Signal Processing 127 (2016) 253–265

Contents lists available at ScienceDirect

Signal Processing

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

Adaptive guided image filter for warping in variational optical flow computation

Zhigang Tu [a,b], Ronald Poppe [a,b], Remco C. Veltkamp [a,b,*]

[a] School of Computing, Informatics, Decision System Engineering, Arizona State University, USA
[b] Department of Information and Computing Sciences, Utrecht University, Utrecht, The Netherlands

ARTICLE INFO

Article history:
Received 22 April 2015
Received in revised form
1 February 2016
Accepted 26 February 2016
Available online 16 March 2016

Keywords:
Variational optical flow
Warped interpolation image correction
Adaptive guided image filter

ABSTRACT

The variational optical flow method is considered to be the standard method to calculate an accurate dense motion field between successive frames. It assumes that the energy function has spatiotemporal continuities and appearance motions are small. However, for real image sequences, the temporal continuity assumption is often violated due to outliers and occlusions, causing inaccurate flow vectors at these regions. After each warping operation, errors are generated at the corresponding regions of the warped interpolation image. This results in an inaccurate discrete approximation of the temporal derivative and thus ends up affecting the accuracy of the estimated flow field. In this paper, we propose an adaptive guided image filter to correct these errors in the warped interpolation image. A guidance image is reconstructed by considering both the feature of the reference image as well as the difference between the warped interpolation image and the reference image, to guide the filtering of the warped interpolation image. To adjust the smoothing degree, the regularization parameter in the guided image filter is adaptively selected based on a confidence measure. Extensive experiments on different datasets and comparison with state-of-the-art variational optical flow algorithms demonstrate the effectiveness of our method.

1. Introduction

Estimation of the apparent motion of a scene plays a fundamental role in computer vision and image processing. Currently, one of the most successful techniques that addresses this problem is the variational optical flow method [1,2], which formulates an energy function by aggregating a data term and a smoothness term. Minimizing the energy function can get the motion of each pixel. Technically speaking, the minimization involves solving the corresponding Euler–Lagrange equations of the energy function numerically [3]. The calculated flow field is a dense field of displacement vectors, which describes how corresponding pixels in consecutive frames match. The direction and magnitude of each flow vector indicate where and how far a pixel moved. This property ensures the optical flow method is widely used in, for example, object tracking [4] and segmentation [5], super-resolution reconstruction [6], and medical diagnostics [7].

The variational optical flow method proposed by Horn and Schunck (HS) [8] is based on two assumptions: 1) the *brightness constancy assumption (BCA)*, which assumes that the brightness of a pixel does not change along its motion trajectory over time; and 2) the *smoothness assumption*, which assumes that the flow field is smooth and ensures the optical flow problem is well posed. In practice, these two assumptions are rarely satisfied. Especially under challenging conditions, such as outliers (e.g., image noise and estimated flow errors [9]), large displacements, occlusions and illumination changes, the performance of HS is notably worse [40]. Furthermore, the HS model is too slow to be implemented in real-time as the large and sparse partial differential equations (PDEs) is computationally expensive. Fortunately, in the past 30 years, considerable progress has been achieved. The progress can be divided into two classes: 1) different advanced concepts are integrated into the variational framework, to enable the methods to preserve discontinuity [5,10,11], to handle large displacements [1,2,20], to treat illumination changes [12,13], to be robust with respect to outliers [9,15], and to tackle occlusions [9,16,17]; and 2) efficient and advanced optimization schemes are presented to target real-time use [3,18].

The popular variational algorithms are focused on how to improve the data term (Eq. (1)) by incorporating new assumptions [19], or to design advanced smoothness constraints that preserve discontinuity. There are few literatures concern the basic feature of the numerical computation, where the flow field is computed from

* Corresponding author at: School of Computing, Informatics, Decision System Engineering, Arizona State University, USA.
E-mail addresses: Zhigang.Tu@asu.edu (Z. Tu), R.W.Poppe@uu.nl (R. Poppe), R.C.Veltkamp@uu.nl (R.C. Veltkamp).

http://dx.doi.org/10.1016/j.sigpro.2016.02.018
0165-1684/

the spatiotemporal derivatives of image intensities. Therefore, the input images and the intermediate warped interpolation image $\mathbf{I}_{\text{warp}}$ ($\mathbf{I}_{\text{warp}}$ refers to the warped interpolation image based on the flow field calculated in the previous step, see Eq. (2)) are constrained to be continuous and differentiable in space-time. Simultaneously, the flow vector should be small and valid at every pixel. Errors are typically produced in the intermediate flow fields due to outliers and occlusions. Subsequently, $\mathbf{I}_{\text{warp}}$ will contain errors caused by these inaccurate flow vectors. Furthermore, the spatiotemporal derivatives will be distorted. Consequently, the accuracy of the final estimated flow field will be degraded. Correcting errors of the intermediate flow fields and the $\mathbf{I}_{\text{warp}}$ during optimization is a good way to improve the performance of the variational algorithms.

Till now, the majority of variational optical flow algorithms try to reduce errors during the numerical computation, concentrating on applying different filters to smooth the intermediate flow fields to remove outliers or to correct flow errors by integrating useful information [10,21].

Xiao et al. [17] proposed a multi-cue driven adaptive bilateral filter (BF), which enables smoothing of the flow field with highly desirable motion discontinuity preservation. The algorithm can handle occlusions partially, but, BF is time consuming. Wedel et al. [12] introduced a median filter (MF) to denoise the flow field, but the MF over-smooths motion boundaries of the flow field. Sun et al. [10] proposed a modified weighted median filter (WMF) to prevent this kind of over-smoothing, but the weight is easily influenced by textured, noisy pixels and illumination changes. Recently, Rashwan et al. [22] improved the weight by using the saliency of image gradients to replace the intensity-based measure. Tu et al. [16] proposed a novel combined post-filtering (CPF) method to efficiently remove outliers and handle occlusions simultaneously.

Brox et al. [3] proposed a two nested fixed point iterations based numerical scheme to combine with the coarse-to-fine strategy to efficiently solve the variational methods. As pointed out in [1,2], this method fails to recover many motion details. Because some fine structures are smoothed out at coarse levels and then "forgotten", therefore, they cannot be correctly estimated at final scale. Xu et al. [2] proposed an extended coarse-to-fine (EC2F) refinement framework to reduce the reliance of flow estimates on their initial values propagated from the coarse level. By integrating available matching information into the continuous flow at each scale, the lost motion of fine structures can be recovered.

These methods only consider filtering the intermediate flow fields or refining the initial flow vectors, they neglect to correct the basic elements of the PDEs – the spatiotemporal derivatives.

The correctness of the temporal derivative $\mathbf{I}_t$ completely depends on $\mathbf{I}_{\text{warp}}$, which means that $\mathbf{I}_{\text{warp}}$ heavily influences the accuracy of the estimated flow field. Currently, nearly all variational methods (e.g. [10,12,22]) simply use the temporal derivative in the numerical iteration. In contrast, we propose an adaptive guided image filter (AGIF) to filer $\mathbf{I}_{\text{warp}}$ before computing the temporal derivative. The AGIF technique is especially useful to correct errors of $\mathbf{I}_{\text{warp}}$ which are caused by outliers and occlusions.

The guided image filter (GIF), which was proposed recently by He et al. [23], smooths the input image by considering the content of a guidance image that can be another image or the input image itself. The GIF is similar to the joint bilateral filter (JBF) [24], which is able to reduce noise while preserve edges. However, the JBF is nonlinear and computationally complex. The computational time of the JBF increases exponentially as the size of the filtering window increases. In contrast, the GIF is implemented as a sequence of box filters, making it linear and efficient. More significantly, its runtime is independent of the filtering window size. Due to these advantages, the GIF is popular in denoising [23,25], sharpness enhancement [26], optical flow computation [28] and stereo matching [27,28]. For example, Xiao et al. [39] applied the GIF to filter the three dimensional cost-volume to preserve edge information and improve operational efficiency. To treat with the problem of flickering-artifacts that is caused by the incoherent disparity maps, Liu et al. [41] proposed a novel temporal consistency enhancement algorithm based on Guided Filter and Temporal Gradient to smooth disparity sequences to improve the consistency of the sequence.

In this work, we present an AGIF technique to smooth $\mathbf{I}_{\text{warp}}$ to improve the performance of the variational optical flow method. To analyze the characteristic of the GIF as well as the relationship between the reference image and $\mathbf{I}_{\text{warp}}$, we reconstruct a guidance image which is a combination of both. The combination is based on a confidence measure which originates from the temporal derivative, as the temporal derivative can be interpreted as an indication of mismatch. Due to this contribution, the filtered $\mathbf{I}_{\text{warp}}$ can gain useful information from the reference image. Except the general edges, we can correct some intensities which are distorted due to outliers and occlusions of the before filtered $\mathbf{I}_{\text{warp}}$. Another contribution is that, we propose a method to select the optimal regularization parameter ε (Eq. (18)) adaptively based on two principles – the error degree of $\mathbf{I}_{\text{warp}}$ and the size of the input testing images. This method effectively improves the smoothing ability of the GIF.

The paper is organized as follows. Section 2 describes a fundamental CPF method and the numerical optimization process. In Section 3, we propose an AGIF technique to correct errors of the warped interpolation image. Experiments are presented in Section 4. Finally, discussion and conclusions are given in Section 5.

2. Variational optical flow model and minimization

2.1. The combined Post-filtering (CPF) method

Let $\mathbf{I}_1, \mathbf{I}_2$: ($\Omega \subset R^2$) be the two consecutive frames at time t and $t+1$. $\mathbf{x}=(x, y, t)^{\mathrm{T}}$ denotes the locations of pixels in the spatial image domain Ω, and $\mathbf{w}=(\mathbf{u}, \mathbf{v})^{\mathrm{T}}$ is the flow field describes the displacement in x-and y-direction between $\mathbf{I}_1$ and $\mathbf{I}_2$. The common assumption of flow estimation is the *brightness constancy assumption* (BCA). Based on the BCA, a data term is formed:

$$E_{\mathrm{D}}(\mathbf{w}) = \int_{\Omega} \left|\mathbf{I}_2(\mathbf{x}+\mathbf{w}) - \mathbf{I}_1(\mathbf{x})\right|^2 d\mathbf{x} \tag{1}$$

where

$$\mathbf{I}_{\text{warp}} = \mathbf{I}_2(\mathbf{x}+\mathbf{w}) \tag{2}$$

is a warped interpolation image with certain interpolation method such as, cubic interpolation, bilinear interpolation [15], or bicubic interpolation [12].

The two unknowns ($\mathbf{u}$, $\mathbf{v}$) be determined from one single Eq. (1). To solve this aperture problem, a global smoothness assumption has been introduced, where a smoothness term can be expressed as [8]:

$$E_{\mathrm{S}}(\mathbf{w}) = \int_{\Omega} |\nabla \mathbf{u}|^2 + |\nabla \mathbf{v}|^2 d\mathbf{x} \tag{3}$$

By integrating the smoothness term Eq. (3) into Eq. (1), and by using a parameter λ to steer the relative importance of the two terms, the optical flow estimation can be formulated as an energy minimization problem in which the energy function is:

$$E(\mathbf{w}) = \int_{\Omega} \underbrace{\left|\mathbf{I}_2(\mathbf{x}+\mathbf{w}) - \mathbf{I}_1(\mathbf{x})\right|^2}_{\text{Data term}} + \lambda \underbrace{\left(|\nabla \mathbf{u}|^2 + |\nabla \mathbf{v}|^2\right)}_{\text{Smoothness term}} d\mathbf{x} \tag{4}$$

This energy function performs poorly when outliers and occlusions occur. A variety of robust penalties [9,10,29] were presented to improve the performance. Recently, Sun et al. [10] pointed out that the L1-norm approximate generalized Charbonnier penalty [30] performs best among them. A generalized Charbonnier penalty based TV-L1 optical flow energy function is defined as:

$$E(\mathbf{w}) = \int_{\Omega} \underbrace{\Psi_D\left(\left|\mathbf{I}_2(\mathbf{x}+\mathbf{w}) - \mathbf{I}_1(\mathbf{x})\right|^2\right)}_{\text{Data term}} + \lambda \underbrace{\Psi_S\left(|\nabla \mathbf{u}|^2 + |\nabla \mathbf{v}|^2\right)}_{\text{Smoothness term}} d\mathbf{x} \tag{5}$$

where $\Psi_D = \Psi_S = \Psi(s^2) = (s^2 + \varsigma^2)^{\alpha}$ ($\alpha = 0.45$), and ς is a small parameter to make sure the energy function differentiable ($\varsigma = 0.001$).

