

Piecewise-Smooth Dense Optical Flow via Level Sets

T. Amiaz and N. Kiryati

School of Electrical Engineering
Tel Aviv University
Tel Aviv 69978, Israel

Abstract

We propose a new algorithm for dense optical flow computation. Dense optical flow schemes are challenged by the presence of motion discontinuities. In state of the art optical flow methods, over-smoothing of flow discontinuities accounts for most of the error. A breakthrough in the performance of optical flow computation has recently been achieved by Brox *et al.* Our algorithm embeds their functional within a two phase active contour segmentation framework. Piecewise-smooth flow fields are accommodated and flow boundaries are crisp. Experimental results show the superiority of our algorithm with respect to alternative techniques. We also study a special case of optical flow computation, in which the camera is static. In this case we utilize a known background image to separate the moving elements in the sequence from the static elements. Tests with challenging real world sequences demonstrate the performance gains made possible by incorporating the static camera assumption in our algorithm.

1 Introduction

Optical flow is the apparent motion in an image sequence, observable through intensity variations [1]. Although optical flow is generally not equivalent to the true motion field, they are quite similar in typical cases. Optical flow computation is a key step in many image and video analysis applications, including tracking, surveillance, dynamic super resolution and shape from motion.

Various approaches to optical flow computation have been suggested [2]. These include differential techniques [3, 4], phase based [5] and statistical methods [6]. Variational optical flow computation is a subclass of differential techniques, that use the calculus of variations to

minimize a functional embodying the constancy of features in the image and smoothness of the resulting flow. Brox *et al* [7] have significantly improved upon the original work of Horn and Schunck [3], by incorporating coarse-to-fine strategies [8, 9, 10], robust statistics [1, 9] and gradient constancy in a nonlinear objective functional.

Dense optical flow methods, i.e. algorithms that estimate the optical flow everywhere in the frame, tend to fail along strong discontinuities in the optical flow field. Even with the algorithm of Brox *et al* [7], that employs a robust smoothness term, most of the error develops there. Coupling optical flow estimation with segmentation offers better support for flow discontinuities [11, 10].

In other approaches, the precise location of the boundary is considered to be of higher importance than the overall accuracy of the optical flow field. Wang and Adelson [12] decomposed the image sequence into layers of affine motion with successively decreasing fidelity. Cremers and Soatto [13] segmented the sequence into regions of affine motion bounded by evolving contours, including an explicit spline based implementation, and two-phase and multi-phase level set implementation. However, these methods focus on the segmentation of the image, and their assumption of motion affinity limits their precision as optical flow computation methods.

In this paper, we embed the functional of [7] within an active contour segmentation framework. The goal is to profit from the excellent performance of that method, while producing crisp flow boundaries. Following [11, 10, 13] we restate the optical flow problem as that of determining piecewise smooth flow fields bounded by contours, simultaneously minimizing the length of the contours and the optical flow functional of [7] inside the smooth regions.

Due to the presence of unspecified discontinuities in the integration domain, minimiza-

tion of Mumford-Shah [14] type functionals is difficult. Notable solution approaches include Ambrosio and Tortorelli’s Γ -convergence technique [15], and Vese and Chan’s level set method [16]. We follow the latter, because it explicitly limits the thickness of the boundary. Experimental testing demonstrates the accuracy of our piecewise-smooth optical flow algorithm.

A special case of the optical flow problem is computing optical flow for a sequence known to be taken by a static camera. This additional information can be used in several ways, such as explicitly forcing homogeneous areas to zero movement [1], or using a long sequence for background subtraction in a tracking context [17]. We propose using this information and a background image to aid in classifying the pixels as moving or being part of the static background.

The conceptual roots of our approach are described in section 2. The proposed piecewise-smooth flow algorithm is derived in section 3. The experimental results, presented in section 4, demonstrate significant performance improvement with respect to the best published methods. In section 5, the static camera case is formulated and the performance boost made possible by incorporating the static camera constraint is demonstrated. Our preliminary results were presented in [18].

2 Background

Our work relies on the superior performance of the algorithm of Brox *et al* [7] in finding smooth optical flow fields, the Mumford-Shah [14] segmentation framework, and its numerical approximation by Vese and Chan [16].

2.1 The optical flow functional of Brox *et al* [7]

Optical flow computation is an underconstrained problem. The introduction of assumptions or a-priori knowledge is necessary for reliable estimation of optical flow. Brox et al's optical flow functional represents the following principles and constraints:

Grey level constancy:

$$I(\vec{x}) = I(\vec{x} + \vec{w}) \quad (1)$$

Here I is the image luminance as a function of spatial position and time, $\vec{x} = (x, y, t)$ is the spatio-temporal position in the image and $\vec{w} = (u, v, 1)$ is the optical flow displacement field between the two frames.

Gradient constancy: An extension of the grey level constancy assumption, that accommodates illumination changes between the two images.

$$\nabla I(\vec{x}) = \nabla I(\vec{x} + \vec{w}) \quad (2)$$

Smoothness: The minimization of an optical flow functional relying only on the image constancy assumptions is ill posed. A constraint on the spatio-temporal smoothness of the flow field provides regularization. There are two versions of smoothness presented in [7], the 3D version uses spatio-temporal smoothness, while the 2D version uses purely spatial smoothness. Note that in our implementation of the algorithm we used the 2D version.

Presence of outliers: Outliers appear in the flow because of image noise and sharp edges. They are handled in the functional by using robust fidelity and smoothness terms.

Incorporating the grey-level and gradient constancy assumptions into a modified L^1 norm

$\Psi(x^2) = \sqrt{x^2 + \epsilon^2}$, the following fidelity (data) term is obtained:

$$f_{of}(u, v) = \Psi(|I(\vec{x} + \vec{w}) - I(\vec{x})|^2 + \gamma|\nabla I(\vec{x} + \vec{w}) - \nabla I(\vec{x})|^2) \quad (3)$$

Here γ is the weight of gradient constancy with respect to grey level constancy. Using the same robust kernel for the smoothness term

$$s_{of}(u, v) = \Psi(|\nabla u|^2 + |\nabla v|^2) \quad (4)$$

results in the optical flow functional

$$F_{of}(u, v) = \int f_{of}(u, v) + \alpha s_{of}(u, v) d\vec{x} , \quad (5)$$

where α is a weight coefficient.

Minimizing this functional is based on solving the Euler-Lagrange equations. The equations are highly non-linear and are linearized using nested fixed point iterations on a coarse-to-fine grid.

2.2 Mumford-Shah segmentation

The Mumford-Shah formulation is based on the observation that natural images are piecewise smooth. Defining the fidelity term

$$f(I) = (I - I_0)^2$$

that compares the segmentation field I with the original image I_0 , and the smoothness term

$$s(I) = |\nabla I|^2$$

leads to the well known Mumford-Shah functional [14]:

$$F_{MS} = \int_{\Omega} f(I) d\vec{x} + \mu \int_{\Omega \setminus C} s(I) d\vec{x} + \nu |C| \quad (6)$$

Here Ω is the domain of the image, C is the discontinuity (edge) set, $|C|$ represents edge length, and μ, ν are the relative weights of the smoothness and edge-length terms. This corresponds to requiring good approximation of the original image by a piecewise smooth segmentation field with few discontinuities.

Minimization of the Mumford-Shah functional is difficult because of the presence of the unknown discontinuity set in the integration domain. One possible approach is Ambrosio and Tortorelli's method [15] of using an auxiliary continuous field to describe whether a pixel belongs to a smooth segment or to an edge. That method, however, does not explicitly require sharp boundaries, and may yield a segmentation field with edges that are too thick.

2.3 Vese and Chan's segmentation scheme [16]

The level set method [19] is an effective tool for computing evolving contours (an early version of this idea appeared in [20]). It accommodates topological changes and is calculated on a grid. In level set methods, regions are defined by contours that have zero width. This is the motivation for using the level-set segmentation approach in sharpening optical flow discontinuities.

