Bilateral Filtering-Based Optical Flow Estimation with Occlusion Detection

Jiangjian Xiao, Hui Cheng, Harpreet Sawhney,
Cen Rao, and Michael Isnardi

Sarnoff Corporation
{jxiao, hcheng, hsawhney, crao, misnardi}@sarnoff.com

Abstract. Using the variational approaches to estimate optical flow between two frames, the flow discontinuities between different motion fields are usually not distinguished even when an anisotropic diffusion operator is applied. In this paper, we propose a multi-cue driven adaptive bilateral filter to regularize the flow computation, which is able to achieve the smoothly varied optical flow field with highly desirable motion discontinuities. First, we separate the traditional one-step variational updating model into a two-step filtering-based updating model. Then, employing our occlusion detector, we reformulate the energy functional of optical flow estimation by explicitly introducing an occlusion term to balance the energy loss due to the occlusion or mismatches. Furthermore, based on the two-step updating framework, a novel multi-cue driven bilateral filter is proposed to substitute the original anisotropic diffusion process, and it is able to adaptively control the diffusion process according to the occlusion detection, image intensity dissimilarity, and motion dissimilarity. After applying our approach on various video sources (movie and TV) in the presence of occlusion, motion blurring, non-rigid deformation, and weak textureness, we generate a spatial-coherent flow field between each pair of input frames and detect more accurate flow discontinuities along the motion boundaries.

1 Introduction

Optical flow estimation has been investigated by computer vision researchers for a long time [10, 12, 19, 3, 4, 11, 1, 6]. Given two input images, how to compute accurate optical flow is still challenging problem in computer vision especially when the images have severe occlusion and non-rigid motion. The basic idea of optical flow computation is maintaining the brightness constancy assumption, which relates the image gradient, ∇I, to the components u and v of the local optical flow. Since this is an ill-posed problem, some additional constraints are required to regularize the motion field during the flow estimation. From the well-known aperture phenomenon, a larger region of integration is more preferable to produce stable motion estimation but it may be more likely contain multiple motions in this region and cannot handle non-rigid deformation very well [4]. Therefore, the fundamental problem of optical flow estimation is still how to design an effective anisotropic smoothness regularizer, such that it not only maintains variable spatial coherence inside each piecewise-smooth region but also keeps accurate flow discontinuities at the motion boundaries.

A. Leonardis, H. Bischof, and A. Pinz (Eds.): ECCV 2006, Part I, LNCS 3951, pp. 211–224, 2006.

Currently, the most popular regularizers of optical flow estimation are the variational-based isotropic and/or anisotropic smoothness operators [10, 8, 2, 1, 6]. However, these techniques have two drawbacks. First, when the input images in the presence of occlusion, these methods cannot correctly handle the flow estimation for the occluded region, and the flow at those occluded regions appears over-smoothing or randomly dragging. Second, if the input images have large homogeneous colored regions, these methods will fail to produce correct flow vector inside those regions due to the poor texture and image gradient field. To overcome these two problems, some researchers propose parametric model or motion segmentation to break the optical flow field into several piecewise-smooth parts [4, 9, 13, 21, 20]. Unfortunately, due to the inherent limitation of the parametric model, these approaches cannot correctly handle the non-rigid scene, where the objects may have irregular deformation.

Aiming to solve those pre-mentioned problems, this paper combines the occlusion detection and an adaptive bilateral filter into a two-step updating variational framework to estimate a high-quality optical flow field between two input frames. In our approach, first we design an occlusion detector to identify the occluded areas, which effectively breaks the spatial coherence over the motion boundaries and makes it possible to produce accurate flow discontinuities. Then, based on this occlusion detector, a novel variational model is proposed where the occlusion detection and occlusion penalty are integrated into the model to explicitly handle the occlusion problem. Third, at the second updating step of the variational model, we substitute the traditional anisotropic filter by our multi-cue driven bilateral filter to deal with the incorrect (or missing) flow estimation of those occlusion regions. As a result, our approach effectively preserve motion discontinuities between the different motion fields and generate smoothly varying motion flow inside each piece of rigid or non-rigid motion field. Furthermore, in this paper we also illustrate the flexibility of integrating more constraints, such as the flow symmetric property, into our framework to compute more accurate optical flow.

The remainder of this paper is organized as follows. Section 2 discusses the existing variational model of optical flow computation and also illustrates how to convert the model into a two-step iteration with a convolution-based diffusion. Based on the new iteration model, Section 3 presents a novel optical flow framework integrated with the explicit occlusion term and a multi-cue driven bilateral filter. In Section 4, we demonstrate several results on various video sources in the presence of occlusion, motion blurring, non-rigid deformation, and weak texture conditions.

2 The Two-Step Variational Updating Model

According to the brightness constancy assumption, given two input images I_1 and I_2, the image brightness of a pixel at $\mathbf{x} = [x\ y]^T$ in I_1 should not be changed by the motion vector $\mathbf{u} = [u\ v]^T$, such that $I_1(\mathbf{x}) = I_2(\mathbf{x} + \mathbf{u})$ [10]. One direct solution of optical flow estimation is to minimize the following quadratic data energy functional over the image domain Ω, such that $E_d(\mathbf{u}) = \int_\Omega \big(I_1(\mathbf{x}) - I_2(\mathbf{x} + \mathbf{u})\big)^2 d\mathbf{x}$. Since this data energy is differentiable, it can be approximated by the first order Taylor expansion

$$E_d(\mathbf{u}) = \int_{\Omega} \left(\nabla I^T \mathbf{u} + I_t\right)^2 d\mathbf{x}, \tag{1}$$

where ∇I is the average gradient of images I_1 and I_2, and I_t is the temporal derivative between I_1 and I_2. In order to avoid the aperture problem and suppress noise during optical flow estimation, a smoothness constraint should be added to regularize the optical flow gradient, $\nabla \mathbf{u}$. The most common smoothness term used in optical flow estimation is the edge-preserving anisotropic operator which can efficiently prevent flow to be smoothed over region boundaries [14, 15, 5, 1, 16]. Therefore, the new overall energy functional for optical flow minimization becomes