The accuracy can be improved when a MF is applied to smooth the intermediate flow fields during optimization. In contrast to [10], which only applied a WMF for denoising, we adopt a more advanced Classic+CPF method [16] for smoothing. The CPF method is a combined filtering technique, which uses a WMF, a BF and a fast MF to post-smooth the detected edges, occlusions, and the flat regions of the flow field respectively. The CPF based TV-L1 optical flow energy function is expressed as:

$$\begin{aligned} E(\mathbf{u},\mathbf{v},\bar{\mathbf{u}},\bar{\mathbf{v}}) = & \sum_{x,y\in\Omega} \Psi_D\left(\left|\mathbf{I}_2(x+u_{x,y}, y+v_{x,y}, t+1) - \mathbf{I}_1(x,y,t)\right|^2\right) \\ & + \lambda \cdot \Psi_S\left(\left|\nabla u_{x,y}\right|^2 + \left|\nabla v_{x,y}\right|^2\right) + \lambda_2\left(\|\mathbf{u}-\bar{\mathbf{u}}\|^2 + \|\mathbf{v}-\bar{\mathbf{v}}\|^2\right) \\ & + \Bigg(\underbrace{\sum_{x_E,y_E} \sum_{xE',yE'\in N_{x_E,y_E}} w_{x_E,y_E,x_E',y_E'} \left(\left|\bar{u}_{x,y} - \bar{u}_{x',y'}\right| + \left|\bar{v}_{x,y} - \bar{v}_{x',y'}\right|\right)_{x_E,y_E,x_E',y_E'}}_{\text{Edge Regions}\rightarrow\text{Weighted Median Filter}} \\ & + \underbrace{\sum_{x_O,y_O} \sum_{xO',yO'\in N_{x_O,y_O}} w_{x_O,y_O,xO',y_O'} \left(\left|\bar{u}_{x,y} - \bar{u}_{x',y'}\right| + \left|\bar{v}_{x,y} - \bar{v}_{x',y'}\right|\right)_{x_O,y_O,x_O',y_O'}}_{\text{Occlusons}\rightarrow\text{Bilateral Filter}} \\ & + \underbrace{\sum_{x_F,y_F} \sum_{xF',yF'\in N_{x_F,y_F}} \text{median}\left(\left|\bar{u}_{x,y} - \bar{u}_{x',y'}\right| + \left|\bar{v}_{x,y} - \bar{v}_{x',y'}\right|\right)_{x_F,y_F,x_F',y_F'}}_{\text{Flat Regions}\rightarrow\text{Median Filter}} \Bigg) \end{aligned} \tag{6}$$

where λ and λ_2 are the weight parameters that control the balance of each term. $u_{x,y}$ and $v_{x,y}$ are the elements of $\mathbf{u}$ and $\mathbf{v}$ respectively. $\bar{\mathbf{u}}$ and $\bar{\mathbf{v}}$ are the auxiliary flow fields of $\mathbf{u}$ and $\mathbf{v}$, and approximate $\mathbf{u}$ and $\mathbf{v}$. $N_{x,y}$ represents the neighborhood of pixel (x, y). The second and the third terms are weighted non-local terms, which conduct a WMF smoothing and a BF smoothing within a specified region of $(\bar{\mathbf{u}}, \bar{\mathbf{v}})$ respectively. $w_{x_E,y_E,x_E',y_E'}$ is the weight of WMF, which is calculated according to Eq. (9) in [10], and $w_{x_O,y_O,xO',y_O'}$ is the weight of BF, which is calculated by using Eq. (3) in [16]. The fourth term conducts a fast median filter (refer to Eq. (2) in [16] for detailed information).

Different from [16], we use the faster Sobel detector to extract the flow field edges, and set standard deviations of the occlusion state function in weights $w_{x_E,y_E,x_E',y_E'}$ and $w_{x_O,y_O,x_O',y_O'}$ to $\sigma_d = 0.75$ and $\sigma_e = 10$. (Table 1 shows the influence of σ_d and σ_e).

2.2. Minimization

To optimize Eq. (6), we apply an alternating optimization strategy [10] combined with a coarse-to-fine warping scheme [3]. In each pyramid level, at every warping step, two operations are implemented: a well-known fixed point iteration scheme is employed to solve Eq. (5) to get the flow field $(\mathbf{u}, \mathbf{v})$, and a CPF method is applied to $(\mathbf{u}, \mathbf{v})$ to remove outliers.

The fixed point iteration scheme is one of the most successful techniques to solve a large system of PDEs. According to the calculus of variations, a minimizer of Eq. (5) must fulfill the Euler–Lagrange equations:

$$\begin{cases} \Psi_D'\left(\mathbf{I}_t^2\right)\cdot\mathbf{I}_x\mathbf{I}_t - \lambda\cdot div\left(\Psi_S'\left(|\nabla\mathbf{u}|^2 + |\nabla\mathbf{v}|^2\right)\nabla\mathbf{u}\right) = 0 \\ \Psi_D'\left(\mathbf{I}_t^2\right)\cdot\mathbf{I}_y\mathbf{I}_t - \lambda\cdot div\left(\Psi_S'\left(|\nabla\mathbf{u}|^2 + |\nabla\mathbf{v}|^2\right)\nabla\mathbf{v}\right) = 0 \end{cases} \tag{7}$$

where

$$\begin{cases} \mathbf{I}_x = \partial_x\mathbf{I}_2(\mathbf{x}+\mathbf{w}) \\ \mathbf{I}_y = \partial_y\mathbf{I}_2(\mathbf{x}+\mathbf{w}) \\ \mathbf{I}_t = \mathbf{I}_2(\mathbf{x}+\mathbf{w}) - \mathbf{I}_1(\mathbf{x}) \end{cases} \tag{8}$$

At each pyramid level l ($l \geq 1$), the flow field is updated according to $(\mathbf{u}^{l+1}, \mathbf{v}^{l+1}) = (\mathbf{u}^l + d\mathbf{u}^l, \mathbf{v}^l + d\mathbf{v}^l)$, where $(\mathbf{u}^l, \mathbf{v}^l)$ is a known part from coarse levels and $(d\mathbf{u}^l, d\mathbf{v}^l)$ is an unknown update part that needs to be computed:

$$\begin{cases} \Psi_D'\left(\mathbf{I}_t^{l,k+1}\right)^2\cdot\mathbf{I}_x^{l,k}\mathbf{I}_t^{l,k+1} - \lambda\cdot div\left(\Psi_S'\left(\left|\nabla\mathbf{u}^{l,k+1}\right|^2 + \left|\nabla\mathbf{v}^{l,k+1}\right|^2\right)\nabla\mathbf{u}^{l,k+1}\right) = 0 \\ \Psi_D'\left(\mathbf{I}_t^{l,k+1}\right)^2\cdot\mathbf{I}_y^{l,k}\mathbf{I}_t^{l,k+1} - \lambda\cdot div\left(\Psi_S'\left(\left|\nabla\mathbf{u}^{l,k+1}\right|^2 + \left|\nabla\mathbf{v}^{l,k+1}\right|^2\right)\nabla\mathbf{v}^{l,k+1}\right) = 0 \end{cases} \tag{9}$$

with the abbreviations:

$$\begin{cases} \left(\Psi_D'\right)^{l,k} = \Psi_D'\left(\left(\mathbf{I}_x^{l,k}d\mathbf{u}^{l,k} + \mathbf{I}_y^{l,k}d\mathbf{v}^{l,k} + \mathbf{I}_t^{l,k}\right)^2\right) \\ \left(\Psi_S'\right)^{l,k} = \Psi_S'\left(\left|\nabla\left(\mathbf{u}^{l,k} + d\mathbf{u}^{l,k}\right)\right|^2 + \left|\nabla\left(\mathbf{v}^{l,k} + d\mathbf{v}^{l,k}\right)\right|^2\right) \end{cases} \tag{10}$$

where k is the inner iteration step. To solve the equations, the nonlinear terms of the form $\mathbf{I}_2(\mathbf{x}+\mathbf{w}^{l+1})$ ($\mathbf{w}^{l+1} = (\mathbf{u}^{l+1}, \mathbf{v}^{l+1})$) need to be linearized via Taylor expansion:

$$\begin{aligned} \mathbf{I}_t^{l+1} &= \mathbf{I}_2\left(\mathbf{x}+\mathbf{w}^{l+1}\right) - \mathbf{I}_1(\mathbf{x}) \\ &\approx \left(\mathbf{I}_2\left(\mathbf{x}+\mathbf{w}^l\right) - \mathbf{I}_1(\mathbf{x})\right) + \mathbf{I}_x^l d\mathbf{u}^l + \mathbf{I}_y^l d\mathbf{v}^l = \mathbf{I}_t^l + \mathbf{I}_x^l d\mathbf{u}^l + \mathbf{I}_y^l d\mathbf{v}^l \end{aligned} \tag{11}$$

The incremental flow $(d\mathbf{u}^{l,k}, d\mathbf{v}^{l,k})$ is solved by the inner fixed point iteration and is refined to $(d\mathbf{u}^{l,k+1}, d\mathbf{v}^{l,k+1})$ iteratively. According to the fixed point iteration scheme, a linear system of PDEs about the unknown incremental flow $(d\mathbf{u}^{l,k+1}, d\mathbf{v}^{l,k+1})$ is formed:

Table 1
Results (AAE/EPE) of RubberWhale, Venus and Urban3 from the Middlebury dataset with different σ_d and σ_e.

(σ_d, σ_e)	(0.30, 20)	(0.30, 10)	(0.5, 10)	(0.75, 15)	(0.75, 10)
RubberWhale	2.240/0.070	2.200/0.070	2.181/0.069	2.205/0.070	2.163/0.069
Venus	3.070/0.223	3.023/0.222	3.025/0.220	3.053/0.223	3.025/0.221
Urban3	2.569/0.370	2.459/0.359	2.460/0.359	2.572/0.366	2.453/0.353
Average	2.626/0.221	2.556/0.218	2.555/0.217	2.610/0.220	**2.548/0.215**

$$\begin{cases} \left(\Psi'_{\mathrm{D}}\right)^{l,k}\cdot\left(\mathbf{I}_x^{l,k}\left(\mathbf{I}_t^{l,k}+\mathbf{I}_x^{l,k}d\mathbf{u}^{l,k+1}+\mathbf{I}_y^{l,k}d\mathbf{v}^{l,k+1}\right)\right)-\lambda\cdot div\left(\left(\Psi'_{\mathrm{S}}\right)^{l,k}\nabla\left(\mathbf{u}^{l,k}+d\mathbf{u}^{l,k+1}\right)\right)=0 \\ \left(\Psi'_{\mathrm{D}}\right)^{l,k}\cdot\left(\mathbf{I}_y^{l,k}\left(\mathbf{I}_t^{l,k}+\mathbf{I}_x^{l,k}d\mathbf{u}^{l,k+1}+\mathbf{I}_y^{l,k}d\mathbf{v}^{l,k+1}\right)\right)-\lambda\cdot div\left(\left(\Psi'_{\mathrm{S}}\right)^{l,k}\nabla\left(\mathbf{v}^{l,k}+d\mathbf{v}^{l,k+1}\right)\right)=0 \end{cases} \tag{12}$$

We set $d\mathbf{u}^{l,0}=d\mathbf{v}^{l,0}=0$ for initialization at each pyramid level l. $\mathrm{I}_x^{l,k}$, $\mathrm{I}_y^{l,k}$and $\mathrm{I}_t^{l,k}$are the intermediate spatio-temporal derivatives. They are usually computed as follows: 1) for the spatial derivatives $\mathrm{I}_x^{l,k}$and $\mathrm{I}_y^{l,k}$, we first calculate the x-and y-direction spatial derivatives of $\mathbf{I}_2$ with 5-point derivative filter $\frac{1}{12}[-1, 8, 0, -8, 1]$, and then warp the spatial derivatives to $\mathbf{I}_1$ with the current estimated flow field with respect to the bicubic interpolation [12]; 2) for the temporal derivative $\mathrm{I}_t^{l,k}$, it is a difference between the warped interpolation image $\mathbf{I}_{\mathrm{warp}}$ and the reference image $\mathbf{I}_1$. After computing the spatial and temporal derivatives, the classic successive over-relaxation (*SOR*) method [31] is used to solve Eq. (12) to obtain $(d\mathbf{u}^{l,k+1}, d\mathbf{v}^{l,k+1})$.

At the second step, the CPF method is applied to smooth the newly updated flow field $(\mathbf{u}^{l}+d\mathbf{u}^{l,k+1}, \mathbf{v}^{l}+d\mathbf{v}^{l,k+1})$, which can efficiently reduce outliers as well as handle occlusions properly.

One significant problem during the numerical calculation is how to process the spatial and temporal derivatives$\mathrm{I}_x^{l,k}$, $\mathrm{I}_y^{l,k}$ and $\mathrm{I}_t^{l,k}$. Since the intermediate flow field $(\mathbf{u}^{l,k}, \mathbf{v}^{l,k})$ is derived directly from the spatiotemporal derivatives, $\mathrm{I}_x^{l,k}$, $\mathrm{I}_y^{l,k}$ and $\mathrm{I}_t^{l,k}$play a crucial role to determine the accuracy of the final flow field. More importantly, these derivatives all depend on the warping interpolation. Therefore, the final flow field is dependent on the warped interpolation results. Wedel et al. [12] proposed a blended derivative approach to modify the warped spatial derivatives I_x^k and I_y^k. However, to our knowledge, no paper is available to handle the temporal derivative I_t^k. As $\mathbf{I}_t=\mathbf{I}_{\mathrm{warp}}-\mathbf{I}_1(\mathbf{x})$, the $\mathbf{I}_{\mathrm{warp}}$ must be appropriately tackled. In the next section, we present an AGIF technique to correct errors of $\mathbf{I}_{\mathrm{warp}}$ to improve the accuracy of the optical flow estimation.