In Vese and Chan’s algorithm [16], two smooth functions I^+ and I^- approximate the original image, and the segmenting contour is the zero-level set of an evolving function ϕ . Using the same fidelity and smoothness terms as in (6), the following functional is obtained:

$$\begin{aligned}
F_{VC}(I^+, I^-, \phi) = & \int (I^+ - I_0)^2 H(\phi) d\vec{x} + \int (I^- - I_0)^2 H(-\phi) d\vec{x} \\
& + \mu \int |\nabla I^+|^2 H(\phi) d\vec{x} + \mu \int |\nabla I^-|^2 H(-\phi) d\vec{x} \\
& + \nu \int |\nabla H(\phi)| d\vec{x}
\end{aligned} \tag{7}$$

$H(\phi)$ is the Heaviside (step) function of ϕ , indicating whether the valid approximation of the image at any given point is I^+ or I^- (in practice a smooth approximation of $H(\phi)$ is used). The contour length in this formulation is $|\nabla H(\phi)|$. Vese and Chan suggest minimizing this functional by iteratively solving the Euler-Lagrange equations for I^+ , I^- and ϕ .

2.4 Affine segmentation of optical flow

The final necessary background element is the affine optical flow segmentation approach [12, 21] that we shall use for the initialization of the level-set function ϕ .

Wang and Adelson [12] suggested decomposing an image sequence into layers based on the assumption that motion is affine. Their method relies on an estimate of the optical flow field. The flow field is iteratively approximated by affine motion layers of decreasing accuracy. In each iteration the optical flow field of unassigned pixels is divided into blocks, and an affine motion approximation of each block is computed. The quality of approximation of each block is tested against a selection threshold T_r (increased in each iteration). Blocks that pass this test are merged using an adaptive k-means clustering algorithm. The affine field pertaining to the largest cluster of blocks is regarded as the result of the iteration;

pixels whose optical flow field's distance from the resultant affine flow is within a threshold T_a are assigned to this iteration's layer. The algorithm is terminated when sufficiently many pixels have been assigned. Borshukov *et al* [21] proposed a variant of this algorithm in which merging by k-means clustering is replaced by using a merge threshold T_m , and selecting the affine motion field with minimal error.

3 The Piecewise Smooth Flow (PSF) algorithm

3.1 Objective functional and Euler-Lagrange equations

The functional for discontinuity preserving optical flow computation is obtained by substituting the expressions for (local) fidelity and smoothness of the optical flow field (Eqs. 3 and 4) in the segmentation functional of Vese and Chan (Eq. 7). Defining the two optical flow fields $\vec{w}^+ = (u^+, v^+, 1)$ and $\vec{w}^- = (u^-, v^-, 1)$, the proposed objective functional is obtained:

$$\begin{aligned}
F(u^+, v^+, u^-, v^-, \phi) = & \\
& \int \Psi (|I(\vec{x} + \vec{w}^+) - I(\vec{x})|^2 + \gamma|\nabla I(\vec{x} + \vec{w}^+) - \nabla I(\vec{x})|^2) H(\kappa\phi)d\vec{x} \\
& + \int \Psi (|I(\vec{x} + \vec{w}^-) - I(\vec{x})|^2 + \gamma|\nabla I(\vec{x} + \vec{w}^-) - \nabla I(\vec{x})|^2) H(-\kappa\phi)d\vec{x} \\
& + \mu \int \Psi (|\nabla u^+|^2 + |\nabla v^+|^2) H(\phi)d\vec{x} \\
& + \mu \int \Psi (|\nabla u^-|^2 + |\nabla v^-|^2) H(-\phi)d\vec{x} \\
& + \nu \int |\nabla H(\phi)|d\vec{x} \tag{8}
\end{aligned}$$

Note the different scaling of the Heaviside function in the fidelity and smoothness terms, via the parameter κ ($\kappa < 1$). It stresses flow variations near optical flow discontinuities and

leads to piecewise smoothness. See also Eq. 14.

According to the calculus of variations, a minimizer of this functional must satisfy the Euler-Lagrange equations. Following the abbreviations of [7], we denote:

$$\begin{aligned}
I_x^\pm &\triangleq \partial_x I(\vec{x} + \vec{w}^\pm) \\
I_y^\pm &\triangleq \partial_y I(\vec{x} + \vec{w}^\pm) \\
I_z^\pm &\triangleq I(\vec{x} + \vec{w}^\pm) - I(\vec{x}) \\
I_{xx}^\pm &\triangleq \partial_{xx} I(\vec{x} + \vec{w}^\pm) \\
I_{xy}^\pm &\triangleq \partial_{xy} I(\vec{x} + \vec{w}^\pm) \\
I_{yy}^\pm &\triangleq \partial_{yy} I(\vec{x} + \vec{w}^\pm) \\
I_{xz}^\pm &\triangleq \partial_x I(\vec{x} + \vec{w}^\pm) - \partial_x I(\vec{x}) \\
I_{yz}^\pm &\triangleq \partial_y I(\vec{x} + \vec{w}^\pm) - \partial_y I(\vec{x})
\end{aligned} \tag{9}$$

The equations for \vec{w}^+ at $\phi > 0$ and for \vec{w}^- at $\phi < 0$ are

$$\begin{aligned}
H(\pm\kappa\phi)\Psi'(I_z^{\pm 2} + \gamma(I_{xz}^{\pm 2} + I_{yz}^{\pm 2}))(I_x^\pm I_z^\pm + \gamma(I_{xx}^\pm I_{xz}^\pm + I_{xy}^\pm I_{yz}^\pm)) - \\
- \mu \operatorname{div}(H(\pm\phi)\Psi'(|\nabla u^\pm|^2 + |\nabla v^\pm|^2)\nabla u^\pm) = 0
\end{aligned} \tag{10}$$

$$\begin{aligned}
H(\pm\kappa\phi)\Psi'(I_z^{\pm 2} + \gamma(I_{xz}^{\pm 2} + I_{yz}^{\pm 2}))(I_y^\pm I_z^\pm + \gamma(I_{yy}^\pm I_{yz}^\pm + I_{xy}^\pm I_{xz}^\pm)) - \\
- \mu \operatorname{div}(H(\pm\phi)\Psi'(|\nabla u^\pm|^2 + |\nabla v^\pm|^2)\nabla v^\pm) = 0
\end{aligned} \tag{11}$$

The equations for \vec{w}^+ at $\phi < 0$ and for \vec{w}^- at $\phi > 0$ are the smoothness constraints:

$$\operatorname{div}(\Psi'(|\nabla u^\pm|^2 + |\nabla v^\pm|^2)\nabla u^\pm) = 0 \quad (12)$$

$$\operatorname{div}(\Psi'(|\nabla u^\pm|^2 + |\nabla v^\pm|^2)\nabla v^\pm) = 0 \quad (13)$$

Eqs. 10-13 are linearized using fixed point iterations (see Appendix, A.2). The resulting system of linear equations is sparse, and can be efficiently solved.

The gradient-descent equation for ϕ is

$$\begin{aligned} \frac{\partial \phi}{\partial t} &= \nu \delta(\phi) \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \\ &- \mu \delta(\phi) [s_{of}(u^+, v^+) - s_{of}(u^-, v^-)] \\ &- \kappa \delta(\kappa \phi) [f_{of}(u^+, v^+) - f_{of}(u^-, v^-)] \end{aligned} \quad (14)$$

where $\delta(\phi) = H'(\phi)$. We solve this equation using standard discretization of the derivatives (see Appendix, A.3).