$$\begin{aligned} E(\mathbf{u}) = E_d(\mathbf{u}) + E_s(\nabla \mathbf{u}) &= \int_{\Omega} \left(\left(\nabla I^T \mathbf{u} + I_t\right)^2 + \nabla \mathbf{u}^T D(\nabla I_1) \nabla \mathbf{u}\right) d\mathbf{x} \\ &= \int_{\Omega} \left(e_d(\mathbf{u}) + e_s(\nabla \mathbf{u})\right) d\mathbf{x}, \end{aligned} \tag{2}$$

where $e_d(\mathbf{u})$ is a data term corresponding to data energy $E_d(\mathbf{u})$, $e_s(\nabla \mathbf{u})$ is the smoothness term to smoothness energy $E_s(\nabla \mathbf{u})$, ∇I_1 is the image gradient of frame 1, and $D(\nabla I_1)$ is an anisotropic diffusion tensor defined by

$$D(\nabla I_1) = \frac{1}{\|\nabla I_1\|^2 + 2\nu^2} \left(\nabla I_1^{\perp} \nabla I_1^{\perp^T} + \nu^2 \mathbf{1}\right), \tag{3}$$

where $\mathbf{1}$ is a 2×2 identity matrix, ν is a parameter to control the degree of isotropy smoothness, and $\nabla I_1^{\perp}$ is the vector perpendicular to ∇I_1. The diffusion tensor, $D(\nabla I_1)$, has two orthogonal eigenvectors: $\eta = \frac{\nabla I_1}{\|\nabla I_1\|}$ and $\xi = \eta^{\perp} = \frac{\nabla I_1^{\perp}}{\|\nabla I_1\|}$ with corresponding eigenvalues,λ_η and λ_ξ, as shown in Fig.1.

To obtain the minimal energy of Eq.2, we can apply Euler Lagrange equation to iteratively update the flow field $\mathbf{u}$ along the gradient descent direction, such that

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial \tau} = \mathbf{u}^{\tau} - \mathbf{u}^{\tau-1} &= -\left(\frac{\partial e_d(\mathbf{u})}{\partial \mathbf{u}} - \text{div}\left(\frac{\partial e_s(\nabla \mathbf{u})}{\partial \nabla \mathbf{u}}\right)\right) \\ &= -\nabla I\left(\nabla I^T \mathbf{u} + I_t\right) + \text{div}\left(D(\nabla I_1) \nabla \mathbf{u}\right), \end{aligned} \tag{4}$$

where the optical flow $\mathbf{u}^{\tau}$ is the flow field at iteration step τ. From this equation, it is clear to see that since the data and smoothness terms are operating on different domains: $\mathbf{u}$ and $\nabla \mathbf{u}$, these two terms will keep separated after applying Euler Lagrange equation. Therefore, instead of updating $\mathbf{u}^{\tau}$ in one step, we divide the updating process into a two-step procedure, such that

$$\mathbf{u}^{\tau'} - \mathbf{u}^{\tau-1} = -\frac{\partial e_d(\mathbf{u})}{\partial \mathbf{u}} = -\nabla I\left(\nabla I^T \mathbf{u}^{\tau-1} + I_t\right), \tag{5}$$

$$\mathbf{u}^{\tau} - \mathbf{u}^{\tau'} = \text{div}\left(\frac{\partial e_s(\nabla \mathbf{u})}{\partial \nabla \mathbf{u}}\right) = \text{div}\left(D(\nabla I_1) \nabla \mathbf{u}^{\tau'}\right), \tag{6}$$

where the first step is updating the flow field to an intermediate result, $\mathbf{u}^{\tau'}$, by minimizing the data energy, and the second step is preforming an independent diffusion process

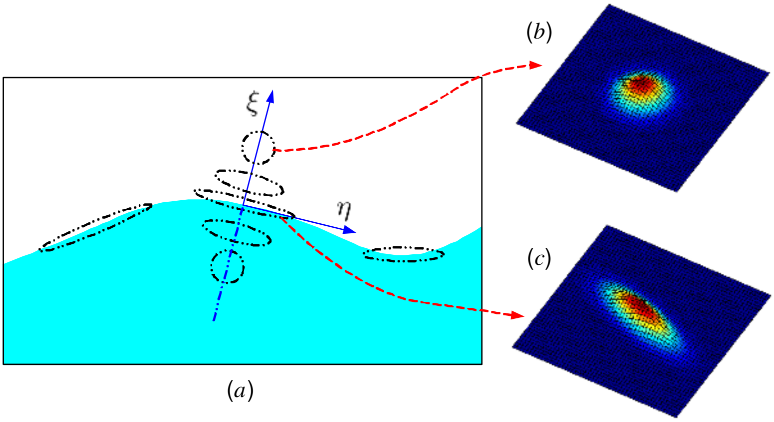


Fig. 1. (a) Given an image with two distinguished regions, one pair of eigenvectors, η and ξ, are shown for a pixel located at the region boundary. Depending on the diffusion tensor $\mathbf{T}$, the shape and size of the Gaussian kernel are varying at different locations. (b) The isotropic Gaussian kernel at the homogeneous region. (c) The anisotropic oriented Gaussian kernel at the region boundary.