3. Adaptive guided image filter

Due to the fact that some of the displacement vectors in the intermediate flow field are not accurate, or even completely wrong, their corresponding warped interpolation pixels are not correct. For example, optical flow is undefined at occlusions because of unreliable intensity matching. Therefore, the warped interpolation pixels that come from the undefined flow vectors are wrong. Furthermore, outliers cause incorrect flow vectors and introduce artifacts to $\mathbf{I}_{\mathrm{warp}}$. In this section, an AGIF technique is presented to correct these kinds of errors of $\mathbf{I}_{\mathrm{warp}}$. Table 3 demonstrates the effectiveness of the proposed AGIF technique.

Recently, GIF was proposed [23] as a high performance image filtering method to effectively reduce noise and preserves edges. Additionally, it performs very fast. Applying the GIF to $\mathbf{I}_{\mathrm{warp}}$ with a guidance image $\boldsymbol{G}$, the output $\mathbf{F}$ at a pixel i (x, y) is defined as a weighted average:

$$\mathbf{F}_i=\frac{1}{\sum_{j\in w_k}\mathbf{W}_{i,j}(\mathbf{G})}\sum_{j\in w_k}\mathbf{W}_{i,j}(\mathbf{G})\mathbf{I}_{\mathrm{warp}}(j) \tag{13}$$

where j is the index of pixel (x', y'), w_k is the kernel window centered at pixel k, and the kernel weights function $\mathbf{W}_{i,j}(\mathbf{G})$ can be expressed as:

$$\mathbf{W}_{i,j}(\mathbf{G})=\frac{1}{|w|^2}\sum_{k:\ (i,j)\in w_k}\left(1+\frac{(\mathbf{G}_i-\mu_k)(\mathbf{G}_j-\mu_k)}{\sigma_k^2+\epsilon}\right) \tag{14}$$

where μ_k and σ_k^2 are the mean and variance of the guidance image $\mathbf{G}$ in a local window w_k, and $|w|$ is the number of pixels in w_k (refer to [23] for more details).

From Eq. (13) we can see that the filtering weights depend on the guidance image $\mathbf{G}$. Locally (in a local windoww_k), the output $\mathbf{F}$ obtains approximate details of $\mathbf{G}$. Globally, the output $\mathbf{F}$ is similar to $\mathbf{G}$. According to this characteristic, we select the reference image $\mathbf{I}_1$ as the guidance image $\mathbf{G}$. The GIF will transfer valid information from $\mathbf{I}_1$ to $\mathbf{I}_{\mathrm{warp}}$ during the filtering, which is very useful to correct imperfect details of $\mathbf{I}_{\mathrm{warp}}$. Some classical techniques to handle occlusions in optical flow estimation in term of label out the occluded pixels and exclude them from the data term [21] or apply diffusion approaches to approximate the occluded flow vectors with their neighbors [32]. In contrast, we correct matching errors due to occlusions or outliers with the application of the GIF – the unreliable intensity matching between $\mathbf{I}_{\mathrm{warp}}$ and $\mathbf{I}_1$ will be reduced and remedied as the un-occluded useful information in $\mathbf{I}_1$ is propagated to $\mathbf{I}_{\mathrm{warp}}$.

However, the $\mathbf{G}$ only considers the difference between pixel i and its neighbors so the error-correction capacity of the traditional GIF is limited. In addition, although we hope $\mathbf{I}_{\mathrm{warp}}$ should be as similar to $\mathbf{I}_1$ as possible – less interpolation error implies a more accurate estimated flow field [29], $\mathbf{I}_{\mathrm{warp}}$ must not be equal to $\mathbf{I}_1$. If $\mathbf{I}_{\mathrm{warp}}$ equals $\mathbf{I}_1$, implying that $\mathbf{I}_{\mathrm{t}}$ is 0, no valid flow vectors can be computed. Based on this significant feature, we should not simply consider the reference image $\mathbf{I}_1$ but also require to consider $\mathbf{I}_{\mathrm{warp}}$ for filtering.

Recently, Ham et al. [35] stated that jointly leveraging structural information of guidance image and input image, rather than unilaterally transferring structures of guidance image to the output image, is more effective for denoising. Inspired by the blended technique of [12], which uses the spatial derivatives $\nabla\mathbf{I}_1$ and the warped spatial derivatives $\mathbf{I}_{\mathrm{x}}$ and $\mathbf{I}_{\mathrm{y}}$ to construct a blended version of derivatives $(\mathbf{I}_{\mathrm{x}}, \mathbf{I}_{\mathrm{y}})=(1-\beta)\cdot(\mathbf{I}_{\mathrm{x}}, \mathbf{I}_{\mathrm{y}})+\beta\cdot(\mathbf{I}_{1\mathrm{x}}, \mathbf{I}_{1\mathrm{y}})$, we blend $\mathbf{I}_1$ and $\mathbf{I}_{\mathrm{warp}}$ to reconstruct a new guidance image $\mathbf{G}$, which takes into account of the properties of both $\mathbf{I}_1$ and $\mathbf{I}_{\mathrm{warp}}$ iteratively:

$$\mathbf{G}=\mathbf{W}_{\mathrm{IG}}\mathbf{I}_{\mathrm{warp}}+(1-\mathbf{W}_{\mathrm{IG}})\mathbf{I}_1 \tag{15}$$

where $\mathbf{W}_{\mathrm{IG}}$ is a weight to control the relative importance between $\mathbf{I}_1$ and $\mathbf{I}_{\mathrm{warp}}$ for reconstruction.

The temporal derivative $\mathbf{I}_{\mathrm{t}}$ is a good error measure to evaluate the deviation. It is widely used for checking whether the computed flow satisfies the initial model constraints, indicating occlusions [17] and evaluating performances of the optical flow algorithms [29]. In this paper, we use it to compute $\mathbf{W}_{IG}$:

$$\mathbf{W}_{\mathrm{IG}}=\exp\left(-\frac{\mathbf{I}_t^2}{\sigma_{\mathrm{IG}}}\right)=\exp\left(-\frac{\left(\mathbf{I}_{\mathrm{warp}}-\mathbf{I}_1\right)^2}{\sigma_{\mathrm{IG}}}\right) \tag{16}$$

For one pixel, if the deviation is small, its corresponding estimated flow vector is accurate, thus, large weight is given to $\mathbf{I}_{\mathrm{warp}}$ for construction. If the deviation is large, its corresponding estimated flow vector is inaccurate and the interpolation pixel needs to be corrected. Consequently, large weight is given to $\mathbf{I}_1$ for construction. To keep $\mathbf{I}_{\mathrm{warp}}$ as the majority in GIF, we threshold $\mathbf{W}_{\mathrm{IG}}$ at each pixel i:

$$\mathbf{W}_{\mathrm{IG}}(i)=\begin{cases}\mathbf{W}_{\mathrm{IG}}(i) & \text{if } \mathbf{W}_{\mathrm{IG}}(i)\geq 0.8\\ 0.8 & \text{otherwise}\end{cases} \tag{17}$$

In general, the $\mathbf{W}_{\mathrm{IG}}$ should be constrained to [0.6 0.95]. Its actual value is not very significant as we are only interested in keeping $\mathbf{I}_{\mathrm{warp}}$ as the primary part in the reconstructed guidance image $\mathbf{G}$. In this paper, we select 0.8 as the threshold experimentally. In Fig. 1, we use the *Urban3* sequence to illustrate the effect of the proposed GIF method. The threshold weight (Fig. 1(b)) reflects the sequence features better than the un-threshold weight (Fig. 1 (a)). For example, the occlusion status is clearly displayed in Fig. 1 (b). By applying the threshold, at occlusions and edges, the $\mathbf{I}_{\mathrm{warp}}$ as

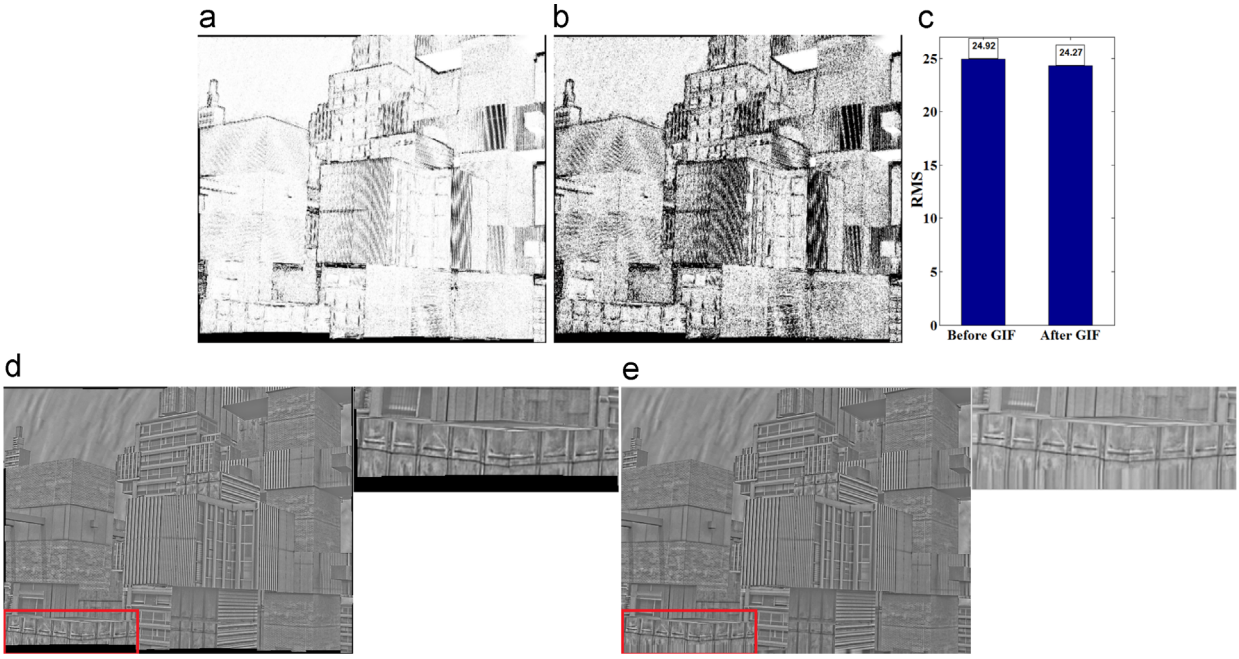

Fig. 1. Illustration of the proposed GIF method by using of Urban3 (at the last warping step). (a) The weight $\mathbf{W}_{IG}$ without a threshold. (b) The weight $\mathbf{W}_{IG}$ with threshold 0.8. (c) The RMS comparison of $\mathbf{I}_{warp}$ before and after the application of our GIF. (d) $\mathbf{I}_{warp}$ before the application of our GIF method. (e) $\mathbf{I}_{warp}$ after the application of our GIF method.

the primary component is integrated into **G** and guides the filtering. In contrast, without the threshold, at some occlusions and edges, $\mathbf{I}_1$ is selected as a primary component for constructing the guidance image. However, this deviates the truth since at these regions the structures of $\mathbf{I}_{warp}$ and $\mathbf{I}_1$ are quite different. With the application of our proposed GIF, the occluded error pixels in $\mathbf{I}_{warp}$ are partly remedied, as shown in Fig. 1(e). It can be also demonstrated in Fig. 1(c) that after using our GIF, the root-mean-square (RMS) error of $\mathbf{I}_{warp}$ is reduced. Fig. 2 displays the benefits of reconstruction a new guidance image **G** and threshold the weight $\mathbf{W}_{IG.}$

The regularization parameter ε of the GIF controls the smoothing degree. The larger ε is, the smoother the filtered image will be [23]. According to some classical literatures and various experiments, we find that the optimal parameter ε is related to the deviation of $\mathbf{I}_t$ as well as the size of input images: the larger $\mathbf{I}_t$, the larger ε should be set. The smaller size of the input images, the ε should be also set larger. Based on this observation, an AGIF is proposed to adaptively select the optimal parameter ε.

$$\epsilon = \begin{cases} \min(10^{-4}\cdot 100^{NR}\cdot 10^{ER}, 100) & \text{if ErrR} < 0.1 \\ \min(10^{-3}\cdot 100^{NR}\cdot 10^{ER}, 100) & \text{else if ErrR} \geq 0.1 \ \& \\ & \text{ErrR} < 0.2 \\ \min(10^{-2}\cdot 100^{NR}\cdot 10^{ER}, 100) & \text{otherwise} \end{cases} \tag{18}$$