The equations for \vec{w}^+ , \vec{w}^- and ϕ are iteratively solved in alternation. Once the solutions are determined, the final estimated optical flow field is:

$$\vec{w} = \begin{cases} \vec{w}^+ & \text{if } \phi > 0 \\ \vec{w}^- & \text{otherwise} \end{cases} \quad (15)$$

3.2 Initialization

The proposed functional is non-convex, and therefore may be prone to local minima. The initialization scheme, based on affine flow segmentation, facilitates the convergence to the correct minimum. The initial values of the fields \vec{w}^+ , \vec{w}^- and ϕ are obtained by a single

application of the original algorithm of [7], followed by detecting the dominant motion using the algorithm of [21]. After these two steps, \vec{w}^- is initialized (throughout the image domain) to the first layer of the affine field, that corresponds to the dominant motion. \vec{w}^+ is initialized (throughout the image domain) to the optical flow computed in the first step (the algorithm of [7]), and ϕ is initialized to

$$\phi = \begin{cases} 1 & \text{pixel belonging to the dominant motion} \\ 2 & \text{otherwise} \end{cases} \quad (16)$$

The rationale for these values is the following: pixels classified to the dominant motion, i.e., to the first affine layer, belong to \vec{w}^- and their initial ϕ value is low. The ϕ value for pixels classified to other motion layers is high. Initially there is a bias towards the dominant motion, because it is smoother than the combination of other motions. To compensate for this, all initial ϕ values are positive.

3.3 Piecewise-Smooth Flow (PSF): algorithm summary

1. Calculate optical flow using [7].
2. Segment the optical flow field using [21].
3. Initialize \vec{w}^+, \vec{w}^- and ϕ .
4. Iteratively solve Eqs. 10-14.
5. Assign the optical flow according to Eq. 15.

4 Experimental results

We demonstrate the performance of the piecewise smooth flow (PSF) algorithm using synthetic and real image sequences. The following parameters, taken from [7, 16], were used in all the experiments: $\mu = 80$, $\gamma = 100$, $\nu = 0.02 * 255$. The numerical approximation of the Heaviside function was $H_{\Delta}(\phi) = \frac{1}{2}(1 + \frac{2}{\pi} \tan^{-1}(\frac{\phi}{\Delta}))$, with $\Delta = 1$. The parameter κ was set to $\kappa = 0.03$, and the number of iterations was 40. The images were pre-blurred using a Gaussian kernel with $\sigma = 0.8$. The parameters for the initialization segmentation [21] were: $T_r = 0.5$, $T_m = 1$, and $T_a = 0.1$ in the first iteration, and the block size was 5×5 pixels.

Figure 1 shows that the over-smoothing in [7], that forces a gradual change of the optical flow field at the sky-ground interface in the *Yosemite* sequence, is replaced in our PSF method by segmentation of the flow field that is crisp and close (≤ 2 pixels) to the ground truth boundaries. Table 1 compares our PSF algorithm with other dense optical flow algorithms on the *Yosemite* sequence (the angular error was computed as in [2]). As can be seen, our method is superior to other published methods, even though it utilizes only two frames. Primarily, our method is an optical flow estimation method, however it also generates a segmentation result as presented in Figure 3. It can be seen that the segmentation provided by our method is much closer to the horizon than that generated by the initialization step.

Similar performance improvement was achieved on the *Street with car* sequence [24]. The sequence was converted from color to grey-levels. Due to the noisy nature of this sequence, pre-blurring leads to improved results. In this case, the best performance of both our algorithm and our implementation of [7] was obtained using a Gaussian kernel with $\sigma = 1.0$. The results for the *Street with car* sequence are presented in Figure 2 and in Table 2. The angular error is again significantly lower than that of other published methods,

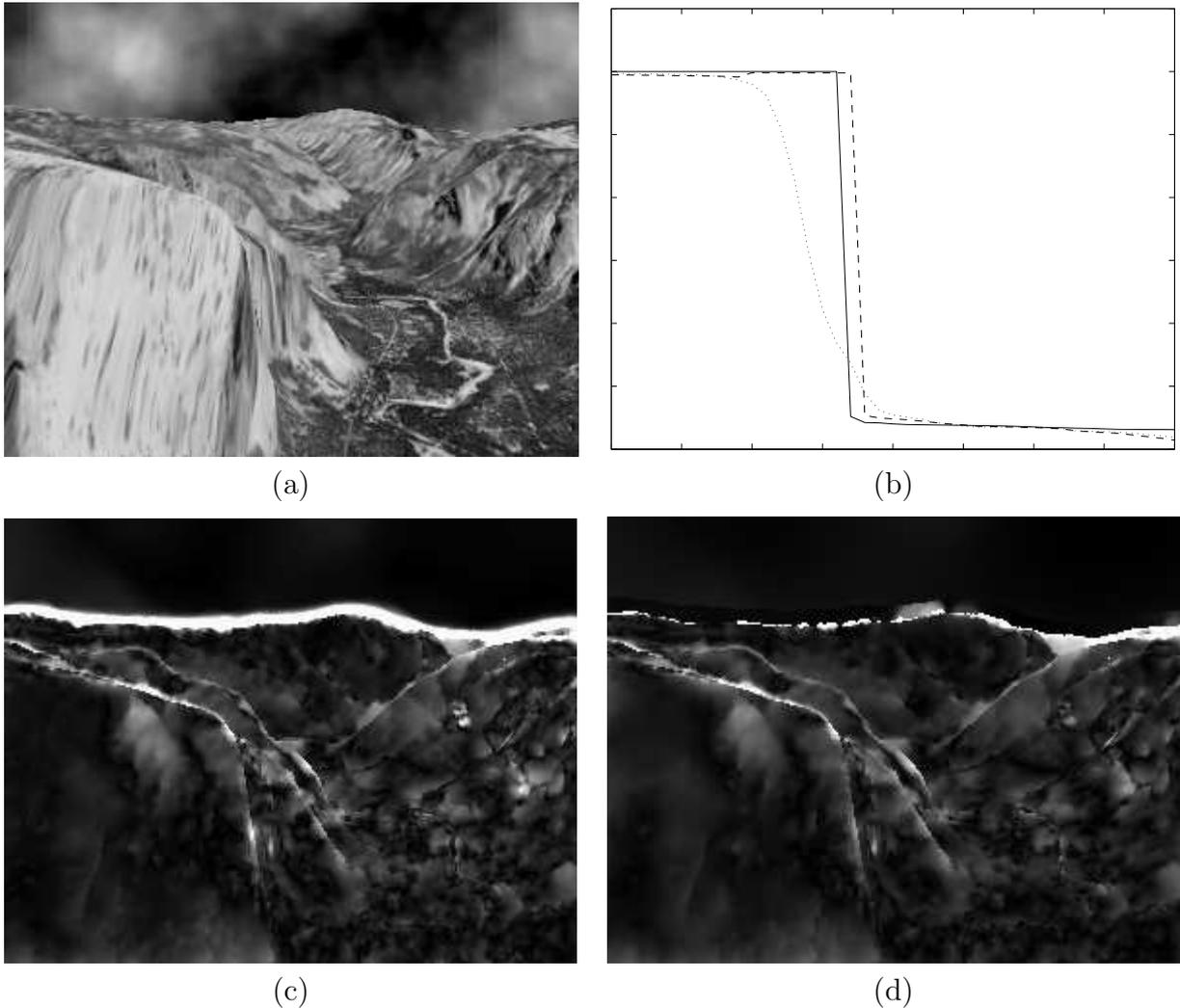


Figure 1: (a) Frame 8 of the *Yosemite* sequence. (b) The horizontal flow as a function of vertical position, at the sky-ground boundary. Solid line: ground truth. Dotted line: Brox *et al*'s 2D result. Dashed line: our piecewise-smooth flow result. (c) Angular error in Brox *et al*'s 2D results. Dark means no error. (d) Angular error using our piecewise-smooth flow algorithm.

Technique	AAE	STD
Horn-Schunck, modified [2]	9.78°	16.19°
Mémin-Pérez [10]	4.69°	6.89°
Brox <i>et al</i> (2D) [7]	2.46°	7.31°
Brox <i>et al</i> (3D) [7]	1.94°	6.02°
Our PSF algorithm	1.64°	5.82°

Table 1: Comparison between several dense optical flow algorithms on the *Yosemite* sequence. (AAE: Average angular error. STD: Standard deviation of the angular error.)