on the estimated motion field $\mathbf{u}^{\tau'}$. One interesting point of this separation is that if we construct a structure tensor $\mathbf{T} = \lambda_\eta \eta\eta^T + \lambda_\xi \xi\xi^T$ and let $\mathbf{H} = \begin{bmatrix} \mathbf{u}^{\tau'}_{xx} & \mathbf{u}^{\tau'}_{xy} \\ \mathbf{u}^{\tau'}_{yx} & \mathbf{u}^{\tau'}_{yy} \end{bmatrix}$, then Eq.6 can be rewritten as $\mathbf{u}^{\tau} - \mathbf{u}^{\tau'} = \text{trace}(\mathbf{TH})$, and this diffusion equation can be further replaced by a 2D oriented Gaussian convolution [18], such that

$$\mathbf{u}^{\tau} = \mathbf{u}^{\tau'} * G(\mathbf{T}, \varDelta\tau), \quad \text{where } \; G(\mathbf{T}, \varDelta\tau) = \frac{1}{4\pi\varDelta\tau} \exp\left(- \frac{\mathbf{x}^T \mathbf{T^{-1} x}}{4\varDelta\tau} \right), \tag{7}$$

and $\varDelta\tau$ is the step length of iteration. If $\varDelta\tau$ is set to more than 1, the size of the oriented Gaussian kernel becomes large and the diffusion process would be speeded up. Fig.1.a shows the variation of the Gaussian kernel at different locations due to its varied structure tensor, $\mathbf{T}$. Notice that the radii of the oriented Gaussian kernel also depend on the eigenvalues of $\mathbf{T}^{-1}$, which are $\frac{1}{\lambda_\eta}$ and $\frac{1}{\lambda_\xi}$. When the pixel $\mathbf{x}$ is located at the interior of a smooth region, $\|\nabla I_1\|$ is small and $\lambda_\xi \simeq \lambda_\eta \simeq \frac{1}{2}$, which is equivalent to applying an isotropic Gaussian kernel for the smoothing as shown in Fig.1.b. If the pixel is located at the sharp boundary between two segments, $\|\nabla I_1\|$ will be large and $\lambda_\xi \simeq 1 \gg \lambda_\eta \simeq 0$, which is equivalent to applying an oriented Gaussian kernel on the images as shown in Fig.1.c.

After separating the updating procedure into two steps, another interesting point is that we can substitute the original diffusion tensor by a more powerful, convolution-based diffusion filter in this variational framework, and this new filter may not be implemented by the traditional PDE iteration. Based on this motivation, the next section will show how to integrate a powerful, convolution-based bilateral filter into the flow estimation framework to achieve highly discontinuous flow field from two input images.

3 Highly Discontinuity-Persevering Optical Flow Estimation with Occlusion Detection

Even with the anisotropic diffusion term in the energy minimization function, the previous work still has difficulties to obtain highly discontinuous flow field due to the unclear occlusion process [19, 8, 1, 6, 16]. In [1, 16], the authors all point out that occlusion detection is critical for the motion estimation especially when the motion gap is large. However, the quality of occlusion detection and optical flow estimation at occluded regions are unsatisfactory in these papers due to the lack of the elaborate occlusion handling. In this section, we first exploit the natural property of the occlusion between two frames, and then provide an occlusion detector to identify the occlusion area. Based on the occlusion analysis, an explicit occlusion term is introduced into the variational framework to balance the data and occlusion energy. Furthermore, we substitute the traditional anisotropic diffusion tensor in the variational framework by a more flexible, multi-cue driven bilinear filter to preform more effective occlusion handling and produce more accurate optical flow field.

3.1 Occlusion Analysis and Detection

Fig.2 illustrates two kinds of occlusion happening in optical flow estimation. The first case is motion occlusion, where the occlusion generation is due to object motion and the occluded areas from two frames are not overlapped at the same location. The second case is mismatching where the occluded regions from different images are overlapped at the same position. The mismatching may happen under different conditions, such as object appearing/disappearing, shadow, color change, or large object deformation (shrinking or expanding), etc.

To detect such occlusion, one way is checking the consistency between the forward and backward flow. If the backward and forward flow is constant, the pixel will be

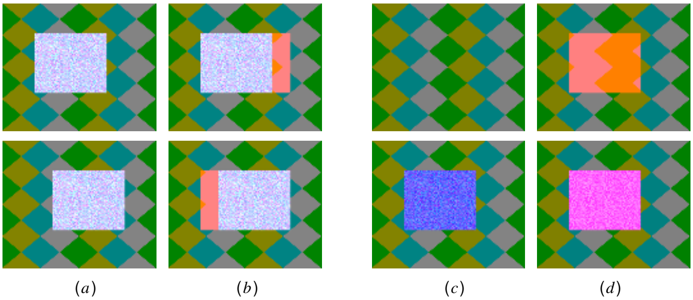


Fig. 2. (a) The case of the motion occlusion, where a rectangle is moving from the left (the top frame) to the right side (the bottom frame). (b) The corresponding occluded areas of (a) are masked in red and the occluded areas locate at different positions due to the object's motion. (c) The case of mismatching, where the top is the first frame and a rectangle suddenly appears in the second frame (the bottom one). (d) The corresponding occluded areas of (c) are also masked in red, but in this case these occluded regions are overlapped at the same location.