ErrR is an error confidence [33], it represents the degree of error pixels in $\mathbf{I}_{warp,}$ and is computed as follows:

$$\text{ErrR} = \text{ErrN}/(\text{H.W}) \tag{19}$$

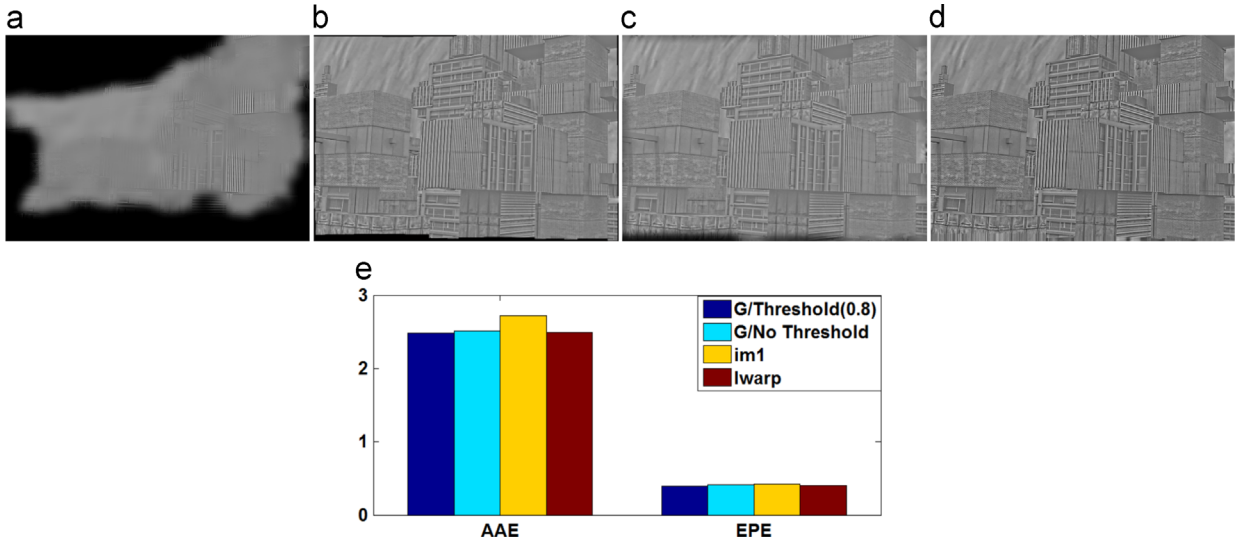

Fig. 2. Results (at the last warping step) of Urban3. (a) The filtered $\mathbf{I}_{warp}$ in term of guidance image $\mathbf{I}_1$. (b) The filtered $\mathbf{I}_{warp}$ in term of guidance image $\mathbf{I}_{warp}$. (c) The filtered $\mathbf{I}_{warp}$ in term of the reconstructed guidance image **G** without threshold $\mathbf{W}_{IG}$. (d) The filtered $\mathbf{I}_{warp}$ in term of **G** with threshold $\mathbf{W}_{IG}$. (e) The AAE and EPE of the 4 different GIF strategies.

ErrN is the number of error pixels in $\mathbf{I}_{warp}$, and H and W are the height and width of $\mathbf{I}_{warp}$ respectively. Bruhn and Weickert [33] pointed out that at locations where the deviation is small, the computed flow is accurate. At locations where the deviation is large, the optical flow model assumptions are violated severely. According to this statement, we consider one pixel to be erroneous if $\mathbf{W}_{IG}(i) < 0.8$.

ER is an error confidence [29,36], which represents the RMS of $\mathbf{I}_t$, and it is given by:

$$\mathrm{ER} = \mathrm{round}\big(\mathrm{RMS}(\mathrm{I_t})/10\big) \tag{20}$$

where round is the mathematical round operation. The RMS of $\mathbf{I}_t$ is the basic and widely measure to evaluate the performance of variational optical flow algorithms without knowing the ground truth [29,36]. In theory, the worse the flow field is, the larger the RMS(I_t) becomes, and the better the flow field is, the smaller the RMS(I_t) gets. This is because the more accurate of the estimated flow field ($\mathbf{u}$, $\mathbf{v}$), the better match that can be achieved between the two frames, and the smaller of the RMS(I_t) will be.

NR is a measure of the size of the input images. If the ratios of noise in two different sized images are the same, from visual, the small sized image appears more noisy. From the mathematical side, their probability to capture noise is different. It is much easier to get noise in the small size image. In other words, the small size image looks like noisier. Hence, for the smaller image, the parameter ε should be set larger. In this paper, we select the resolution $640 \cdot 480$ as a fixed divide for all the experiments [37]. NR is computed:

$$\mathrm{NR} = \max(0, \mathrm{round}(640.480/H/W)-1) \tag{21}$$

This strategy can adaptively select a proper parameter ε to improve the smoothing. As shown in Fig. 3, noise at smooth areas are reduced with the adaptive ε selection strategy, and the motion boundary over-smoothing problem is also reduced. Furthermore, in Table 2, we test the influence of the proposed size fact NR quantitatively. By comparing the results of AGIF (NoSize) i.e. deleting the NR from ε, with AGIF of the 4 small size sequences, we can find that with the contribution of the proposed size fact NR, the estimation accuracy is increased (Table 3).

4. Implementation and experimental results

4.1. Implementation

To be robust against illumination changes, we preprocess the input images by applying the Rudin–Osher–Fatemi (ROF) [26] based structure-texture decomposition [12]. The incremental coarse-to-fine technique in combination with the fixed pointed iteration scheme is used to estimate the optical flow. In the outer loop of the optimization, a graduated nonconvexity (GNC) strategy is used [34], which linearly combines a quadratic penalty function with a generalized Charbonnier penalty function [10]: $E_\gamma = (1-\gamma)\,E_Q(u, v) + \gamma E_C(u, v)$. According to [10,38], we implement 3 GNC steps with $\gamma = \{0, 0.5, 1\}$. In building the pyramid, we use a downsampling factor of 2/3 (see Table 4), and the number of pyramid levels is adaptively determined with a constraint that the coarsest level has a height or width around 20–30 pixels. Referring the source code of the WLIF-Flow method [38] for more detail, which is available at: www.projects.science.uu.nl/opticalflow/WLIF-Flow/.

At each warping step, we apply the CPF method [16] to denoise the intermediate flow field: a 7×7 WMF is used to smooth the Sobel detected edge regions, a 9×9 BF is used to smooth the occluded regions and a 5×5 fast MF is used to smooth other flat regions. Look Fig. 4 to overview the proposed method. All the experiments have been executed on a PC with an Intel Core i5-2410M 2.30 GHz processor and 4 GB memory, and the pixel values of the experimental images are in the range [0, 255].

Table 2
AAE/EPE results of four training sequences from the Middlebury dataset and the UCL dataset v1.2.

	Venus	YoesmiteSun	Mayan1	Mayan2
GIF	3.206/0.230	2.691/0.150	2.460/0.909	1.391/0.238
AGIF(NoSize)	3.161/0.228	2.705/0.151	2.447/0.911	1.403/0.233
AGIF	3.028/0.221	2.484/0.132	2.398/0.895	1.400/0.229

4.2. Experimental results

This paper focuses on using an AGIF technique to smooth $\mathbf{I}_{warp}$ to correct errors that may be caused by outliers and occlusions during numerical iterations. For a quantitative measurement, we first test our method on eight training sequences from the Middlebury dataset [29]. Table 5 shows the average angular error (AAE) and the average end-point error (EPE) of four methods: the proposed method, referred to as AGIF+OF, the relevant method GIF+OF (i.e. without adaptive ε selection strategy), the Classic+NL method [10] and the Classic+CPF method [16]. The four methods have similar frameworks – the same data term and the smoothness term, and the similar non-local smoothing strategies. Comparing our AGIF+OF method to both the Classic+NL method and the Classic+CPF, we can find that with the application of the AGIF technique, the accuracy of the estimated flow field is improved. Especially for some complex sequences, such as *RubberWhale*, *Venus* and *Urban3* (contain an amount of outliers and occlusions), the AAE and EPE of our method is much lower than other two methods. Additionally, Table 5 compares the performance of the GIF technique with the AGIF technique. Clearly, our AGIF

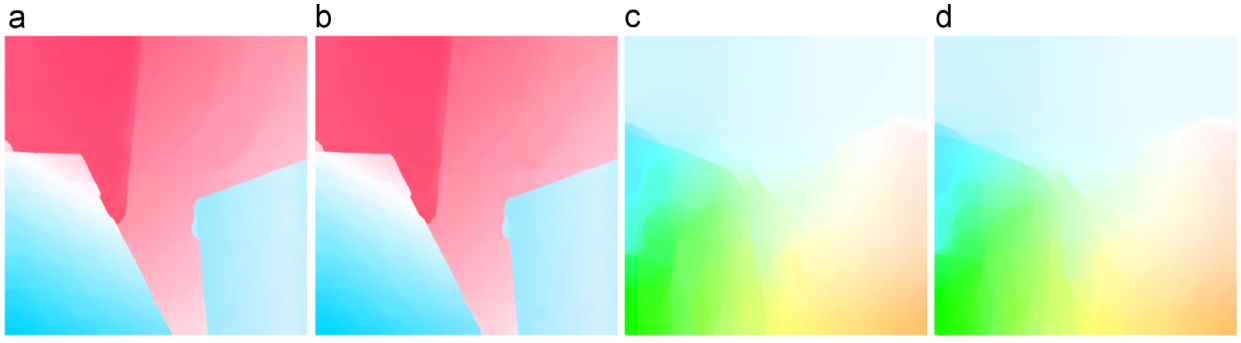

Fig. 3. Comparison of the estimated flow fields on the Venus sequence (from Middlebury dataset) and the YoesmiteSun sequence (from UCL dataset v1.2). (a, b) Estimated flow fields of Venus with the GIF and the AGIF respectively. (c, d) Estimated flow fields of YoesmiteSun with the GIF and the AGIF respectively.

Table 3
AAE/EPE results of eight training sequences from the Middlebury dataset without the CPF and GNC operations.

	Urban2	Urban3	Grove2	Grove3	RubberWhale	Venus	Dimetrodon	Hydrangea
None	2.208/0.278	5.311/0.609	1.825/0.133	7.974/0.842	3.057/0.097	5.459/0.331	2.589/0.133	1.888/0.172
GIF	2.210/0.279	5.260/0.603	1.806/0.126	7.956/**0.832**	3.027/0.096	5.457/0.328	2.579/**0.131**	**1.883/0.171**
AGIF (With)	**2.190/0.270**	**5.252/0.600**	**1.801/0.121**	7.901/0.833	**3.012/0.096**	**5.417/0.328**	**2.563/0.132**	1.892/**0.169**

outperforms GIF, which demonstrates our strategy to adaptively select a proper ε is beneficial for improving smoothing. Table 5 shows the RMS errors of $\mathbf{I}_{\text{warp}}$ before AGIF and after AGIF at the final warping step of the last iteration. After using the AGIF, some errors in $\mathbf{I}_{\text{warp}}$ were corrected. Besides, in the whole coarse-to-fine and iteration procedure, the GIF will be implemented many times, and the accumulated difference is large. Therefore, the preciseness of the temporal derivative $\mathbf{I}_t$ is improved and the final optical flow accuracy can be largely improved. It can be further demonstrated in Table 6, where we compare the GIF with our AGIF in terms of the training sequences from the MPI-Sintel dataset (using frame 10 and frame 11 for testing). The AAE/EPE results of our AGIF are more accurate than the results of GIF. Fig. 5 shows the AGIF technique has the correction ability from a visual aspect. By comparing the corresponding flow fields of each sequence, especially at occlusions and discontinuities (red rectangle highlighted regions), we find that errors are reduced with the application of our AGIF technique. Like the *Urban3*, the occlusion caused errors, are clearly seen in Fig. 5(f)–(h), while they are effectively remedied in Fig. 5(i).

Furthermore, we test our method on some sequences from the MIT dataset (available at http://people.csail.mit.edu/celiu/motionAnnotation/) and the UCL dataset (available at http://visual.cs.ucl.ac.uk/ pubs/flowConfidence/supp/indx.html). Table 7 shows the AAE and EPE of three methods: our AGIF+OF method, the Classic+NL [10] and the Classic+CPF [16]. Both the lowest AAE and EPE of our method verify the validity of the proposed AGIF technique. For some sequences, such as *Robot*, the improvement even reaches to 50%. Fig. 6 shows the visual flow fields of sequences *fish, Cameramotion, Crates2 and drop1Txtr1* with respect to these three methods. For the *fish* sequence, we can see a large number of noise and outliers that are distributed in the fields of the Classic+NL and the Classic+CPF algorithms, in contrast, the noise and outliers are greatly reduced in our flow field. Besides, the motion parts of the four fishes are clearly estimated. For the *Cameramotion* sequence, the errors caused by occlusion around the rear-vision mirror region are corrected. For others: *Crates2* and *drop1Txtr1*, our AGIF technique also has high-performance. One remarkable discovery is that for all the four tested sequences, outliers are smoothed out, occluded parts are adequately recovered, and boundaries are well preserved.

We evaluate our AGIF+OF method on the Middlebury benchmark. Fig. 7 shows that both the AAE and the EPE rank 15th at the time of submission, and the average normalized interpolation error (ANIE) is more accurate than some methods which only have a good performance on the AAE and EPE. Most importantly, we can see that our method outperforms nearly all the TV-L1 non-local (NL) approaches, such as the Efficient-NL algorithm, the Classic+NL algorithm, the NL-TV-NCC algorithm and the Occlusion-TV-L1 algorithm (refer to the Middlebury testing website for more details). Moreover, our AGIF technique is beneficial for all Euler–Lagrange based variational algorithms which apply both the GNC method and the warping strategy. Table 8 shows the performance of the classical HS [8] method and the BA [9] method is significantly improved when our AGIF is used.