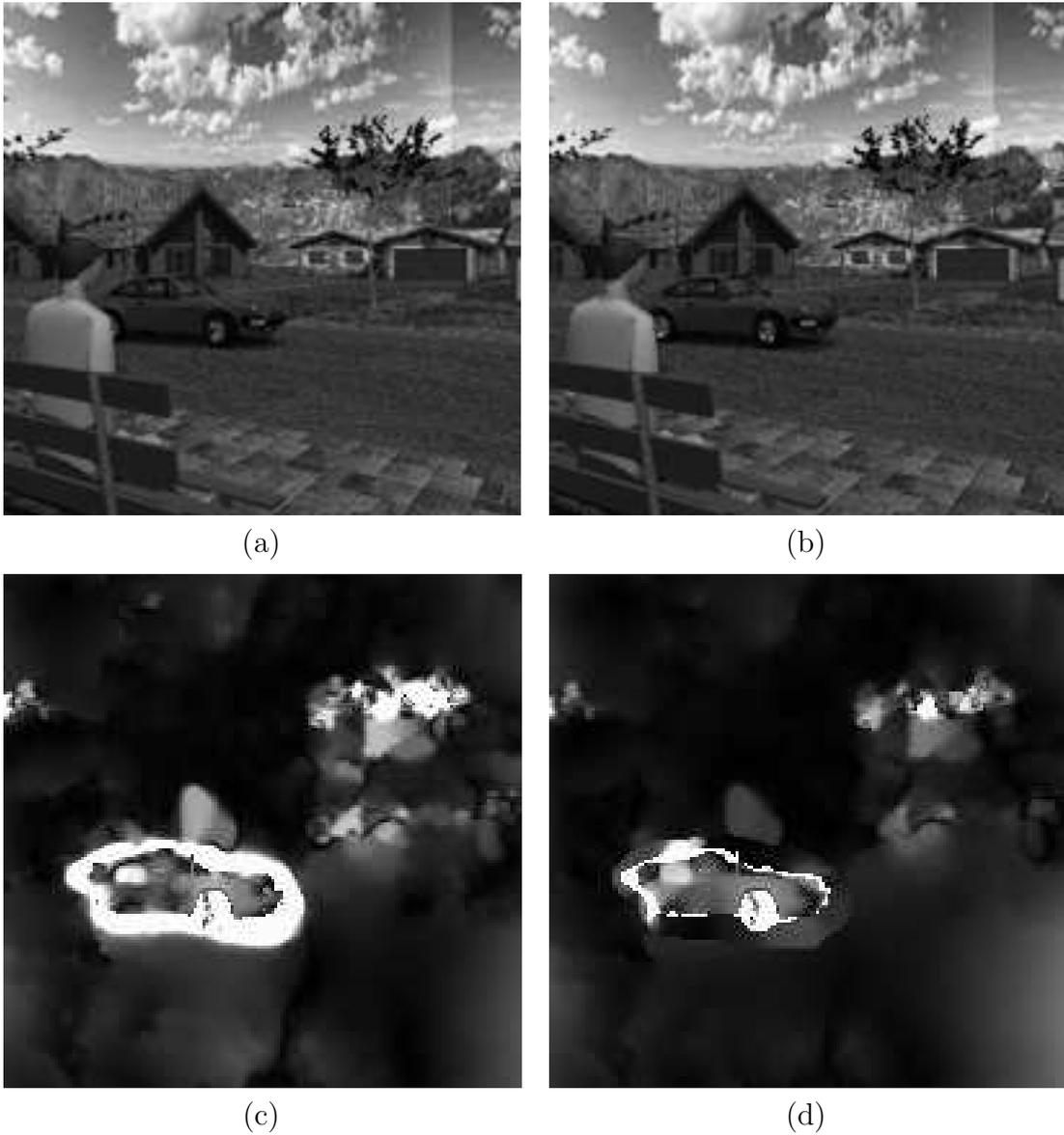


Figure 2: (a) Frame 10 of the *Street with car* sequence. (b) Frame 11. (c) Angular error with Brox *et al*'s 2D method (black: no error). (d) Angular error using our piecewise-smooth flow algorithm.

Technique	AAE	STD
Proesmans <i>et al</i> [24]	7.41°	
Weickert-Schnörr (3D) [25]	4.85°	
Brox <i>et al</i> (2D)	3.03°	10.18°
Our PSF algorithm	2.01°	8.84°

Table 2: Comparison between several dense optical flow algorithms on the *Street with car* sequence.

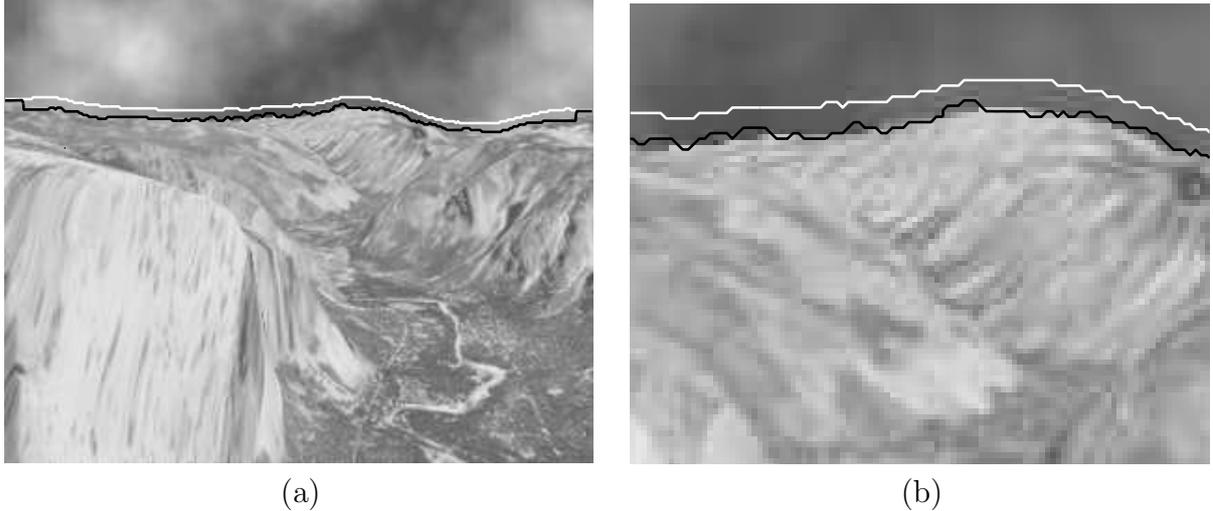


Figure 3: Boundary after initialization and at the end state of our algorithm. (a) Sky segmentation after initialization in white, black is the segmentation at the end of our algorithm. (b) Detail of (a).

κ	Yosemite		Street with Car	
	AAE	STD	AAE	STD
0.03	1.64°	5.82°	2.01°	8.84°
0.06	1.79°	6.49°	2.03°	8.66°
0.015	1.62°	6.02°	2.14°	9.72°
0.12	1.95°	7.38°	2.22°	9.43°
0.0075	1.71°	6.60°	2.43°	11.12°

Table 3: Robustness of the algorithm to variation in the parameter κ when tested on the *Yosemite* and *Street with Car* sequences.

and the error around the motion boundary is smaller.

Parameter robustness for [7] was demonstrated in their work. We introduced in this work an additional parameter: κ . We tested its sensitivity by varying it by a factor of up to 4 from its optimum setting on both the *Yosemite* and *Street with Car* sequences. Table 3 demonstrates that the algorithm shows high robustness with respect to this parameter on both sequences.

To test the PSF algorithm on real data, we used the *NASA* sequence [2]. Optical flow discontinuities were amplified by considering the flow between frames 10 and 15 of the sequence.