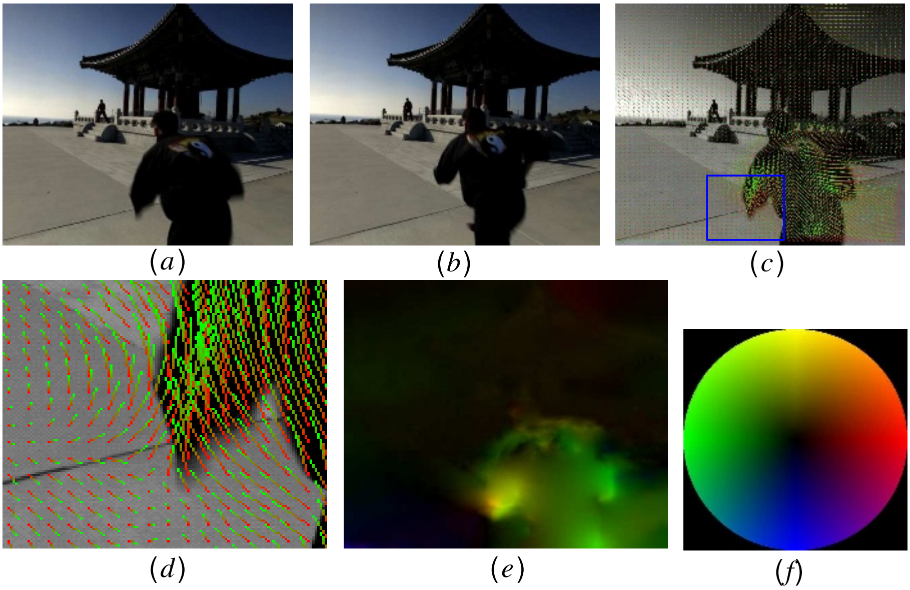


Fig. 3. (a) The first input frame. (b) The second input frame. (c) The estimated optical flow using the traditional variational approach, where the flow of the weak-textured regions are dragged by the high gradient region boundaries (*Note: for comparison, please refer Fig.5.b*). (d) The zoomed image from the blue box in (c). (e) The dense flow field shown in color coded fashion where it is easy to see the dragging around the high gradient boundaries. (f) The color code map where the color represents the orientation of the vector and brightness stands for its magnitude. *Note: in (c) and (d), we also draw the flow vector using a line segment which starts from red and ends at green.*

considered as non-occluded [1]. However, this forward-backward matching may not be reliable for some cases, such as mismatching where the flow inside the both overlapping occluded regions may be zero as shown in Fig.2.$c - d$. As a result, this detector will not detect the error from forward and backward flow and it will calm such regions as non-occluded, which is contradictory to our analysis. In order to avoid such missing detection, we propose a simple but robust solution to detect the occlusion for the both cases by employing the squared image residue as

$$\rho(\mathbf{u}) = \begin{cases} 0 & \text{if } \big(I_1(\mathbf{x}) - I_2(\mathbf{x}+\mathbf{u})\big)^2 > \epsilon_I \\ 1 & \text{otherwise.} \end{cases} \tag{8}$$

where ϵ_I is a threshold to decide the occlusion, $\rho = 0$ means the pixel is occluded, and $\rho = 1$ denotes this pixel is visible in the both frames. To obtain a continuous function of $\rho(\mathbf{u})$ for PDE differentiation, a numerical approximation of the Heaviside function is used, such that

$$\rho(\mathbf{u}) = \frac{1}{2} + \frac{1}{\pi}\tan^{-1}\Big(\big(I_1(\mathbf{x}) - I_2(\mathbf{x}+\mathbf{u})\big)^2 - \epsilon_I\Big). \tag{9}$$

3.2 Energy Model with Occlusion Detection

One mishandling in the current variational model is trying to minimize the squared intensity error or data energy for every pixel regardless if the pixel is occluded or not. As a result, the warped image, $I_2(\mathbf{x}+\mathbf{u})$, has to perform incorrect deformation to fill the occluded area of frame $I_1(\mathbf{x})$ even though no corresponding pixel at I_2 can match the occluded pixel $\mathbf{x}$ at the first frame.

Fig.3 shows one example when a large occlusion between two images, this minimization will produce some serious distortion or dragging. In this example, there is a large motion difference between non-rigid foreground and the rigid background. Using the traditional framework, the weak-textured regions would be dragged to follow the movement of the high-gradient region boundaries. Another possible common case is when camera has apparent zooming or pan, a larger number of pixels should be occluded at the image boundary. If without correct occlusion handling, the energy of those pixels will be minimized to cause the serious distortion along the image boundary.

To fix these problems, we need to exclude the occluded pixels from the minimization process and add a corresponding penalty into the energy functional to balance occlusion and visibility. Therefore, our new energy model can be written as

$$E(\mathbf{u}) = \big(E_d(\mathbf{u}) + E_s(\nabla\mathbf{u})\big)\cdot\rho(\mathbf{u}) + \big({E_d}^{oc} + {E_s}^{oc}(\nabla\mathbf{u})\big)\cdot\big(1-\rho(\mathbf{u})\big), \qquad (10)$$

where the first part of this equation is dealing with the energy of the non-occluded pixels and it includes two components, E_d and E_s, which correspond to the conventional data and smoothness energy similar to the model in the previous section. The second part of the equation is handling the energy of the occluded pixels, where ${E_d}^{oc}$ is occlusion energy and ${E_s}^{oc}$ is the smooth regulation for the occluded pixels. If the smooth processing of E_s and ${E_s}^{oc}$ are same, we can merge these two terms into one, such that

$$\begin{aligned} E(\mathbf{u}) &= \big(E_d(\mathbf{u}) - {E_d}^{oc}\big)\cdot\rho(\mathbf{u}) + {E_d}^{oc} + E_s(\mathbf{u}), \\ &= \int_{\Omega}\Big(\big(e_d(\mathbf{u}) - {e_d}^{oc}\big)\cdot\rho(\mathbf{u}) + {e_d}^{oc} + e_s(\nabla\mathbf{u})\Big)d\mathbf{x}, \end{aligned} \qquad (11)$$

where ${e_d}^{oc}$ is a constant occlusion penalty corresponding to the occlusion energy ${E_d}^{oc}$, $e_d(\mathbf{u})$ and $e_s(\nabla\mathbf{u})$ are data and smoothness terms same as Eq.2. From this equation, it is obvious when the occlusion penalty ${e_d}^{oc}$ increase, the occlusion detection will become more difficult and less pixels will be claimed as occluded. Therefore, a proper occlusion penalty will balance energies between the occlusion and data terms, and correctly locate the occlusion regions. In our experiment, we set ${e_d}^{oc} = \epsilon_I$, same as the occlusion detector threshold in Section 3.1.

Then, after applying Euler Lagrange equation, we can update the flow field by the two-step updating scheme as (Eq.5-7) becomes

$$\mathbf{u}^{\tau'} - \mathbf{u}^{\tau-1} = -\frac{\partial e_d(\mathbf{u})}{\partial\mathbf{u}}\rho(\mathbf{u}) - \big(e_d(\mathbf{u}) - e_d^{oc}\big)\frac{\partial\rho(\mathbf{u})}{\partial\mathbf{u}}, \qquad (12)$$

$$\mathbf{u}^{\tau} - \mathbf{u}^{\tau'} = \mathrm{div}\bigg(\frac{\partial e_s(\nabla\mathbf{u})}{\partial\nabla\mathbf{u}}\bigg) \quad \text{or} \quad \mathbf{u}^{\tau} = \mathbf{u}^{\tau'} * G(\mathbf{T}, \Delta\tau), \qquad (13)$$

where the first step is updating the flow field only based on the data and occlusion penalty, and the second step is performing diffusion process to suppress the noise and propagate the flow to non-textured region by either PDE updating or Gaussian convolution.