We also evaluate our AGIF+OF method on the new MPI-Sintel [14] dataset. From Table 9, we can see that our method is much better than the Classic+NL [10] method. More importantly, on the clean testing part, our AGIF+OF method outperforms the MDP-Flow2 method [2], which is the topmost algorithm on the Middlebury testing. The MDP-Flow2 method has similar idea as us, which focuses on reducing initial flow errors and preserving motion details during the coarse-to-fine refinement framework. The result demonstrates that correcting errors of $\mathbf{I}_{\text{warp}}$ during optimization with our AGIF technique is valid and useful.

Fig. 8 shows three examples from the Middlebury real sequences *Backyard*, *Basketball*, and *MiniCooper*. We can find that the proposed AGIF+OF method is effective to handle some complex real conditions (comparing the red rectangle labeled regions of the flow fields of three methods). For the *Backyard*, because of the occlusion, the motion of the right foot of the little boy, the right leg and left foot of the oldest girl, and the right leg of the little girl who is hold, is lost or is wrong in Fig. 8(a) and (b). In contrast, the motion of these parts is recovered in Fig. 8(c). For the *Basketball*, as shown in Fig. 8(f), the motion of the arms and hands of the right person is correctly estimated, and his profile is clearly recovered that we can easily discern his hair, noise and mouth. The motion of the basketball and its corresponding shadow are also correctly computed. However, the estimated flow vectors of these parts contain some serious errors with the other two methods [10,16]. For example, the motion of the shadow of the basketball is confused with the background in Fig. 8(d), while some detailed motion information of the profile of the right person is oversmoothed in Fig. 8(e). For the *MiniCooper*, it is hard to capture the motion of the head and arms of the man due to noise and occlusion. Therefore, some important motion of these parts are unrecovered in Fig. 8(g) and (h). Fig. 8(i) shows that these problems are properly tackled with the AGIF technique: the motion boundary is sharp enough to distinguish the man and the outline of the window of the trunk is clearly reflected. To be specific, the motion of the forehead and the right arm is lost in Fig. 8(g) and (h), while is correctly estimated in Fig. 8(i).

Fig. 9 shows the results of the challenging *Football* sequence. For the *Football* sequence, errors are easily generated at regions of the player's right foot and hand, as well as the football. Comparing Fig. 9(c) with (d), we can find that although the state-of-the-art MDP-Flow2 method [2] can recover the motion at these regions, the motion information, especially the motion boundaries, is very

Table 4
AAE/EPE results of eight training sequences from the Middlebury dataset with different downsampling factors.

	Urban2	Urban3	Grove2	Grove3	RubberWhale	Venus	Dimetrodon	Hydrangea
Factor=1/2	1.889/0.213	2.512/0.387	1.306/0.092	4.953/0.481	2.231/0.071	3.073/0.223	2.468/0.127	1.887/0.159
Factor=4/5	1.894/0.211	3.211/0.457	1.307/0.092	4.454/0.449	2.226/0.070	3.002/0.220	2.472/0.127	1.887/0.158
Factor=2/3	**1.825/0.204**	**2.465/0.357**	**1.272/0.090**	**4.373/0.437**	**2.166/0.069**	**3.028/0.221**	**2.429/0.125**	**1.861/0.155**

Input: two images $\mathbf{I}_1$ and $\mathbf{I}_2$
Preprocess $\mathbf{I}_1$ and $\mathbf{I}_2$ (structure-texture decomposition)
for l = 1 to *max_level* (automatically compute)
  Compute pyramid images ${}^{l}\mathbf{I}_1$ and ${}^{l}\mathbf{I}_2$;
end
Initialize
$\mathbf{W}$ = 0, and L = *max_level*; ($\mathbf{W}$ = ($\mathbf{u}$, $\mathbf{v}$) is the flow field)
for iGNC = 1 to *gnc_iteration*
  for l = L to 1
    for k = 1 to *warp_iteration;*
      if l == 1
        warp_iteration = *warp_iteration*+2
      end
      Initialize $d\mathbf{W}^{l,k}$ = 0; ($d\mathbf{W}^{l,k}$ = ($d\mathbf{u}$, $d\mathbf{v}$)^{l,k} is the incremental flow field, see Eq. (11))
      Set $\mathbf{W}^{l}$ = $\mathbf{W}^{l}$ + $d\mathbf{W}^{l,k}$ (before compute $d\mathbf{W}^{l,k}$)
      if l==1 && k >1
        Perform AGIF
      else
        Perform GIF
      end
      Compute $d\mathbf{W}^{l,k}$ by solving Eq. (12) with *SOR* solver
      Update $\mathbf{W}^{l}$ = $\mathbf{W}^{l}$ + $d\mathbf{W}^{l,k}$
      Perform CPF to $\mathbf{W}^{l}$
    end
  end
end
Output: optical flow field $\mathbf{W}$ ($\mathbf{u}$, $\mathbf{v}$)

Fig. 4. Method overview.

Table 5
AAE/EPE results of eight training sequences from the Middlebury dataset and the RMS error of $\mathbf{I}_{warp}$ of the proposed AGIF+OF method.

	Urban2	Urban3	Grove2	Grove3	RubberWhale	Venus	Dimetrodon	Hydrangea
Classic+NL [3]	2.011/0.211	2.862/0.413	1.413/0.098	4.833/0.458	2.327/0.072	3.256/0.232	2.476/0.127	**1.816**/**0.151**
Classic+CPF [13]	1.866/**0.203**	2.604/0.402	1.323/0.092	4.686/0.456	2.285/0.071	3.181/0.229	2.491/0.127	1.856/0.154
GIF+OF	1.896/0.219	**2.438**/0.384	1.319/0.092	4.920/0.476	2.231/0.071	3.206/0.230	2.468/0.127	1.880/0.157
AGIF+OF	**1.825**/0.204	2.465/**0.357**	**1.272**/**0.090**	**4.373**/**0.437**	**2.166**/**0.069**	**3.028**/**0.221**	**2.429**/**0.125**	1.861/0.155
	RMS	RMS	RMS	RMS	RMS	RMS	RMS	RMS
$\mathbf{I}_{warp}$ (Before AGIF)	17.2176	24.9190	17.1546	25.4570	9.6728	17.2574	8.9276	15.4837
$\mathbf{I}_{warp}$ (After AGIF)	**16.6114**	**24.2727**	**15.8344**	**23.8594**	**9.1588**	**16.2729**	**8.5278**	**14.5161**

Table 6
AAE/EPE results of the training sequences (both clean pass and final pass) from the MPI-Sintel dataset.

Clean	alley_2	ambush_2	bamboo_2	cave_2	market_2	shaman_2	sleeping_2	temple_2
GIF+OF	2.266/0.208	13.409/7.088	5.175/0.518	3.029/0.922	6.056/1.048	4.363/0.317	1.295/0.060	4.000/0.707
AGIF+OF	**2.131**/**0.205**	**13.379**/**7.075**	**5.138**/**0.501**	**3.020**/**0.913**	**5.906**/**1.035**	**4.258**/**0.301**	**1.273**/**0.052**	**3.904**/**0.675**
Final								
GIF+OF	2.242/0.171	47.840/28.817	5.503/0.598	3.321/1.188	7.864/1.359	4.965/0.337	1.336/0.062	7.552/1.329
AGIF+OF	**2.223**/**0.168**	**47.705**/**28.798**	**5.493**/**0.584**	**3.211**/**1.152**	**7.797**/**1.332**	**4.895**/**0.318**	**1.321**/**0.051**	**7.413**/**1.293**

dim (see Fig. 9(d)). For example, the motion boundaries of the right fingers of the player is smoothened too much to discern, and also the motion boundary of the football is not distinctive. In contrast, our AGIF+OF method (see Fig. 9(c)) corrects these defects. The motion boundaries of the fingers are well preserved. The outline of the football is very clear. Additionally, we can even distinguish the right shoe of the player. All in all, from the well preserved motion boundaries of our flow field, we can see that our AGIF technique is good at correcting motion errors.

In the last experiment, we select the KITTI complex outdoor sequences dataset for testing (Available at http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=flow). As shown in Fig. 10 (a) and (c), our AGIF+OF method correctly estimates the motion fields of them. In Fig. 10(a), the motion of the car, and the motion of the trees at the right side are distinctive. In Fig. 10(c), the motions of the two cars, the rider and even his shadow, and the backgrounds (e.g. houses and the mountain-like building) are all clearly recovered. The warped images, as shown in Fig. 10(b) and (d), are also correct.

5. Discussion and conclusions

This paper proposed an effective AGIF technique to improve the performance of the variational optical flow method. It corrects

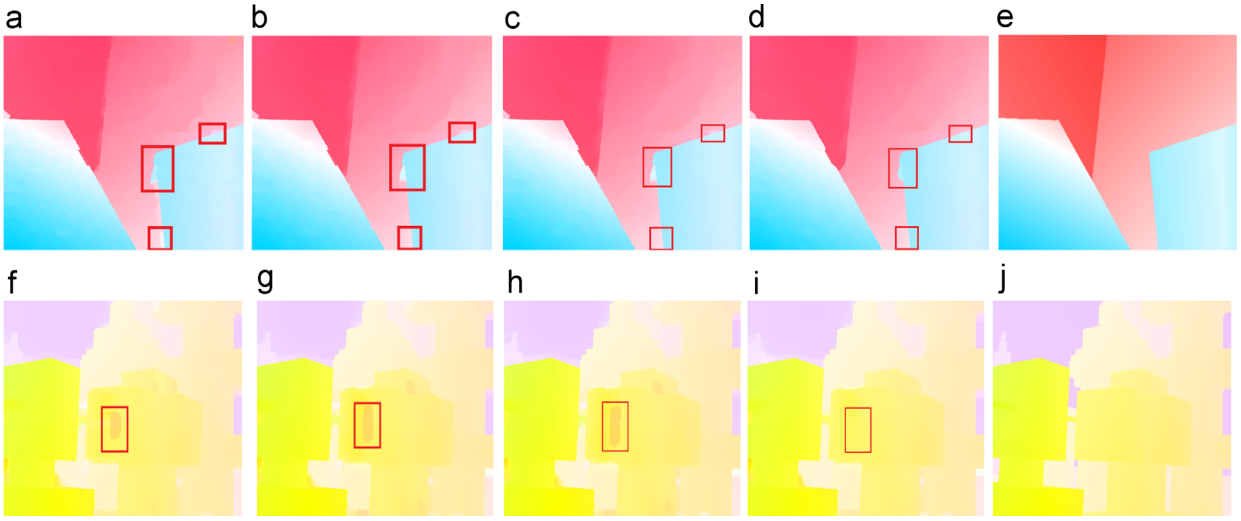

Fig. 5. Visual comparison of the estimated flow fields on sequences Venus and Urban3 (each row, from left to right). (a, f) Estimated flow fields of the Classic+NL[10]; (b, g) Estimated flow fields of the Classic+CPF[16]; (c, h) Estimated flow fields without using the proposed AGIF technique. (d, i) Estimated flow fields with using the proposed AGIF technique. (e, j) Corresponding ground-truths.

Table 7
AAE/EPE results of training sequences from the MIT dataset and the UCL dataset.