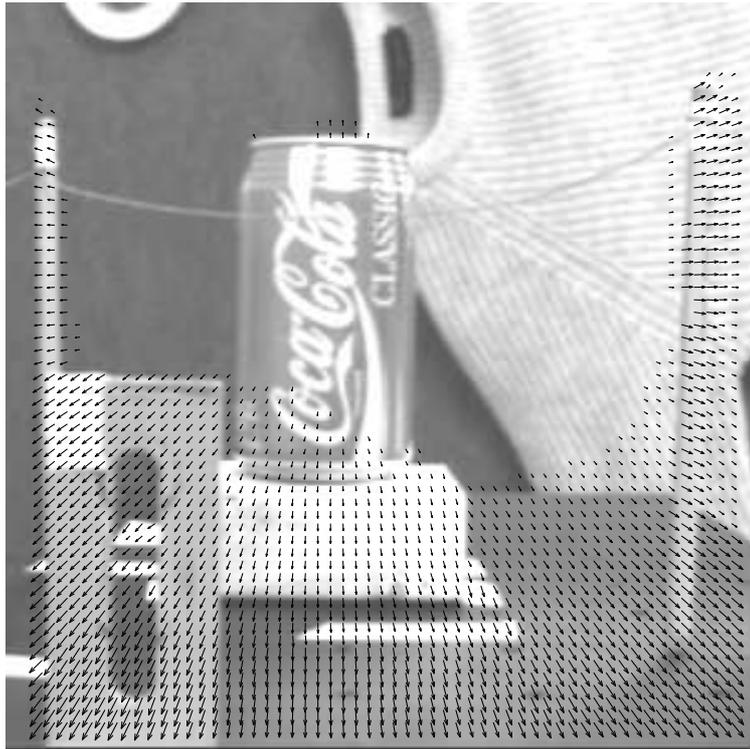


Figure 4: The optical flow between frames 10 and 15 of the NASA sequence, computed using the suggested Piecewise-Smooth Flow (PSF) algorithm. The computed flow is dense, but for clarity, vectors that represent negligible motion are not shown.

In this sequence the camera is moving toward the objects, and major flow discontinuities arise near the two vertical pencils. Figure 4 shows that the flow edges computed using our PSF algorithm are sharp. Comparison with the performance of previous techniques on this sequence [2] reveals the superior performance of the PSF algorithm.

5 Special case: static camera

Homogeneous regions in an image sequence provide no cues for optical flow computation. Optical flow methods that are not committed to dense output, e.g., Lucas-Kanade [4], simply do not provide optical flow estimation in uniform regions. Dense optical flow algorithms, that emphasize the smoothness of the computed flow, extrapolate flow from informative segments

to adjacent homogeneous regions. This leads to large errors in static homogeneous areas. The optical flow computed in uniform regions can be forced to be static [1]. Obviously, errors will arise wherever the underlying assumption (that homogeneous regions are static) does not hold. The inherent ambiguity about the motion of uniform regions can be resolved by exploiting additional information. In this section, the case of a static camera is considered, and the additional constraint is used to reduce the optical flow estimation error.

5.1 The Static Camera Flow (SCF) algorithm

When the camera is known to be static, an image of the static background can improve the separation between the moving elements and the static parts of the frames. When the background image is not directly available, it can be recovered using temporal filtering [22]. A simple background estimation method is temporal weighted median filtering [23], with weights inversely proportional to the magnitude of movement detected by the algorithm of [7]. The weighted median filter converges to the background image much faster than a non-weighted median filter, because moving elements are mostly ignored.

In the piecewise-smooth flow (PSF) algorithm described in section 3, the estimated optical flow in each pixel is taken either from the field \vec{w}^- (dominant motion) or from \vec{w}^+ (other motions), see Eq. 8. Here, given the static camera assumption, the optical flow assigned to each pixel is either zero (static region) or \vec{w} (moving). The likelihood that a given point \vec{x} in an image is moving is quantified by the dissimilarity of the image I and the background image I_{bg} at that point:

$$f_{BG} = \Psi (|I_{bg}(\vec{x}) - I(\vec{x})|^2 + \gamma|\nabla I_{bg}(\vec{x}) - \nabla I(\vec{x})|^2)$$

In the static camera flow (SCF) method, the objective functional embeds the selection between the estimated optical flow in moving areas and zero flow in static regions within the Vese-Chan segmentation framework [16]:

$$\begin{aligned}
F(u, v, \phi) = & \\
& \int \Psi (|I(\vec{x} + \vec{w}) - I(\vec{x})|^2 + \gamma|\nabla I(\vec{x} + \vec{w}) - \nabla I(\vec{x})|^2) H(\kappa\phi)d\vec{x} \\
& + \lambda \int \Psi (|I_{bg}(\vec{x}) - I(\vec{x})|^2 + \gamma|\nabla I_{bg}(\vec{x}) - \nabla I(\vec{x})|^2) H(-\kappa\phi)d\vec{x} \\
& + \mu \int \Psi (|\nabla u|^2 + |\nabla v|^2)H(\phi)d\vec{x} \\
& + \nu \int |\nabla H(\phi)|d\vec{x}
\end{aligned} \tag{17}$$

Compare this functional to that of the PSF method (Eq. 8). The term in the latter that represents the optical flow fidelity of the dominant motion is replaced by the image-background dissimilarity term. The dominant motion smoothness term that appears in Eq. 8, but is zero in the static case, is discarded.

From this functional the Euler-Lagrange equations for \vec{w} and ϕ are derived. The equations for \vec{w} are (10-13), and the equation for ϕ is

$$\frac{\partial \phi}{\partial t} = \nu \delta(\phi) \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) - \mu \delta(\phi) s_{of}(u, v) - \kappa \delta(\kappa \phi) [f_{of}(u, v) - \lambda f_{BG}] \tag{18}$$

In practice, to account for noise, slight Gaussian blurring is applied to the images as a preprocessing step.

The initialization of \vec{w} is to the optical flow calculated using [7]. The level set function ϕ is initialized to -1 throughout the image. Once the solutions \vec{w} and ϕ are determined, the

optical flow field is estimated as

$$\vec{w} = \begin{cases} \vec{w} & \text{if } \phi > 0 \\ 0 & \text{otherwise} \end{cases} \quad (19)$$

5.2 Experiments with the Static Camera Flow (SCF) algorithm

To test the static camera flow algorithm with a known background image we used a sequence from the *TAU-Dance* database [27]. Optical flow estimation in this sequence is difficult because of the homogeneous regions in the frames. Figure 5 demonstrates that the static camera flow algorithm classifies the background as static and the flow within the dancer contour is smooth, while the algorithm of [7] and the piecewise smooth flow (PSF) algorithm classify much of the static background as moving.

The *Ettlinger Tor* sequence [26] shows traffic in an intersection, viewed by a static camera, see Fig. 6a. The background image was extracted from the fifty frames available, and is shown in Figure 6b. Due to the limited number of frames and the motion of the large bus, the estimated background image is not ideal, note the artifacts near the light post. The optical flow computed using the algorithm of Brox *et al* is shown in Figure 6c-d. It is generally adequate, but in certain areas, e.g., between the bus and the preceding car, oversmoothing leads to false flow. The output of the static camera flow algorithm is presented in Figure 6e-f. It provides reliable estimation of the optical flow near optical flow discontinuities, while maintaining the good performance of [7] in other parts of the flow field.

Note that identical parameters were used for both the *TAU-Dance* and *Ettlinger Tor* sequences: $\gamma = 100$, $\mu = 80$, $\nu = 0.04 * 255$, $\lambda = 1.5$, $\kappa = 0.4$, and the number of iterations was 50. The parameters for the initialization segmentation [21] were the same as in Section 4.