3.3 Occlusion Diffusion Using Multi-cue Driven Adaptive Bilateral Filter

Theoretically, the pixels at the occlusion area should not be assigned any flow vector since there is no correspondence available in the other frame. Nevertheless, in practice, the occluded pixels will be associated with certain motion flow by the diffusion operation in the variational model, and therefore the estimated flow at these areas will heavily depend on the diffusion process. Unfortunately, using the current variational-based anisotropic diffusion or oriented Gaussian smoothing, the diffusion process lacks the occlusion handling mechanism and also cannot distinguish the flow influence from different regions very well, which may produces serious distortion at the region boundaries.

Fig.4.a shows two kinds of mishandling of the current anisotropic diffusion on a simple example, where the cyan box is moving from the left to the right and the red region is occluded region similar to the Fig.2.b (To save space, we only show the first frame). In the first non-occluded case at pixel $\mathbf{x}_1$, an oriented Gaussian kernel is generated to perform diffusion process based on the diffusion tensor $D(\nabla I_1)$. Even though this Gaussian kernel is stretched along the region boundary, the diffusion process will still convolute with a certain of flow information from the dissimilar regions to estimate its flow vector. Hence, the flow influence from the cyan region may dramatically distort the flow field in the white background region. In the second case, the pixel, $\mathbf{x}_2$, is located at the occluded region, similarly an oriented Gaussian kernel is generated as shown in Fig.4.a. However, if the occlusion gap is large, the radius of the oriented Gaussian kernel may not be possible to cover the size of occlusion area. Therefore, the only information convoluted for the flow estimation of pixel $\mathbf{x}_2$ is from the unreliable occluded region.

Therefore, in order to overcome these two mishandling, we need to redesign the diffusion process which can adaptively change the diffusion kernel's size and shape to minimize the flow influence from the inconsistent regions. In this section, we present an adaptive, multi-cue driven bilateral filter to block such incorrect flow influence between different regions and simultaneously infer the motion flow for the occluded regions from the surrounding non-occluded pixels. In Fig.4.b, one possible solution of Fig.4.a is given. In the both cases, the kernel size is adaptively changed and the kernel shape is truncated into two parts according to the occlusion detection and image intensity. The first part of these kernels is the support region marked as green where the motion information inside this region is used to estimate the flow vector for pixels $\mathbf{x}_i$. The remaining part of the kernels is the unsupport region and its information is discarded or reduced by certain weights during the flow estimation.

The original bilateral filter is introduced by Tomasi and Manduchi to preform a nonlinear diffusion on image restoration [17], where two Gaussian kernels are stacked together such that

$$I'(\mathbf{x}_1) = \frac{1}{k(\mathbf{x}_1)} \int_{\Omega} I(\mathbf{x}) \cdot g_s(\mathbf{x} - \mathbf{x}_1) \cdot g_I\big(I(\mathbf{x}) - I(\mathbf{x}_1)\big) d\mathbf{x}, \tag{14}$$

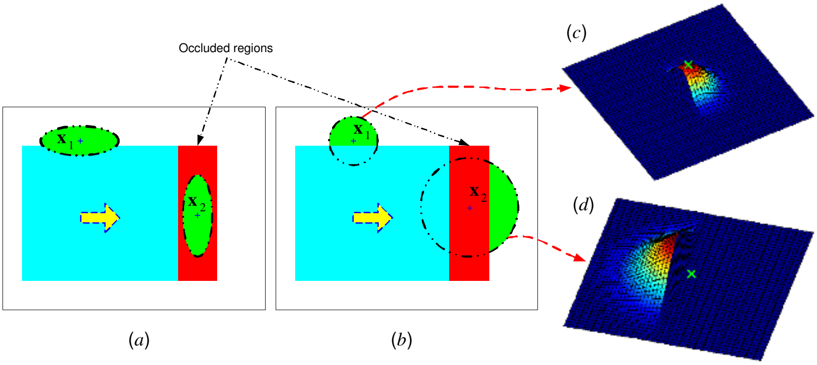


Fig. 4. Comparison between the variational-based anisotropic diffusion and our adaptive bilateral filter. Here the cyan box is moving from the left to the right side as indicated by the big yellow arrow. The red region is occluded region similar as the Fig.2.b (here we only show the first frame). (a) Two kinds of Mishandling for the pixel located near the region boundary and occluded area by using the variational-based anisotropic diffusion. (b) Employing our adaptive bilateral filter, the shape and size of the Gaussian kernel are adaptively changed for different cases and the optical flow is correctly estimated for the both cases. *Note: only the green area is used for the diffusion.* (c) and (d) are 3D visualization of the bilateral filter kernels where a green cross is marked at the kernel center.

where the normalize term $k(\mathbf{x}_1) = \int_\Omega g_s(\mathbf{x} - \mathbf{x}_1) \cdot g_I\big(I(\mathbf{x}) - I(\mathbf{x}_1)\big)d\mathbf{x}$, $I'(\mathbf{x}_1)$ is the output of the bilateral filter for pixel $\mathbf{x}_1$, $g_s(\cdot)$ and $g_I(\cdot)$ are two Gaussian functions for spatial and intensity domains respectively. Using the function $g_I(\cdot)$, the influence of the intensity-dissimilar pixels are effectively reduced. One can simplify Eq.14 by a convolution format such that

$$I' = I * G_s(\mathbf{x}, \sigma_s) * G_I(I, \sigma_I), \tag{15}$$

where $G_s(\sigma_s)$ is a Gaussian kernel on spatial domain $\mathbf{x}$ with variance σ_s, which corresponds to $g_s(\mathbf{x} - \mathbf{x}_1)$ of Eq.14. $G_I(\sigma_I)$ is another Gaussian kernel on intensity domain I with variance σ_I, which corresponds to $g_I\big(I(\mathbf{x}) - I(\mathbf{x}_1)\big)$ of Eq.14.