	Fish	Cameramotion	Table	Crates2	Robot	Brickbox1t1	blow1Txtr1	drop1Txtr1
Classic+NL	20.574/0.837	6.819/0.592	4.032/1.197	16.614/10.301	9.276/1.534	0.657/0.225	0.528/0.026	1.329/0.050
Classic+CPF	15.306/0.627	5.785/0.533	3.750/1.140	16.458/10.390	8.898/1.514	0.618/0.237	0.428/0.023	1.010/0.041
AGIF+OF	**12.325/0.505**	**5.220/0.499**	**3.618/1.237**	**13.156/2.421**	**4.064/0.886**	**0.525/0.190**	**0.407/0.020**	**0.898/0.036**
Improvement	19.48%/19.46%	9.77%/6.38%	3.52%/8.51%	20.06%/76.7%%	54.33%/41.48%%	15.05%/30.4%	4.91%/13.04%	11.09%/12.2%

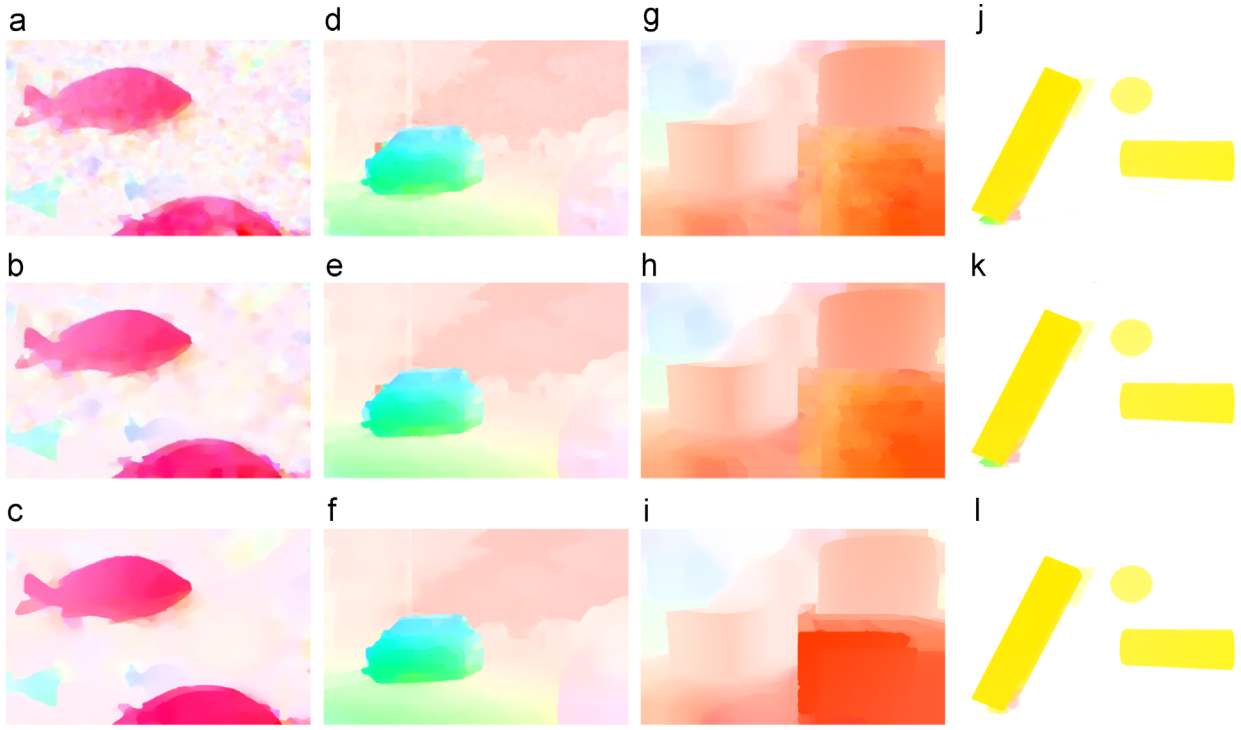

Fig. 6. Visual comparison of the estimated flow fields on sequences fish, Cameramotion, Crates2 and drop1Txtr1 (each row, from left to right). (a), (d), (g) and (j) Estimated flow fields of the method Classic+NL[10]. (b), (e), (h) and (k) Estimated flow fields of the method Classic+CPF[16]. (c), (f), (i) and (l) Estimated flow fields of the proposed method AGIF+OF.

a

Average angle error	avg. rank	Army (Hidden texture)			Mequon (Hidden texture)			Schefflera (Hidden texture)			Wooden (Hidden texture)			Grove (Synthetic)			Urban (Synthetic)			Yosemite (Synthetic)			Teddy (Stereo)		
		GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext
AGIF+OF [92]	19.8	3.06 14	8.20 13	2.55 16	3.17 40	10.6 33	2.46 39	3.46 21	8.97 21	2.24 26	2.61 10	13.7 14	1.33 13	2.63 12	3.46 10	2.11 16	2.88 11	8.34 10	2.35 8	2.10 16	3.56 27	2.09 28	1.80 25	3.68 24	2.24 29
Efficient-NL [60]	20.0	2.99 11	8.23 16	2.28 7	2.72 21	8.95 21	2.25 25	3.81 29	9.87 28	2.07 18	2.77 22	14.3 21	1.46 24	2.61 10	3.48 11	1.96 9	3.31 21	8.33 9	2.59 15	2.60 40	3.75 35	2.54 41	1.60 18	3.02 11	1.66 17
ALD-Flow [67]	20.3	2.82 5	7.86 7	2.16 5	2.84 27	10.1 27	1.86 15	3.73 27	10.4 30	1.67 10	3.10 31	16.8 40	1.28 9	2.69 15	3.60 18	1.85 7	2.79 8	11.3 28	2.32 7	2.07 13	3.25 7	3.10 62	2.03 32	5.11 34	1.94 24
IROF++ [58]	21.4	3.17 22	8.69 23	2.61 19	2.79 24	9.61 24	2.33 28	3.43 18	8.86 18	2.38 31	2.87 25	14.8 25	1.52 30	2.74 17	3.57 17	2.19 19	3.20 18	9.70 20	2.71 20	1.96 9	3.45 16	1.22 5	1.80 25	4.06 26	2.50 35
SCR [73]	21.7	3.12 16	8.48 19	2.59 17	2.95 33	10.4 30	2.35 30	3.19 12	8.09 12	2.43 35	2.63 13	13.9 17	1.35 15	2.81 20	3.64 20	2.30 22	3.02 13	8.29 8	2.39 12	2.77 48	3.79 38	2.89 54	1.39 12	2.85 9	1.60 16
Classic+CPF [90]	22.3	3.14 18	8.60 21	2.63 23	3.03 38	10.6 33	2.33 28	3.66 24	9.58 24	2.20 25	2.61 10	14.1 19	1.34 14	2.68 14	3.53 14	2.21 20	2.85 9	7.95 7	2.38 9	2.44 34	3.49 21	2.90 56	1.67 21	3.40 20	2.43 33
TC-Flow [46]	23.0	2.91 7	8.00 10	2.34 11	2.18 7	8.77 18	**1.52** 1	3.84 31	10.7 34	**1.49** 1	3.13 32	16.6 38	1.46 24	2.78 19	3.73 25	1.96 9	3.08 15	11.4 29	2.66 16	1.94 7	3.43 15	3.20 67	3.06 39	7.04 38	4.08 60
Sparse-NonSparse [56]	23.0	3.14 18	8.75 25	2.76 29	3.02 36	10.6 33	2.43 34	3.45 20	8.96 19	2.36 29	2.66 15	13.7 14	1.42 18	2.85 24	3.75 26	2.33 24	3.28 20	9.40 17	2.73 21	2.42 31	3.31 9	2.69 44	1.47 14	3.07 16	1.66 17
LSM [39]	24.6	3.12 16	8.62 22	2.75 28	3.00 35	10.5 32	2.44 36	3.43 18	8.85 17	2.35 28	2.66 15	13.6 12	1.44 20	2.82 21	3.68 21	2.36 26	3.38 24	9.41 18	2.81 25	2.69 44	3.52 24	2.84 49	1.59 17	3.38 19	1.80 23
Correlation Flow [78]	24.7	3.38 32	8.40 18	2.64 24	2.23 8	7.54 10	1.56 3	5.14 42	13.1 41	1.60 6	2.09 4	**8.15** 1	1.35 15	3.12 33	4.09 36	2.34 25	4.01 42	11.5 31	4.00 52	2.59 39	3.61 30	3.00 61	1.49 15	3.04 13	1.42 12
Ramp [62]	25.5	3.18 24	8.83 26	2.73 27	2.89 30	10.1 27	2.44 36	3.27 15	8.43 15	2.38 31	2.74 21	14.2 20	1.46 24	2.82 21	3.69 23	2.29 21	3.37 23	9.31 16	2.93 29	2.62 42	3.38 14	3.19 66	1.54 16	3.21 17	2.24 29
PMF [75]	26.0	3.61 36	9.07 27	2.62 21	2.40 11	8.05 11	1.83 14	2.61 8	6.27 8	1.65 9	3.35 40	15.4 29	1.58 33	2.54 7	3.27 7	1.71 5	3.59 34	11.1 27	3.46 43	4.07 77	6.18 85	4.02 77	1.06 3	2.38 3	1.25 9
COFM [59]	26.3	3.17 22	9.90 41	2.46 13	2.41 13	8.34 15	1.92 16	3.77 28	10.5 31	2.54 40	2.71 20	14.9 27	1.19 7	3.08 32	3.92 30	3.25 58	3.83 37	10.9 24	3.15 35	2.20 23	3.35 10	2.91 58	1.62 20	2.56 5	2.09 26
Classic+NL [31]	28.0	3.20 25	8.72 24	2.81 31	3.02 36	10.6 33	2.44 36	3.46 21	8.84 16	2.38 31	2.78 23	14.3 21	1.46 24	2.83 23	3.68 21	2.31 23	3.40 26	9.09 15	2.76 23	2.87 52	3.82 41	2.86 52	1.67 21	3.53 22	2.26 32
TV-L1-MCT [64]	28.6	3.16 21	8.48 19	2.71 26	3.28 48	10.8 42	2.60 48	3.95 33	10.5 31	2.38 31	2.69 18	13.9 17	1.45 23	2.94 28	3.79 27	2.63 42	3.50 31	9.75 21	3.06 33	2.08 14	3.35 10	2.29 32	1.95 29	3.89 25	2.71 38

b

Average endpoint error	avg. rank	Army (Hidden texture)			Mequon (Hidden texture)			Schefflera (Hidden texture)			Wooden (Hidden texture)			Grove (Synthetic)			Urban (Synthetic)			Yosemite (Synthetic)			Teddy (Stereo)		
		GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext	GT all	im0 disc	im1 untext
AGIF+OF [92]	20.2	0.08 7	0.22 13	0.07 14	0.23 41	0.73 33	0.18 33	0.28 17	0.66 22	0.18 17	0.14 7	0.70 9	0.08 11	0.57 7	0.85 7	0.38 10	0.47 29	0.97 11	0.31 21	0.13 30	0.13 26	0.22 33	0.51 29	0.99 23	0.74 34
TC-Flow [46]	21.7	**0.07** 1	0.21 3	0.06 5	**0.15** 1	0.59 10	**0.11** 1	0.31 31	0.78 34	**0.14** 1	0.16 28	0.86 31	0.08 11	0.75 31	1.11 33	0.54 29	0.42 17	1.40 43	0.25 7	0.11 11	0.12 8	0.29 61	0.62 37	1.35 37	0.93 49
FESL [74]	21.7	0.08 7	0.21 3	0.07 14	0.25 49	0.75 39	0.19 42	0.27 13	0.61 14	0.18 17	0.14 7	0.68 7	0.08 11	0.61 12	0.89 10	0.44 15	0.47 29	1.03 16	0.32 25	0.14 39	0.15 54	0.25 45	0.50 24	0.96 17	0.63 12
Classic+CPF [90]	21.7	0.08 7	0.23 18	0.07 14	0.22 29	0.73 33	0.17 24	0.30 26	0.70 25	0.18 17	0.14 7	0.72 16	0.08 11	0.63 13	0.93 14	0.45 17	0.51 39	1.03 16	0.32 25	0.14 39	0.12 8	0.30 66	0.48 18	0.93 13	0.72 26
ALD-Flow [67]	21.8	**0.07** 1	0.21 3	0.06 5	0.19 13	0.64 20	0.13 10	0.30 26	0.73 29	0.15 6	0.17 33	0.92 43	0.07 4	0.78 34	1.14 34	0.59 34	0.33 7	1.30 35	0.21 3	0.12 20	0.12 8	0.28 56	0.54 33	1.19 35	0.73 30
SCR [73]	21.8	0.08 7	0.23 18	0.07 14	0.22 29	0.71 29	0.17 24	0.27 13	0.60 13	0.19 27	0.14 7	0.73 17	0.08 11	0.63 13	0.92 13	0.44 15	0.51 39	1.08 21	0.33 30	0.15 49	0.13 26	0.29 61	0.47 15	0.93 13	0.67 20
COFM [59]	22.1	0.08 7	0.26 35	0.06 5	0.18 9	0.62 15	0.14 14	0.30 26	0.74 31	0.19 27	0.15 17	0.86 31	0.07 4	0.79 35	1.14 34	0.74 50	0.35 10	0.87 7	0.28 11	0.14 39	0.12 8	0.28 56	0.49 20	0.94 15	0.71 25
Sparse-NonSparse [56]	22.3	0.08 7	0.23 18	0.07 14	0.22 29	0.73 33	0.18 33	0.28 17	0.64 17	0.19 27	0.14 7	0.71 13	0.08 11	0.67 22	0.99 23	0.48 21	0.49 35	1.06 18	0.32 25	0.14 39	**0.11** 1	0.28 56	0.49 20	0.98 20	0.73 30
Efficient-NL [60]	22.5	0.08 7	0.22 13	0.06 5	0.21 24	0.67 22	0.17 24	0.31 31	0.73 29	0.18 17	0.14 7	0.71 13	0.08 11	0.59 9	0.88 9	0.39 11	1.30 69	1.35 38	0.67 65	0.14 39	0.13 26	0.26 47	0.45 11	0.85 6	0.55 7
LSM [39]	23.8	0.08 7	0.23 18	0.07 14	0.22 29	0.73 33	0.18 33	0.28 17	0.64 17	0.19 27	0.14 7	0.70 9	0.09 23	0.66 18	0.97 19	0.48 21	0.50 37	1.06 18	0.33 30	0.15 49	0.12 8	0.29 61	0.50 24	0.99 23	0.73 30
Ramp [62]	24.4	0.08 7	0.24 26	0.07 14	0.21 24	0.72 31	0.18 33	0.27 13	0.62 15	0.19 27	0.15 17	0.71 13	0.09 23	0.66 18	0.97 19	0.49 23	0.51 39	1.09 23	0.34 36	0.15 49	0.12 8	0.30 66	0.48 18	0.96 17	0.72 26
Classic+NL [31]	26.3	0.08 7	0.23 18	0.07 14	0.22 29	0.74 37	0.18 33	0.29 22	0.65 21	0.19 27	0.15 17	0.73 17	0.09 23	0.64 17	0.93 14	0.47 18	0.52 42	1.12 25	0.33 30	0.16 59	0.13 26	0.29 61	0.49 20	0.98 20	0.74 34
TV-L1-MCT [64]	26.5	0.08 7	0.23 18	0.07 14	0.24 45	0.77 41	0.19 42	0.32 34	0.76 33	0.19 27	0.14 7	0.69 8	0.09 23	0.72 28	1.03 24	0.60 35	0.54 44	1.10 24	0.35 37	0.11 11	0.12 8	0.20 22	0.54 33	1.04 30	0.84 41