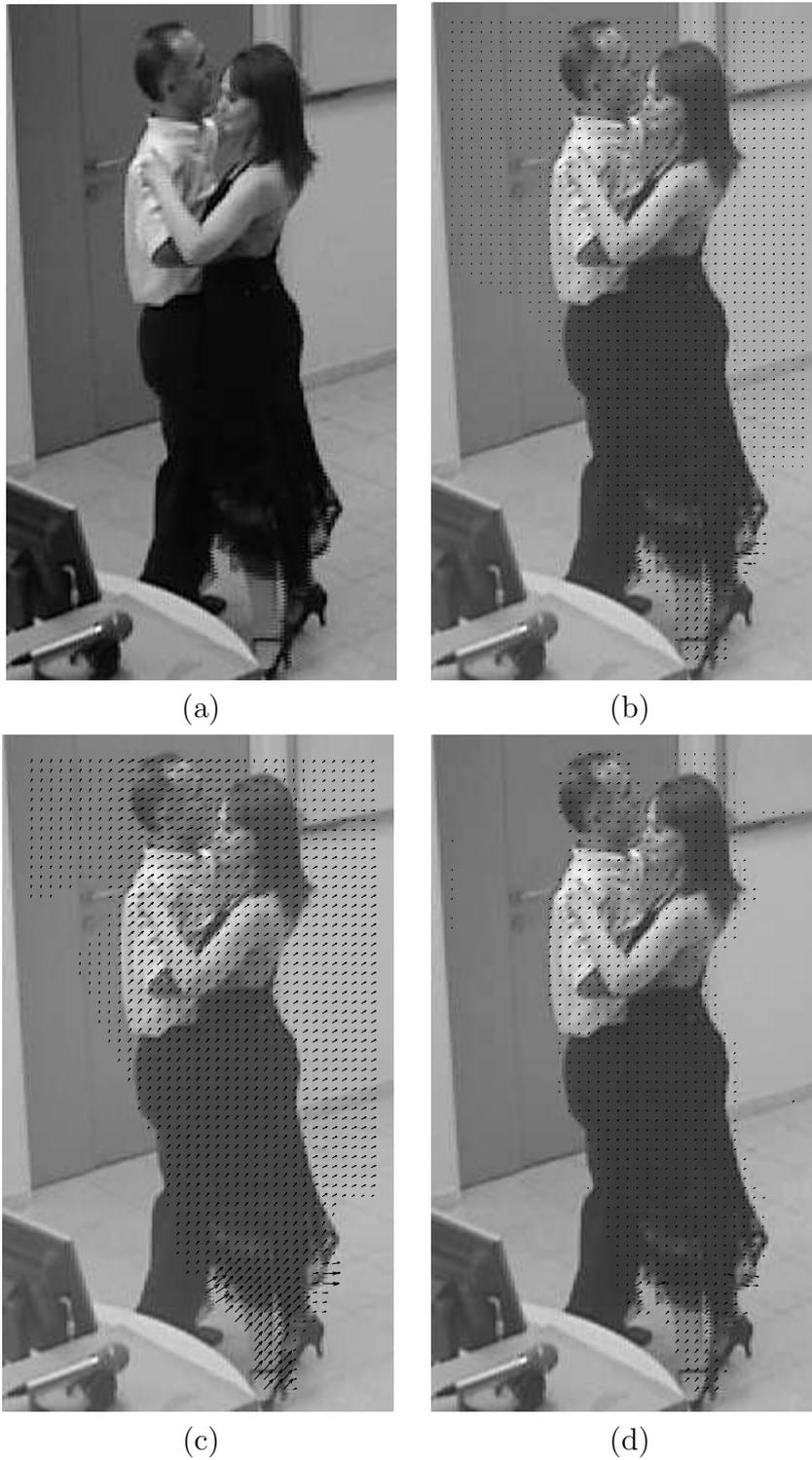


Figure 5: (a) An image from the *TAU Dance* database. (b) Optical flow calculated using the algorithm of [7]. (c) Optical flow calculated using the piecewise smooth flow (PSF) algorithm. (d) Optical flow calculated using the static camera flow (SCF) algorithm. In (b)-(d), the computed flow is dense, but for clarity, vectors that represent negligible motion are not shown.

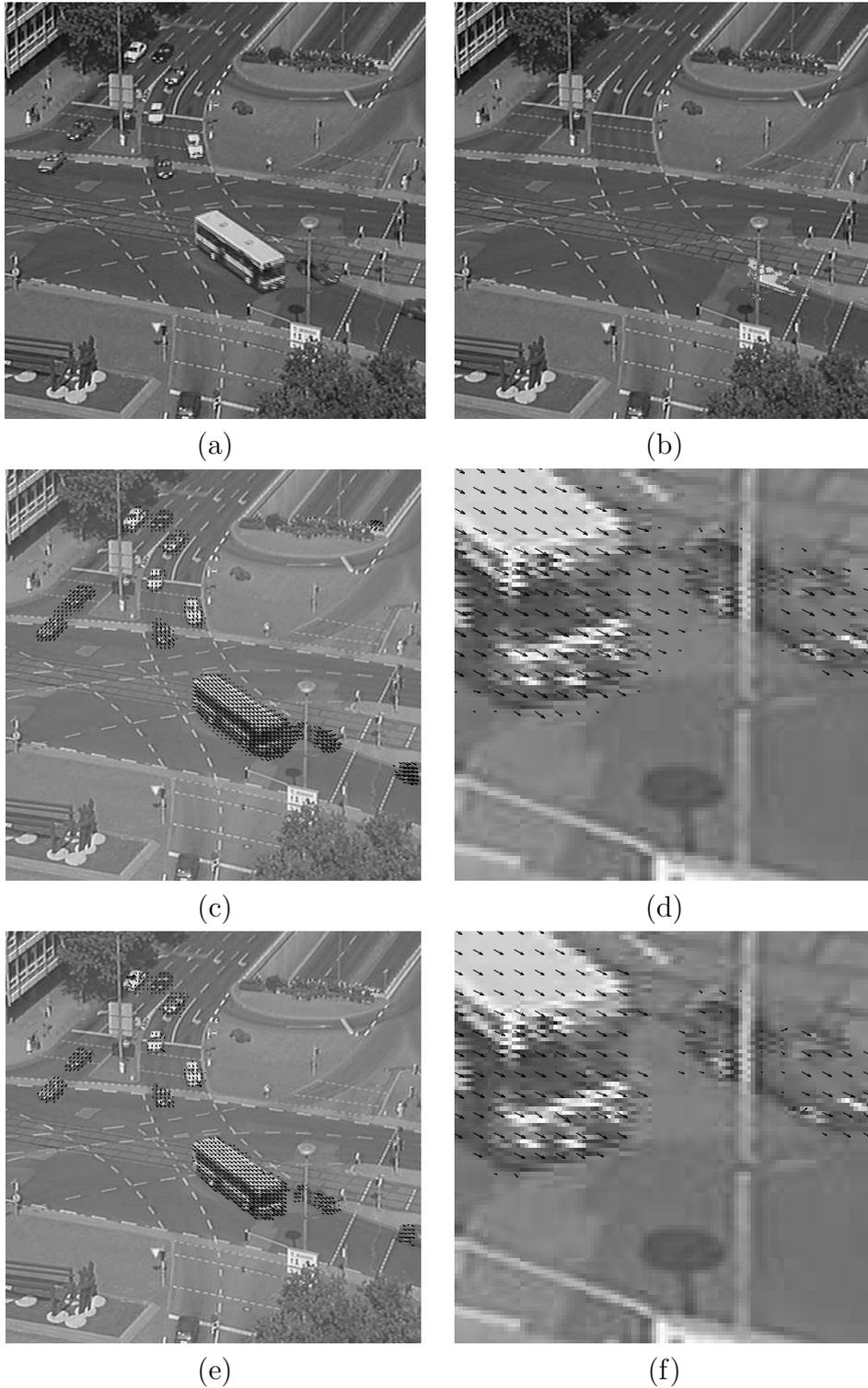


Figure 6: (a) Frame 1 of the *Ettlinger Tor* sequence. (b) Background image derived from fifty images. (c) Optical flow calculated using Brox *et al*'s algorithm. (d) Detail of (c): the bus and preceding car. (e) Optical flow calculated using our static camera flow (SCF) algorithm. (f) Detail of (e): the bus and preceding car. In (c)-(f), the computed flow is dense, but for clarity, vectors that represent negligible motion are not shown.