In our two-step optical flow estimation model, since the diffusion process is explicitly separated from the motion estimation step, we can simply substitute the oriented Gaussian filter in Eq.13 by our adaptive bilateral filter, such that

$$\mathbf{u}^\tau = \mathbf{u}^{\tau'} * G_s\big(\mathbf{x}, \sigma_s(\rho, \chi)\big) * G_I(I, \sigma_I) * G_\mathbf{u}(\mathbf{u}, \sigma_\mathbf{u}) * \rho. \tag{16}$$

Compared to the original bilateral filter (Eq.15), two additional convolution function are added. One is the occlusion function, ρ, which can fully disable the influence of the occluded region during the diffusion process. The other is a one dimensional Gaussian kernel, $G_\mathbf{u}$, to reduce the influence based on motion dissimilarity. Moreover, we also modify the spatial Gaussian kernel, G_s, which is able to adaptively change the kernel size by the occlusion function ρ and a varied occlusion region radius, χ, such that

$$\sigma_s(\rho, \pi)) = \begin{cases} \sigma_0 & \text{if } \rho = 1 \\ \sigma_0 + \frac{\chi}{3} & \text{if } \rho = 0 \end{cases}, \tag{17}$$

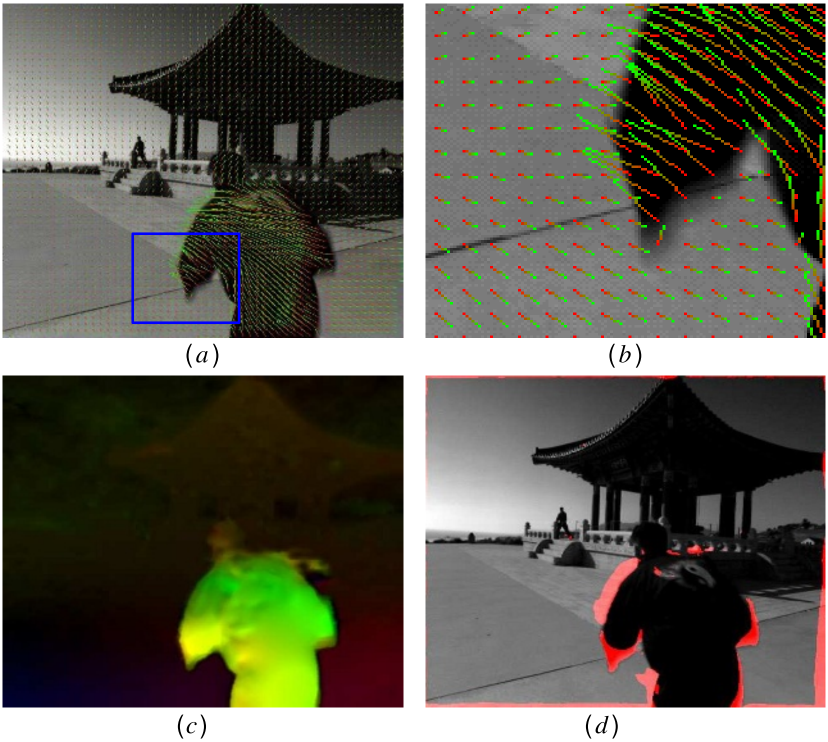


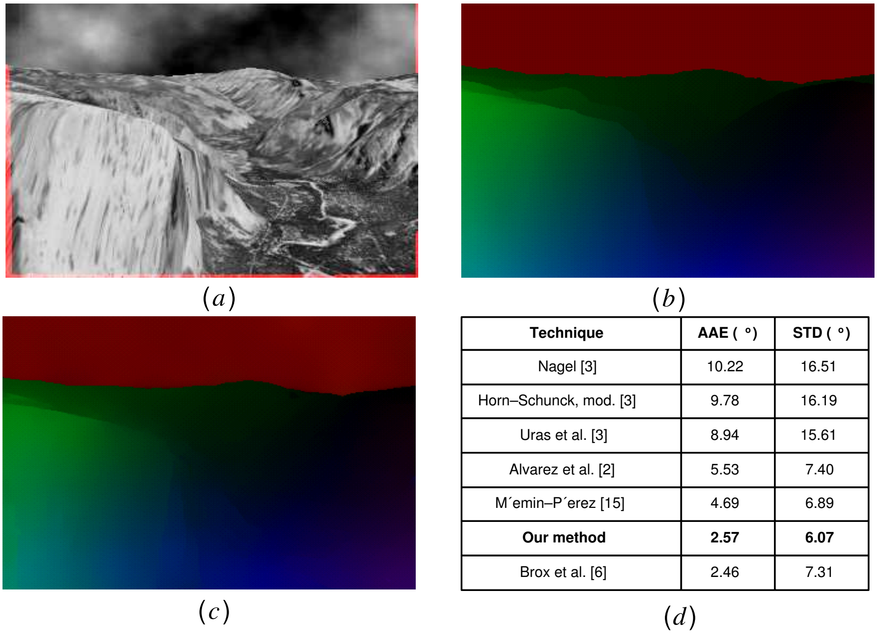
Fig. 5. (a) The estimated optical flow of Fig.3 using our approach, where the flow of the weak-textured regions are not dragged by the high gradient region boundaries any more. (b) The zoomed image from the blue box in (a). Compared to Fig.3.d, the flow vectors at the background region are not dragged by high gradient boundary any more. (c) Dense flow field. (d) The occluded areas in frame 1 (red regions).