c

Average normalized interpolation error	avg. rank	Mequon (Hidden texture)			Schefflera (Hidden texture)			Urban (Synthetic)			Teddy (Stereo)			Backyard (High-speed camera)			Basketball (High-speed camera)			Dumptruck (High-speed camera)			Evergreen (High-speed camera)		
		im0 all	GT disc	im1 untext	im0 all	GT disc	im1 untext	im0 all	GT disc	im1 untext	im0 all	GT disc	im1 untext	im0 all	GT disc	im1 untext	im0 all	GT disc	im1 untext	im0 all	GT disc	im1 untext	im0 all	GT disc	im1 untext
AGIF+OF [92]	40.0	0.61 16	0.78 17	0.64 3	0.67 21	0.92 21	0.61 19	1.17 40	2.07 57	1.37 44	0.97 20	1.06 21	1.25 35	1.20 71	1.30 71	1.69 72	1.18 50	1.75 57	1.07 24	0.75 28	1.44 32	0.72 29	1.13 89	1.72 90	0.64 34
Classic++ [32]	40.1	0.62 32	0.80 26	0.66 47	0.78 47	1.10 47	0.66 45	0.93 14	1.36 11	0.75 6	1.04 57	1.12 45	1.28 55	1.08 34	1.11 34	1.18 32	1.18 50	1.69 50	1.10 44	0.89 55	1.86 58	0.72 29	0.99 55	1.47 56	0.64 34
Sparse Occlusion [54]	40.8	0.63 47	0.87 54	0.65 26	0.77 44	1.11 50	0.63 34	0.91 9	1.45 19	0.86 15	0.96 16	1.08 26	1.21 8	1.21 78	1.32 77	1.69 72	1.14 36	1.63 44	1.12 56	0.86 50	1.77 53	0.71 24	1.04 69	1.56 69	0.63 4
Ramp [62]	41.2	0.62 32	0.82 36	0.65 26	0.66 16	0.90 16	0.61 19	1.59 64	3.35 82	2.06 64	0.95 11	1.05 19	1.21 8	1.16 51	1.25 54	1.58 59	1.22 68	1.85 70	1.12 56	0.85 47	1.73 49	0.70 18	0.96 45	1.43 45	0.64 34
Local-TV-L1 [65]	41.9	0.65 59	0.83 39	0.69 76	0.88 70	1.15 60	0.76 72	0.87 2	1.27 5	0.71 4	1.01 47	1.06 21	1.31 68	1.13 46	1.19 45	1.35 48	1.14 36	1.48 23	1.13 63	0.81 42	1.63 44	0.73 33	0.85 16	1.23 16	0.66 71
CRTflow [87]	42.0	0.64 54	0.89 58	0.67 62	0.83 57	1.16 66	0.69 56	1.12 32	1.66 32	1.02 28	1.00 40	1.10 33	1.28 55	1.18 60	1.27 59	1.69 72	1.05 7	1.33 6	1.05 11	0.93 63	1.95 64	0.69 14	0.94 35	1.40 39	0.63 4
SCR [73]	43.2	0.61 16	0.78 17	0.64 3	0.66 16	0.90 16	0.61 19	1.63 66	3.39 83	2.09 65	0.97 20	1.09 30	1.22 20	1.20 71	1.31 76	1.68 69	1.14 36	1.61 38	1.08 31	0.93 63	1.96 65	0.80 59	1.06 76	1.60 77	0.63 4
Fusion [6]	43.6	0.64 54	0.94 64	0.65 26	0.70 29	0.98 27	0.61 19	1.35 55	1.48 20	1.70 56	1.06 63	1.26 71	1.22 20	1.12 45	1.20 47	1.22 39	1.29 79	2.07 83	1.19 81	0.78 36	1.54 37	0.72 29	0.85 16	1.24 17	0.64 34
Classic+NL [31]	44.1	0.62 32	0.82 36	0.65 26	0.67 21	0.92 21	0.62 26	1.56 63	3.23 79	1.95 61	0.97 20	1.11 36	1.25 35	1.17 57	1.27 59	1.54 56	1.17 48	1.71 52	1.09 37	0.92 60	1.92 61	0.77 47	1.00 60	1.50 62	0.63 4
ACK-Prior [27]	45.1	0.61 16	0.83 39	0.64 3	0.69 27	0.98 27	0.60 12	2.41 81	1.84 44	3.18 82	1.02 51	1.12 45	1.27 48	1.22 83	1.32 77	1.72 88	1.17 48	1.64 45	1.12 56	0.79 38	1.54 37	0.78 51	0.83 13	1.19 13	0.65 59
FESL [74]	45.2	0.61 16	0.77 11	0.64 3	0.66 16	0.91 20	0.60 12	1.18 42	2.09 58	1.11 34	0.99 38	1.10 33	1.27 48	1.21 78	1.32 77	1.69 72	1.20 60	1.81 69	1.12 56	0.90 59	1.87 59	0.74 37	1.06 76	1.60 77	0.64 34
Black & Anandan [4]	45.8	0.68 75	0.96 71	0.69 76	0.94 80	1.21 76	0.76 72	2.33 79	1.75 37	2.52 75	1.08 65	1.15 54	1.25 35	1.04 17	1.05 17	1.16 23	1.11 24	1.54 25	1.07 24	0.73 22	1.37 23	0.70 18	0.87 20	1.26 19	0.66 71
Classic+CPF [90]	46.2	0.61 16	0.79 22	0.64 3	0.68 25	0.93 24	0.62 26	1.21 43	2.25 62	1.22 39	0.94 4	1.02 8	**1.19** 1	1.22 83	1.34 84	1.69 72	1.23 72	1.88 75	1.11 51	0.92 60	1.92 61	0.74 37	1.14 90	1.73 91	0.65 59
NL-TV-NCC [25]	46.6	0.63 47	0.84 45	0.65 26	0.77 44	1.10 47	0.64 39	1.02 21	1.71 34	0.90 18	1.07 64	1.30 77	1.32 71	1.07 29	1.06 18	1.38 51	1.25 75	1.91 76	1.14 67	0.75 28	1.40 26	0.74 37	0.99 55	1.46 53	0.66 71
Adaptive [20]	46.7	0.64 54	0.91 60	0.66 47	0.88 70	1.22 79	0.71 63	1.06 27	1.76 39	1.05 30	1.03 52	1.17 59	1.33 75	1.09 36	1.12 36	1.15 6	1.20 60	1.78 63	1.14 67	0.86 50	1.76 52	0.71 24	0.93 33	1.38 34	0.63 4
Efficient-NL [60]	47.0	0.61 16	0.77 11	0.64 3	0.71 34	0.99 30	0.62 26	2.03 75	1.80 43	2.75 78	0.97 20	1.11 36	1.22 20	1.18 60	1.28 62	1.64 64	1.19 58	1.77 60	1.09 37	0.96 70	2.04 71	0.78 51	1.10 85	1.65 85	0.64 34
Occlusion-TV-L1 [63]	47.2	0.62 32	0.86 51	0.66 47	0.85 61	1.20 74	0.68 52	0.92 11	1.58 30	0.90 18	1.13 74	1.43 81	1.30 65	1.04 17	1.06 18	1.15 6	1.20 60	1.78 63	1.15 72	0.89 55	1.54 37	0.83 66	1.04 69	1.56 69	0.63 4

Fig. 7. Evaluation results on the Middlebury dataset. The proposed method AGIF+OF is highlighted. (a) The AAE. (b) The EPE. (c) The ANIE.

Table 8
AAE/EPE results of eight training sequences from the Middlebury dataset of methods: HS [1] and HS+AGIF, BA [2] and BA+AGIF.

	Urban2	Urban3	Grove2	Grove3	RubberWhale	Venus	Dimetrodon	Hydrangea
HS	4.060/0.459	7.520/0.856	2.853/0.204	6.809/0.690	3.798/0.118	5.533/0.337	4.675/0.224	2.218/0.190
HS+AGIF	**2.999/0.282**	**5.749/0.668**	**1.975/0.143**	**5.638/0.550**	**3.026/0.096**	**3.621/0.245**	**4.363/0.208**	**2.047/0.174**
BA	2.965/0.376	4.728/0.605	2.492/0.172	6.496/0.660	3.156/0.097	4.752/0.293	4.110/0.199	2.060/0.177
BA+AGIF	**1.909/0.225**	**3.529/0.472**	**1.484/0.105**	**5.346/0.519**	**2.171/0.067**	**3.135/0.220**	**3.060/0.154**	**1.926/0.164**

Table 9
Evaluation results on the MPI-Sintel test set.

	AGIF+OF	MDP-Flow2 [2]	Classic+NL [10]
EPE (clean)	5.766	5.837	7.961
EPE (final)	8.514	8.445	9.153

errors which are caused by inaccurate flow vectors in the warped interpolation image $\mathbf{I}_{warp}$. This technique is especially suited to reduce errors due to outliers and occlusions. With this contribution, the accuracy of the discrete approximate temporal derivative $\mathbf{I}_t$ is modified. Consequently, this technique is useful for all Euler–Lagrange based variational methods. The AGIF technique is based on the GIF method but with two improvements. One improvement is that, we reconstruct a guidance image by properly combining the reference image $\mathbf{I}_1$ and $\mathbf{I}_{warp}$ according to a temporal derivative measured error weight. As have shown, useful information (e.g., intensity and edge) will transferred from the reference image to the filtered $\mathbf{I}_{warp}$. The other improvement is that, we use an error confidence to adaptively select the optimal regularization parameter ε, which can properly adjust the degree of smoothing. In this work, we did not find a sufficient way to use the GIF method to

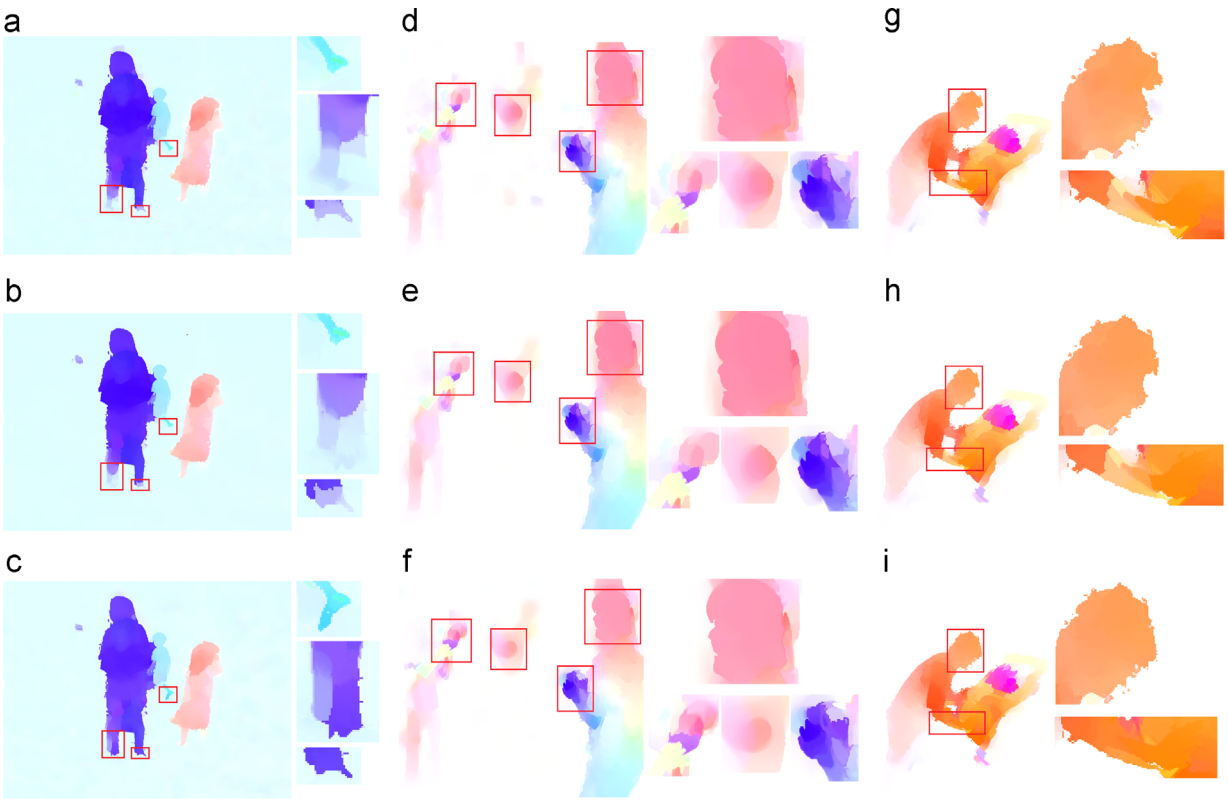

Fig. 8. Visual comparison of the estimated flow fields on sequences Backyard, Basketball, and MiniCooper (each row, from left to right) from Middlebury dataset. (a), (d) and (g) Estimated flow fields of the method Classic+NL. (b), (e) and (h) Estimated flow fields of the method Classic+CPF. (c), (f) and (i) Estimated flow fields of the proposed method AGIF+OF.

