding of Computer Vision and Pattern Recognition, 2010, pp. 2432–2439.
[15] J. Xiao, H. Cheng, H. Sawhney, C. Rao, M. Isnardi, Bilateral filtering based optical flow estimation with occlusion detection, in: Proceedings of European Conference on Computer Vision, 2006, pp. 211–224.
[16] A. Geiger, P. Lenz, C. Stiller, R. Urtasun, Vision meets robotics: the Kitti dataset, Int. J. Robot. Res. 32 (11) (2013) 1231–1237.
[17] H.A. Rashwan, M.A. Mohamed, M. Angel, Illumination robust optical flow model based on histogram of oriented gradients, Pattern Recognit. 8142 (2013) 354–363.
[18] M.A. Mohamed, H.A. Rashwan, B. Mertsching, M.A. García, D. Puig, Illumination-robust optical flow using a local directional pattern, IEEE Trans. Circuits Syst. Video Technol. 24 (9) (2014) 1499–1508.
[19] M.J. Black, P. Anandan, The robust estimation of multiple motions: parametric and piecewise smooth flow fields, Comput. Vis. Image Understand. 63 (1) (1996) 75–104.
[20] H.A. Rashwan, M.A. García, D. Puig, Variational optical flow estimation based on stick tensor voting, IEEE Trans. Image Process. 22 (7) (2013) 2589–2599.
[21] H. Chang, Y.C.F. Wang, Superpixel-based large displacement optical flow, in: Proceedings of International Conference on Image Processing, 2013, pp. 3835–3839.
[22] M. Hornacek, F. Besse, J. Kautz, A. Fitzgibbon, C. Rother, Highly over-parameterized optical flow using patchmatch belief propagation, in: Proceedings of European Conference on Computer Vision, 2014, pp. 220–234.
[23] J. Diaz, E. Ros, F. Pelayo, E.M. Ortigosa, S. Mota, FPGA-based real-time optical-flow system, IEEE Trans. Circuits Syst. Video Technol. 16 (2) (2006) 274–279.
[24] G. Botella, A. García, M. Rodríguez-Álvarez, E. Ros, U. Meyer-Baese, M. C. Molina, Robust bioinspired architecture for optical-flow computation, IEEE Trans. Very Large Scale Integr. Syst. 18 (4) (2010) 616–629.
[25] K. Pauwels, M. Tomasi, J. Diaz, E. Ros, M.M.V. Hulle, A comparison of FPGA and GPU for real-time phase-based optical flow, stereo, and local image features, IEEE Trans. Comput. 61 (7) (2011) 999–1012.
[26] L. Sevilla-Lara, D. Sun, E.G. Learned-Miller, M.J. Black, Optical flow estimation with channel constancy, in: European Conference on Computer Vision, 2014, pp. 423–438.
[27] L. Alvarez, J. Weickert, J. Sanchez, Reliable estimation of dense optical flow fields with large displacements, Int. J. Comput. Vis. 39 (1) (2000) 41–56.
[28] F. Steinbruecker, T. Pock, D. Cremers, Large displacement optical flow computation without warping, in: Proceedings of International Conference on Computer Vision, 2009, 2009, pp. 1609–1614.
[29] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, M. Gelautz, Fast cost-volume filtering for visual correspondence and beyond, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2) (2013) 504–511.
[30] C. Liu, J. Yuen, A. Torralba, Sift flow: dense correspondence across scenes and its applications, IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2011) 978–994.
[31] D.G. Lowe, Object recognition from local scale-invariant features, in: Proceedings of Computer Vision and Pattern Recognition, 1999, pp. 1150–1157.
[32] T. Brox, J. Malik, Large displacement optical flow: descriptor matching in variational motion estimation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (3) (2011) 500–513.
[33] M. Stoll, S. Volz, A. Bruhn, Adaptive integration of feature matches into variational optical flow methods, in: Proceedings of Asian Conference on Computer Vision, 2012, pp. 1–14.
[34] C. Barnes, E. Shechtman, D.B. Goldman, A. Finkelstein, The generalized patch match correspondence algorithm, in: Proceedings of European Conference on Computer Vision, 2010, pp. 29–43.
[35] P.L. Hammer, P. Hansen, B. Simeone, Roof duality, complementation and persistency in quadratic 0-1 optimization, Math. Program. 28 (1984) 121–155.
[36] C. Rother, V. Kolmogorov, V.S. Lempitsky, M. Szummer, Optimizing binary MRFs via extended roof duality, in: Proceedings of Computer Vision and Pattern Recognition, 2007, pp. 1–8.
[37] Z. Tu, N. van der Aa, C. van Gemeren, R.C. Veltkamp, A combined post-filtering method to improve accuracy of variational optical flow estimation, Pattern Recognit. 47 (5) (2014) 1926–1940.
[38] W. Josh, S. Agarwal, S. Belongie, What went where, in: Proceedings of Computer Vision and Pattern Recognition, 2003, pp. 37–44.
[39] W. Andreas, D. Cremers, T. Pock, H. Bischof, Structure and motion-adaptive regularization for high accuracy optic flow, in: Proceedings of International Conference on Computer Vision, 2009, pp. 1663–1668.
[40] S. Baker, D. Scharstein, J.P. Lewis, S. Roth, M.J. Black, R. Szeliski, A database and evaluation methodology for optical flow, Int. J. Comput. Vis. 92 (1) (2011) 1–31.
[41] D. J. Butler, J. Wul, G.B. Stanley, M.J. Black, A naturalistic open source movie for optical flow evaluation, in: Proceedings of European Conference on Computer Vision, 2012, pp. 611–625.
[42] L. Sigal, A.O. Balan, M.J. Black, Humaneva: synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion, Int. J. Comput. Vis. 87 (1–2) (2010) 4–27.
[43] J. Barron, D. Fleet, S. Beauchemin, Performance of optical flow techniques, Int. J. Comput. Vis. 12 (1) (1994) 43–77.
[44] M. Otte, H. H. Nagel, Optical flow estimation: advances and comparisons, in: Proceedings of European Conference on Computer Vision, 1994, pp. 51–60.
[45] D. Sun, S. Roth, M.J. Black, A quantitative analysis of current practices in optical flow estimation and the principles behind them, Int. J. Comput. Vis. 106 (2) (2014) 115–137.

Zhigang Tu started his M.Phil. Ph.D. in Image Processing in the School of Electronic Information, Wuhan University, China, 2008. Since September 2011, he is with the Multimedia and Geometry Group at Utrecht University, Netherlands. His current research interests include optical flow estimation, super-resolution construction, multimedia systems and technologies and human-computer interaction.

Ronald Poppe received a Ph.D. in Computer Science from the University of Twente, the Netherlands. In 2009, 2010 and 2012, he was a Visiting Researcher at the Delft University of Technology, Stanford University and University of Lancaster, respectively. He is currently an Assistant Professor at the Information and Computing Sciences Department of Utrecht University. His research interests include the analysis of human behavior from videos and other sensors, the understanding and modeling of human (communicative) behavior and the applications of both in real-life settings. In 2012 and 2013, he received the most cited paper award from the "Image and Vision Computing" journal, published by Elsevier.

Remco C. Veltkamp is a Full Professor of Multimedia at Utrecht University, Netherlands. His research interests include the analysis, recognition and retrieval of, and interaction with, music, images, and 3D objects and scenes, in particular the algorithmic and experimentation aspects. He has written over 150 refereed papers in reviewed journals and conferences, and supervised 15 Ph.D. theses. He was the Director of the National Project GATE – Game Research for Training and Entertainment.