where σ_0 is a default value of the kernel variance. When $\rho = 1$, the pixel is located at the non-occluded area where the estimated flow is reliable. With the convolution of the intensity kernel G_I and motion kernel $G_{\mathbf{u}}$, a small Gaussian kernel with $\sigma_s = \sigma_0$ is applied to preform diffusion as shown at position $\mathbf{x}_1$ in Fig.4.b, and the influence from the dissimilar pixels are efficiently reduced by G_I and $G_{\mathbf{u}}$. When $\rho = 0$, the pixel is occluded and the kernel size is increased by an additional term, $\frac{\chi}{3}$, where χ is an occlusion region radius function and it is pre-computed for each pixel after the occlusion detection step. With this new term, we can guarantee the radius of spatial kernel is always larger than the radius of the occluded region. Then employing the convolution of function $G_I * G_{\mathbf{u}} * \rho$, the flow influence from the unreliable occluded region is disabled, and the influence from the other dissimilar regions is also reduced according to the intensity and motion similarities. As a result, our adaptive bilateral filter can effectively collect the flow influence from the non-occluded, intensity and motion similar, surrounding regions to estimate correct flow vector for the occlude pixel as shown at position $\mathbf{x}_2$ in Fig.4.b.

Fig.5 shows the estimated flow field between frame Fig.3.a and 3.b by using our approach. Compared to the previous results in Fig.3.$c - e$, our approach correctly detects the occluded regions and effectively excludes these occluded pixels from the data minimization process to avoid the undesirable background dragging. Then, with the multi-cue bilateral filter, the motion flow for these occluded regions are inferred from the surrounding non-occluded pixels. As a result, the sharp motion discontinuities are obtained between different flow fields, and the non-rigid, continuous flow inside each flow fields are maintained as well.

4 Experiments and Evaluation

In the case of the optical flow is more than one pixel, a multi-scale pyramid [7] is necessary to be applied to avoid the minimization process trapped into a local minimum. After creating pyramids for two input reference frames, we start from the top level and iteratively update the flow field in two steps: first we estimate the flow vectors between the reference frame and the corresponding warped frame, then an adaptive bilateral diffusion process (Eq.16) is applied to correct the flow field and suppress the noise.



(a)

(b)

(c)

Technique	AAE (°)	STD (°)
Nagel [3]	10.22	16.51
Horn–Schunck, mod. [3]	9.78	16.19
Uras et al. [3]	8.94	15.61
Alvarez et al. [2]	5.53	7.40
M´emin–P´erez [15]	4.69	6.89
Our method	**2.57**	**6.07**
Brox et al. [6]	2.46	7.31

(d)

Fig. 6. (a) One frame from the Yosemite sequence with clouds. The occluded regions are masked in red, which hasn't been done in the literature. (b) The corresponding dense flow field of the ground truth. (c) Dense flow field of our result. (d) Comparison with the results from the literature with 100% density for the Yosemite sequence with clouds. AAE denotes average angular error and STD denotes standard deviation.

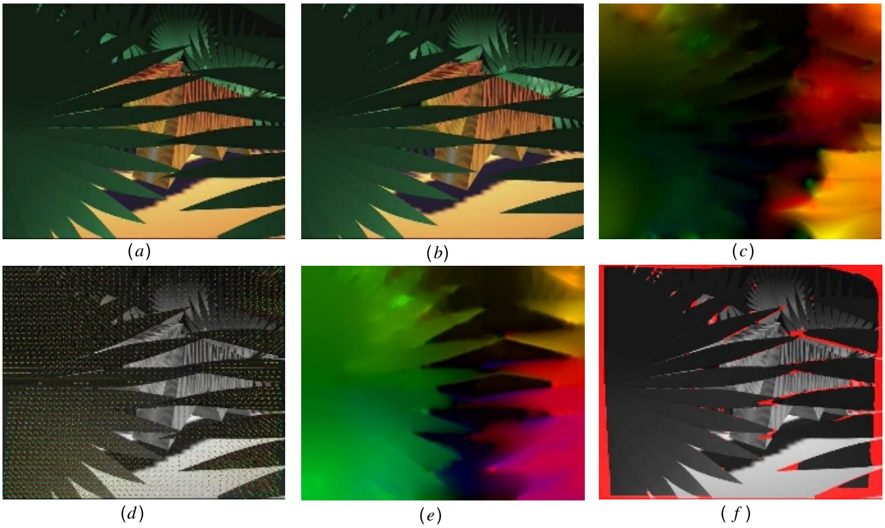


Fig. 7. (a) The first frame. (b) The second frame. (c) Dense flow field using the traditional approach. (d) The estimated optical flow in the first frame using our approach. (e) Dense flow field using our approach. (f) The occluded areas in frame 1 (red regions).

In order to evaluate our algorithm, we test our method on the synthetic data which has the ground truth. In Fig.6, we show our results for the well-known Yosemite with clouds sequence, and also compare them to the results from the literature. From the table of Fig.6.d, our results are slightly worse than the current best results [6] in this small motion case, but outperform the rest algorithms. The average computation time of this sequence is 4.03 sec/frame at 3.6GHz Intel Xeon CPU. For Yosemite sequence without clouds, the average angular error of our results is $1.57°$ with 100% density, which is also comparable to the most state-of-arts algorithms [3, 11, 6].

Beside this, we also test our algorithm in different real videos from movie or TV. In these videos, some non-rigid objects have serious deformation and large displacement of the moving objects produce severe occlusion and motion blurring as shown in Fig.7-8. Fig.7.c shows two frames from one cartoon video, "Tiger", where the leaves have large motion along different directions and some parts of the scene without texture. Using the traditional approach, the flow vectors of the background are dragged with the high-gradient boundaries and the motion discontinuities are not preserved very well along the leave boundaries as shown in Fig.7.c. In our results (Fig.7.$d-f$), we correctly detect the boundary occlusion and achieve more accurate motion discontinuities between the leaves and background regions. In Fig.8, we also show one result from Football TV. The first two images are the input frames. Our results (Fig.8.$d-f$) is apparently better than the traditional optical flow algorithm. Using our approach, we obtain more accurate and highly contrast motion discontinuities for this non-rigid, fast motion sequence with irregular occlusions.