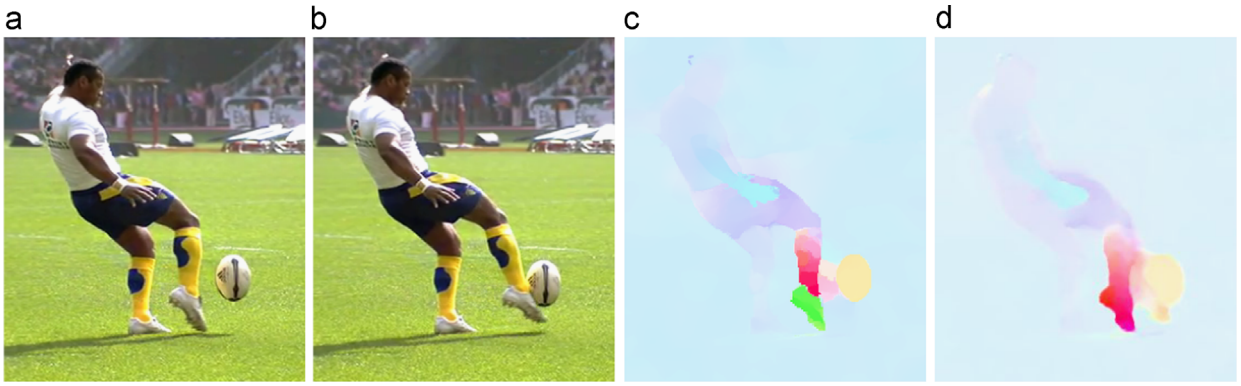

Fig. 9. Visual comparison of the proposed AGIF+OF method and the MDP-Flow2 method [2] on the Football sequence. (a) Frame 3. (b) Frame 4. (c) The result of the AGIF+OF method. (d) The result the MDP-Flow2 method.

improve the accuracy of the spatial derivatives $\mathbf{I}_x$ and $\mathbf{I}_y$, which will be a challenge for future work.

5.1. Limitations

There are several limitations of the AGIF+OF method. First, as shown in Tables 5 and 6, the improvement of AGIF over GIF is modest, because former is based on the latter and both have the same filtering process. Moreover, the approach to adaptively selecting the optimal regularization parameter ε is according to various experiments, we do not find a proper mathematical selection measure (Eq. (18)) which causes the selection measure to still suboptimal. In Table 5, we can see that the AGIF method is not better than the GIF method with respect to the Urban3.

Second, the method to combine $\mathbf{I}_1$ and $\mathbf{I}_{\text{warp}}$ to reconstruct a guidance image $\mathbf{G}$ for AGIF cannot handle all variations. For different sequences, the error degree of $\mathbf{I}_{\text{warp}}$ is totally different. Even for one sequence, the error degree between different areas may have large differences. Creating a more appropriate combined approach is very important. Furthermore, splitting the reference image $\mathbf{I}_1$ into different parts, capturing useful information and discarding invalid information should be done, but an effective way to do this has yet to be found. The results of the MPI-Sintel test set demonstrate the urgency of this important work: for the clean pass and the final pass, our method has different performances – better than MDP-Flow2 [2] in the clean pass while worse than MDP-Flow2 in the final pass. This is because the two passes have different complexities.

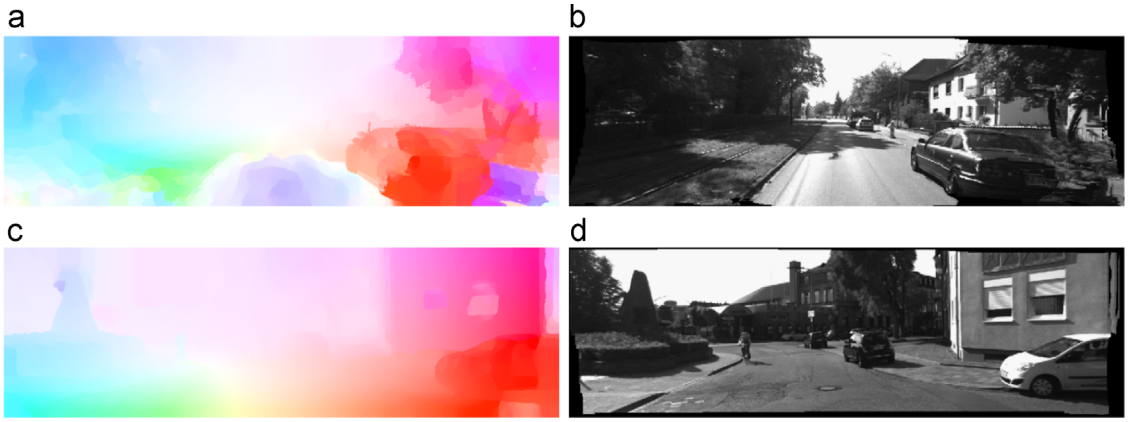

Fig. 10. Results of the proposed AGIF+OF method on real sequences 0000053 and 0000163 from the KITTI dataset. (a) The estimated flow field of the sequence 0000053. (b) The corresponding backward warped image of the sequence 0000053. (c) The estimated flow field of the sequence 0000163. (b) The corresponding backward warped image of the sequence 0000163.

Acknowledgments

This work was supported by the Dutch National Program COMMIT/, and supported in part the by Grant (#1135616) from the National Science Foundation. Any opinions expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

References

[1] 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.
[2] L. Xu, J. Jia, Yasuyuki Matsushita, Motion detail preserving optical flow estimation, IEEE Trans. Pattern Anal. Mach. Intell. 34 (9) (2012) 1744–1757.
[3] T. Brox, A. Bruhn, N. Papenberg, J. Weickert, High accuracy optical flow estimation based on a theory for warping, in: Proceedings of European Conference on Computer Vision, 2004.
[4] Y. Motai, S.K. Jha, D. Kruse, Human tracking from a mobile agent: optical flow and Kalman filter arbitration, Signal Process. Image Commun. 27 (1) (2012) 83–95.
[5] E. Memin, P. Perez, Hierarchical estimation and segmentation of dense motion fields, Int. J. Comput. Vis. 46 (2) (2002) 129–155.
[6] Y. Mochizuki, Y. Kameda, A. Imiya, T. Sakai, T. Imaizumi, Variational method for super-resolution optical flow, Signal Process. 91 (7) (2011) 1535–1567.
[7] A.B. Hadiashar, R.B. Tennakoon, M. Bruijne, Quantification of smoothing requirement for 3D optic flow calculation of volumetric images, IEEE Trans. Image Process. 22 (6) (2013) 2128–2137.
[8] B.K. Horn, B.G. Schunck, Determining optical flow, Artif. Intell. 17 (1981) 185–203.
[9] 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.
[10] D. Sun, S. Roth, M.J. Black, Secrets of optical flow estimation and their principles, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 2432–2439.
[11] H. Zimmer, A. Bruhn, J. Weickert, Optic flow in harmony, Int. J. Comput. Vis. 93 (3) (2011) 368–388.
[12] A. Wedel, T. Pock, C. Zach, D. Cremers, H. Bischof, An improved algorithm for TV–L1 optical flow, Statistical and Geometrical Applications to Visual Motion Analysis, 5064, 2008, pp. 23–45.
[13] Y. Mileva, A. Bruhn, J. Weickert, Illumination-robust variational optical flow with photometric invariants, in: Proceedings of DAGM, 2007.
[14] 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.
[15] A. Bruhn, J. Weickert, C. Schnörr, Lucas/Kanade meets Horn/Schunck: combining local and global optic flow methods, Int. J. Comput. Vis. 61 (3) (2005) 211–231.
[16] Z. Tu, N. Aa, R.C. Veltkamp, A combined post-filtering method to improve accuracy of variational optical flow estimation, Pattern Recognit. 47 (5) (2014) 1926–1940.
[17] 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.
[18] J. Kim, J. Kim, Effective nonlinear approach for optical flow estimation, Signal Process. 81 (10) (2001) 2249–2252.
[19] C. Vogel, S. Roth, K. Schindler, An evaluation of data costs for optical flow, in: Proceedings of German Conference on Pattern Recognition, 2013, pp. 343–353.
[20] O. Demetz, D. Hafner, J. Weickert, The complete rank transform: a tool for accurate and morphologically invariant matching of structures, in: Proceedings of BMVC, 2013.
[21] P. Sand, S. Teller, Particle video: long-range motion estimation using point trajectories, Int. J. Comput. Vis. 80 (1) (2008) 72–91.
[22] H.A. Rashwan, M.A. García, Puig Domenec, Variational optical flow estimation based on stick tensor voting, IEEE Trans. Image Process. 22 (7) (2013) 2589–2599.
[23] K. He, J. Sun, X. Tang, Guided image filtering, in: Proceedings of European Conference on Computer Vision, 2010, pp.1–14.
[24] G. Petschnigg, M. Agrawala, H. Hoppe, R. Szeliski, M. Cohen, K. Toyama, Digital photography with flash and no-flash image pairs, in: Proceedings of SIGGRAPH, 23, 3, 2004, pp. 664–672.
[25] Y.Q. Zhang, Y. Ding, J. Liu, Z. Guo, Guided image filtering using signal subspace projection, IET Image Process. 7 (3) (2013) 270–279.
[26] C.C. Pham, U. Ha, J.W. Jeon, Adaptive guided image filtering for sharpness enhancement and noise reduction, in: Proceedings of Advances in Image and Video Technology, Lecture Notes in Computer Science, 7087, 2012, pp. 323–334.
[27] J. Lu, H. Yang, D. Min, M.N. Do, PatchMatch filter: efficient edge-aware filtering meets randomized search for fast correspondence field estimation, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2013.
[28] A. Hosni, C. Rhemann, 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.
[29] 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, Available ⟨http://vision.middlebury.edu/flow/data/⟩.
[30] L.I. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, J. Phys. D 60 (1) (1992) 259–268.
[31] D.M. Young, Iterative Solution of Large Linear Systems, Academic Press, New York, 1971.
[32] S. Ince, J. Konrad, Occlusion-aware optical flow estimation, IEEE Trans. Image Process. 17 (8) (2008) 1443–1451.
[33] A. Bruhn, J. Weickert, Confidence measures for variational optic flow methods, Geometric Properties for Incomplete data, 31, 2006, pp. 283–298.
[34] A. Blake, A. Zisserman, Visual Reconstruction, The MIT Press, Cambridge, Massachusetts, 1987.
[35] B. Ham, M. Cho, J. Ponce, Robust image filtering using joint static and dynamic guidance, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2015.
[36] Z. Tu, W. Xie, W. Hürst, S. Xiong, Q. Qin, Weighted root mean square approach to select the optimal smoothness parameter of the variational optical flow algorithms, Opt. Eng. 51 (03) (2012) 037202.
[37] Z. Chen, H. Jin, Z. Lin, S. Cohen, Y. Wu, Large displacement optical flow from nearest neighbor fields, in: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2443–2450.
[38] Z. Tu, P. Ronald, R.C. Veltkamp, Weighted local intensity fusion method for variational optical flow estimation, Pattern Rec. 50 (2016) 223–232.
[39] J. Xiao, M. Wan, Y. Wu, Improved adaptive support-weight algorithm based on guide filtering and disparity calibration, in: Proceedings of International Seminar on Computation, Communication and Control, IS3C, 2015.
[40] Z. Tu, C. Gemeren, R.C. Veltkamp, Improved color patch similarity measure based weighted median filter, in: Proceedings of Asian Conference on Computer Vision, 9007, 2015, pp. 413–427.
[41] H. Liu, C. Liu, Y. Tang, H. Sun, X. Li, Spatio-temporal consistency enhancement for disparity sequence, Int. J. Signal Process. Image Process. Pattern Rec. 7 (5) (2014) 229–238.

Zhigang Tu started his M.Phil. Ph.D. in image processing at the School of Electronic Information, Wuhan University, China, 2008. He received a Ph.D. degree in Communication and Information System from Wuhan University, 2013. In November, 2015, he received a Ph.D. degree in Computer Science from the Utrecht University, Netherlands. He is currently a researcher assistant at the School of Computing, Informatics, Decision System Engineering, Arizona State University, US. His research interests include motion estimation, action recognition, object tracking, human-computer interaction.

Ronald Poppe received a Ph.D. degree 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 motion from videos and other sensors, the understanding and modeling of human (communicative) behavior and the applications of both in real-life settings.

Remco C. Veltkamp is full professor of Multimedia at Utrecht University, the 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, with a special focus on game research: http://www.gameresearch.nl/. He has written over 155 refereed papers in reviewed journals and conferences, and supervised 16 Ph.D. theses.