6 Discussion

The primary contribution of this work is the novel piecewise smooth flow algorithm, that embeds Brox *et al*'s optical flow functional within Vese and Chan's segmentation scheme. The sharp discontinuities in the estimated optical flow field provide significant performance gains. Excellent results were shown for image sequences with two dominant motions. We developed an initialization scheme based on affine flow segmentation to facilitate convergence to the correct minimum of the proposed non-convex functional.

Further research should be conducted regarding sequences with more than two dominant motions. One possible approach is to use Vese and Chan's *four-phase* formulation [16], which theoretically is equivalent to the Mumford-Shah formulation. However, while their two-phase model has a piecewise smooth extension [16], their four-phase model is piecewise constant, which is not directly applicable to our needs. An initialization scheme for a four-phase piecewise smooth Vese-Chan model is a complex problem, and may warrant future research by itself. A multi-phase segmentation model of motion is supported by Cremers and Soatto [13]. They achieve this by restricting the motion to an affine space, and thus in each phase the motion has a constant parameter vector. This model supports complex motion topologies, but is not effective as an optical flow estimation method, as it does not support non-parametric motion.

Variational methods are often perceived as slow. However, it was recently demonstrated that variational optical flow calculation can be accelerated to real-time speeds using multigrid methods [29]. The computational cost of our method is related to that of Brox *et al* [7], because it applies [7] iteratively, in conjunction with the level-set evolution.

Optical flow calculation has traditionally focused on grayscale sequences, with some re-

search on color sequences [28]. The extension of the piecewise smooth flow functional to support color is straightforward, and the robust kernel in the fidelity term should protect it from outliers due to edge-related color-channel mismatch.

The extension of optical flow to range sequences, also known as *range flow* [30] is a greater challenge, because the brightness constancy assumption does not hold in that context. To accommodate the brightness changes caused by objects moving perpendicular to the image plane, the flow calculated must be a 3D vector, the additional component representing movement towards or away from the camera.

An additional contribution of this work is employing an estimate of the background image when the sequence is known to be taken with a static camera. We used the level set method to separate the static background from the moving elements in the frames. The static camera flow algorithm performed better than state of the art optical flow methods on challenging sequences that include large homogeneous regions.

Appendix A: Numerical scheme

A.1 Basics

This appendix provides details on the numerical approximation and discretization methods used in the implementation of our algorithm. We denote

$$h \triangleq \Delta x = \Delta y .$$

Discrete spatial indices are denoted by subscripts, and the temporal index (iteration count) by a superscript, as in

$$\phi_{i,j}^n \triangleq \phi(n\Delta t, i\Delta x, j\Delta y) .$$

The numerical approximations of the Heaviside function and its derivative are

$$\begin{aligned} H_{\Delta}(x) &= \frac{1}{2} \left[1 + \frac{2}{\pi} \arctan \left(\frac{x}{\Delta} \right) \right] \\ \delta_{\Delta}(x) &= H'_{\Delta}(x) = \frac{1}{\pi} \frac{\Delta}{\Delta^2 + x^2} \end{aligned}$$

The L^1 norm and its derivative are approximated by

$$\begin{aligned} \Psi(x) &= \sqrt{x + \epsilon^2} \\ \Psi'(x) &= \frac{1}{2\sqrt{x + \epsilon^2}} \end{aligned}$$

Given the warping of an image by an optical flow field, the image values on the resampling grid are obtained by bilinear interpolation.

A.2 Optical flow

Consider the linearization and discretization of Eq. 10 for u^+ ($\phi > 0$). The treatment of Eq. 11 at $\phi > 0$ for v^+ , and of both equations at $\phi < 0$ (for u^- and v^-) are similar.

$$\begin{aligned} H(\kappa\phi)\Psi'(I_z^{+2} + \gamma(I_{xz}^{+2} + I_{yz}^{+2}))(I_x^+ I_z^+ + \gamma(I_{xx}^+ I_{xz}^+ + I_{xy}^+ I_{yz}^+)) - \\ - \mu \operatorname{div}(H(\phi)\Psi'(|\nabla u^+|^2 + |\nabla v^+|^2)\nabla u^+) = 0 \end{aligned}$$

This equation is highly non-linear. Using fixed point iterations of (u^k, v^k) , and denoting by $I_*^{k,+}$ the adaptation of the abbreviations (9) to the functions in the k -th iteration, we obtain

$$H_\Delta(\kappa\phi)\Psi' [(I_z^{k+1,+})^2 + \gamma ((I_{xz}^{k+1,+})^2 + (I_{yz}^{k+1,+})^2)] \cdot (I_x^{k,+} I_z^{k+1,+} + \gamma(I_{xx}^{k,+} I_{xz}^{k+1,+} + I_{xy}^{k,+} I_{yz}^{k+1,+})) \\ - \mu \operatorname{div} [H_\Delta(\phi)\Psi' (|\nabla u^{k+1,+}|^2 + |\nabla v^{k+1,+}|^2) \nabla u^{k+1,+}] = 0$$

To remove the non-linearity that still remains in this equation, we use the first order Taylor expansion

$$I_z^{k+1,+} \approx I_z^{k,+} + I_x^{k,+} du^{k,+} + I_y^{k,+} dv^{k,+}$$

$$I_{xz}^{k+1,+} \approx I_{xz}^{k,+} + I_{xx}^{k,+} du^{k,+} + I_{xy}^{k,+} dv^{k,+}$$

$$I_{yz}^{k+1,+} \approx I_{yz}^{k,+} + I_{xy}^{k,+} du^{k,+} + I_{yy}^{k,+} dv^{k,+}$$

where

$$du^{k,+} \triangleq u^{k+1,+} - u^{k,+}$$

$$dv^{k,+} \triangleq v^{k+1,+} - v^{k,+}$$

For conciseness, we introduce the abbreviations

$$(\Psi')_{\text{Fidelity}}^{k,+} \triangleq \Psi' [(I_z^{k,+} + I_x^{k,+} du^{k,+} + I_y^{k,+} dv^{k,+})^2 \\ + \gamma ((I_{xz}^{k,+} + I_{xx}^{k,+} du^{k,+} + I_{xy}^{k,+} dv^{k,+})^2 + (I_{yz}^{k,+} + I_{xy}^{k,+} du^{k,+} + I_{yy}^{k,+} dv^{k,+})^2)]$$

$$(\Psi')_{\text{Smooth}}^{k,+} \triangleq \Psi' (|\nabla(u^{k,+} + du^{k,+})|^2 + |\nabla(v^{k,+} + dv^{k,+})|^2)$$

Substituting the Taylor expansions and applying the abbreviations, we obtain

$$\begin{aligned}
& H(\kappa\phi) (\Psi')_{\text{Fidelity}}^{k,+} \cdot [I_x^{k,+}(I_z^{k,+} + I_x^{k,+} du^{k,+} + I_y^{k,+} dv^{k,+}) \\
& \quad + \gamma I_{xx}^{k,+}(I_{xz}^{k,+} + I_{xx}^{k,+} du^{k,+} + I_{xy}^{k,+} dv^{k,+}) \\
& \quad + \gamma I_{xy}^{k,+}(I_{yz}^{k,+} + I_{xy}^{k,+} du^{k,+} + I_{yy}^{k,+} dv^{k,+})] \\
& \quad - \mu \operatorname{div} \left[H(\phi) (\Psi')_{\text{Smooth}}^{k,+} \nabla(u^{k,+} + du^{k,+}) \right] = 0
\end{aligned}$$

We finally obtain a linear equation by another, nested, fixed point iterations loop. The internal fixed point iteration count is denoted by an additional superscript (n or $n + 1$).