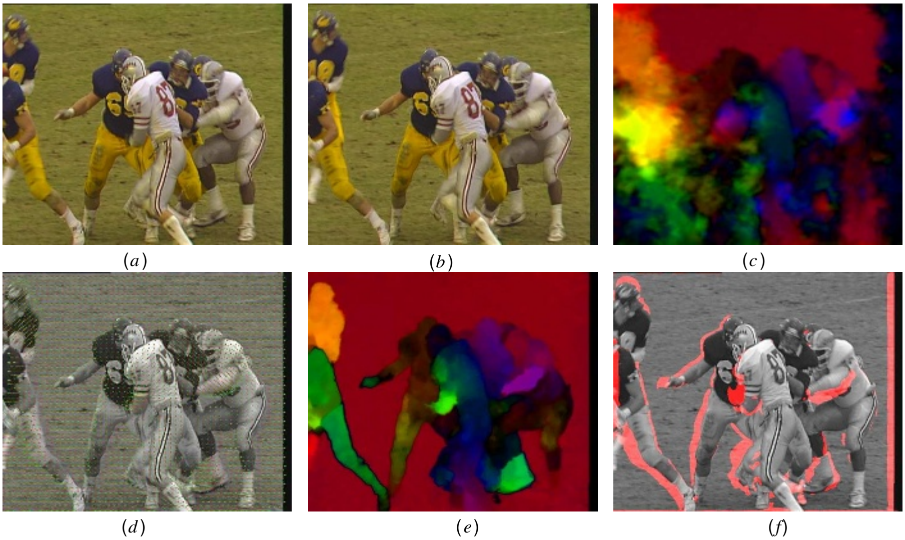


Fig. 8. (*a*) The first frame. (*b*) The second frame. (*c*) Dense flow field using the traditional approach. (*d*) The estimated optical flow in the first frame using our approach. (*e*) Dense flow field using our approach. (*f*) The occluded areas in frame 1 (red regions).

5 Conclusion

In this paper, we present a novel variational-based framework to compute the optical flow for the video sequence in the presence of large occlusion and non-rigid motion. Our main contributions consist of: (1) We explicitly introduce an occlusion term into variational model to balance the data energy with occlusion handling process. (2) We initialize a two-step updating model for optical flow estimation, and further seamlessly integrate it with our multi-cue driven bilateral diffusion process to solve the occlusion mishandling of the previous approaches. Using our approach, the occluded regions are explicitly excluded from the optical flow computation, and our bilateral diffusion effectively infer the flow vectors for the occluded regions. After applying our approach on various video sources, the experiments show that our method can maintain piecewise spatial-coherent flow field for the rigid or non-rigid objects and also preserve accurate flow discontinuities along the motion boundaries simultaneously.

References

1. Alvarez, L., Deriche, R., Papadopoulo, T., Sanchez, J.: Symmetrical dense optical flow estimation with occlusion detection. In: European Conference on Computer Vision, Springer (2002) 721–735
2. Alvarez, L., Weickert, J., Sanchez, J.: Reliable estimation of dense optical flow fields with large displacements. International Journal of Computer Vision **39** (2000) 41–56
3. Barron, J., Fleet, D., Beauchemin, S.: Performance of optical flow techniques. International Journal of Computer Vision **12** (1994) 43–77

4. Black, M., Anandan, P.: The robust estimation of multiple motions: parametric and piecewise smooth flow fields. Computer Vision and Image Understanding **63** (1996) 75–104
5. Black, M., Sapiro, G., Marimont, D., Heeger, D.: Bobust anisotropic diffusion. IEEE Trans. on Image Processing **7** (1998) 421–432
6. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: European Conference on Computer Vision, Springer (2004) 25–36
7. Burt, P., Adelson, E.: The laplacian pyramid as a compact image code. IEEE Trans. on Communications **31** (1983) 532–540
8. Deriche, R., Kornprobst, P., Aubert, G.: Optical-flow estimation while preserving its discontinuities: a variational approach. In: Asian Conference on Computer Vision. (1995) 290–295
9. Farneback, G.: Very high accuracy velocity estimation using orientation tensors, parametric motion, and simultaneous segmentation of the motion field. In: International Conference on Computer Vision. (2001) 171–177
10. Horn, B., Schunck, B.: Determining optical flow. Artificial Intelligence **17** (1981) 185–203
11. Ju, S., Black, M., Jepson, A.: Skin and bones: Multi-layer, locally affine, optical flow, and regularization with transparency. Computer Vision and Pattern Recognition (1996) 307–314
12. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: International Joint Conference on Artificial Intelligence. (1981) 674–679
13. Memin, E., Perez, P.: Hierarchical estimation and segmentation of dense motion fields. International Journal of Computer Vision **46** (2002) 129–155
14. Nagel, H., Enkelmann, W.: An inverstigation of smoothness constraints for the estimation of displacement vector fields from image sequences. IEEE Trans. on Pattern Analysis and Machine Intelligence **8** (1986) 565–593
15. Pernoa, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE Trans. on Pattern Analysis and Machine Intelligence **12** (1990) 629–639
16. Strecha, C., Fransens, R., Van Gool, L.: A probabilistic approach to large displacement optical flow and occlusion detection. In: Workshop on Statistical Methods in Video Processing. (2004) 71–82
17. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: International Conference on Computer Vision. (1998) 839–846
18. Tschumperle, D., Deriche, R.: Vector-valued image regularization with pde's: A common framework for different applications. In: Computer Vision and Pattern Recognition. (2003) 651–656
19. Weber, J., Marlik, J.: Bobust computation of optical flow in a multi-scale differential framework. International Journal of Computer Vision **2** (1994) 5–19
20. Xiao, J., Shah, M.: Accurate motion layer segmentation and matting. In: Computer Vision and Pattern Recognition. (2005) 698–703
21. Xiao, J., Shah, M.: Motion layer extraction in the presence of occlusion using graph cuts. IEEE Trans. on Pattern Analysis and Machine Intelligence **27** (2005) 1644–1659