$$\begin{aligned}
& H(\kappa\phi^n) (\Psi')_{\text{Fidelity}}^{k,n,+} \cdot [I_x^{k,+}(I_z^{k,+} + I_x^{k,+} du^{k,n+1,+} + I_y^{k,+} dv^{k,n+1,+}) \\
& \quad + \gamma I_{xx}^{k,+}(I_{xz}^{k,+} + I_{xx}^{k,+} du^{k,n+1,+} + I_{xy}^{k,+} dv^{k,n+1,+}) \\
& \quad + \gamma I_{xy}^{k,+}(I_{yz}^{k,+} + I_{xy}^{k,+} du^{k,n+1,+} + I_{yy}^{k,+} dv^{k,n+1,+})] \\
& \quad - \mu \operatorname{div} \left[H(\phi^n) (\Psi')_{\text{Smooth}}^{k,n,+} \nabla(u^{k,+} + du^{k,n+1,+}) \right] = 0
\end{aligned}$$

A similar linearized equation is obtained for v^+ . Discretization of both equations leads to a sparse linear system of equations, that can be solved using successive over-relaxation, or more sophisticated solvers. The solution for u^- and v^- is similar.

A.3 Segmentation

We wish to numerically solve Eq. 14:

$$\begin{aligned}\frac{\partial\phi}{\partial t} &= \nu\delta(\phi)\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right) \\ &- \mu\delta(\phi)\left[s_{of}(u^+,v^+) - s_{of}(u^-,v^-)\right] \\ &- \kappa\delta(\kappa\phi)\left[f_{of}(u^+,v^+) - f_{of}(u^-,v^-)\right]\end{aligned}$$

We adopt the notations [16]

$$\begin{aligned}C_1 &= \frac{1}{\sqrt{\left(\frac{\phi_{i+1,j}^n - \phi_{i,j}^n}{h}\right)^2 + \left(\frac{\phi_{i,j+1}^n - \phi_{i,j-1}^n}{2h}\right)^2}} \\ C_2 &= \frac{1}{\sqrt{\left(\frac{\phi_{i,j}^n - \phi_{i-1,j}^n}{h}\right)^2 + \left(\frac{\phi_{i-1,j+1}^n - \phi_{i-1,j-1}^n}{2h}\right)^2}} \\ C_3 &= \frac{1}{\sqrt{\left(\frac{\phi_{i+1,j}^n - \phi_{i-1,j}^n}{2h}\right)^2 + \left(\frac{\phi_{i,j+1}^n - \phi_{i,j}^n}{h}\right)^2}} \\ C_4 &= \frac{1}{\sqrt{\left(\frac{\phi_{i+1,j-1}^n - \phi_{i-1,j-1}^n}{2h}\right)^2 + \left(\frac{\phi_{i,j}^n - \phi_{i,j-1}^n}{h}\right)^2}}\end{aligned}$$

and

$$m = \frac{\Delta t}{h^2}\delta_\Delta(\phi_{i,j})\nu$$

$$C = 1 + m(C_1 + C_2 + C_3 + C_4)$$

The discretized fidelity and smoothness terms are denoted by

$$\begin{aligned}
\text{fid}_{i,j}^{n,+} &= \Psi \left[(I_{z \ i,j}^{n,+})^2 + \gamma (I_{xz \ i,j}^{n,+})^2 + \gamma (I_{yz \ i,j}^{n,+})^2 \right] \\
\text{fid}_{i,j}^{n,-} &= \Psi \left[(I_{z \ i,j}^{n,-})^2 + \gamma (I_{xz \ i,j}^{n,-})^2 + \gamma (I_{yz \ i,j}^{n,-})^2 \right] \\
\text{grad}_{i,j}^{n,+} &= \Psi \left[\left(\frac{u_{i+1,j}^{n,+} - u_{i-1,j}^{n,+}}{2h} \right)^2 + \left(\frac{u_{i,j+1}^{n,+} - u_{i,j-1}^{n,+}}{2h} \right)^2 + \right. \\
&\quad \left. \left(\frac{v_{i+1,j}^{n,+} - v_{i-1,j}^{n,+}}{2h} \right)^2 + \left(\frac{v_{i,j+1}^{n,+} - v_{i,j-1}^{n,+}}{2h} \right)^2 \right] \\
\text{grad}_{i,j}^{n,-} &= \Psi \left[\left(\frac{u_{i+1,j}^{n,-} - u_{i-1,j}^{n,-}}{2h} \right)^2 + \left(\frac{u_{i,j+1}^{n,-} - u_{i,j-1}^{n,-}}{2h} \right)^2 + \right. \\
&\quad \left. \left(\frac{v_{i+1,j}^{n,-} - v_{i-1,j}^{n,-}}{2h} \right)^2 + \left(\frac{v_{i,j+1}^{n,-} - v_{i,j-1}^{n,-}}{2h} \right)^2 \right]
\end{aligned}$$

The discrete evolution equation for ϕ finally takes the form

$$\begin{aligned}
\phi_{i,j}^{n+1} &= \frac{1}{C} \left[\phi_{i,j}^n + m (C_1 \phi_{i+1,j}^n + C_2 \phi_{i-1,j}^n + C_3 \phi_{i,j+1}^n + C_4 \phi_{i,j-1}^n) \right. \\
&\quad \left. - \Delta t \mu \delta_{\Delta}(\phi_{i,j}^n) \cdot (\text{grad}_{i,j}^{n,+} - \text{grad}_{i,j}^{n,-}) \right. \\
&\quad \left. - \Delta t \kappa \delta_{\Delta}(\kappa \phi_{i,j}^n) \cdot (\text{fid}_{i,j}^{n,+} - \text{fid}_{i,j}^{n,-}) \right]
\end{aligned}$$

Acknowledgments

This research was supported by MUSCLE: Multimedia Understanding through Semantics, Computation and Learning, a European Network of Excellence funded by the EC 6th Framework IST Programme.

References

- [1] G. Aubert and P. Kornprobst, *Mathematical Problems in Image Processing*, Springer, 2002.
- [2] J.L. Barron, D.J. Fleet and S.S. Beauchemin, “Performance of Optical Flow Techniques,” *Int. J. Computer Vision*, Vol. 12, pp. 43-77, 1994.
- [3] B. Horn and B. Schunck, “Determining optical flow,” *Artificial Intelligence*, Vol. 17, pp. 185-203, 1981.
- [4] B. Lucas and T. Kanade, “An Iterative Image Registration Technique with Application to Stereo Vision,” *Proc. DARPA Image Understanding Workshop*, pp. 121-130, 1981.
- [5] D.J. Fleet and A.D. Jepson, “Computation of Component Image Velocity from Local Phase Information,” *Int. J. Computer Vision*, Vol. 5, pp. 77-104, 1990.
- [6] M.J. Black and D.J. Fleet, “Probabilistic Detection and Tracking of Motion Boundaries,” *Int. J. Computer Vision*, Vol. 38, pp. 231-245, 2000.
- [7] T. Brox, A. Bruhn, N. Papenberger, and J. Weickert, “High Accuracy Optical Flow Estimation Based on a Theory for Warping,” *Proc. 8th European Conf. Computer Vision*, Part IV: *Lecture Notes in Computer Science*, Vol 3024, pp. 25-36, 2004.
- [8] P. Anandan, “A Computational Framework and an Algorithm for the Measurement of Visual Motion,” *Int. J. Computer Vision*, Vol. 2, pp. 283-310, 1989.
- [9] M.J. Black and P. Anandan, “The Robust Estimation of Multiple Motions: Parametric and Piecewise Smooth Flow Fields,” *Computer Vision and Image Understanding*, Vol. 63, pp. 75-104, 1996.

- [10] E. Mémin and P. Pérez, “A Multigrid Approach for Hierarchical Motion Estimation,” *Proc. 6th Int. Conf. Computer Vision*, pp. 933-938, Bombay, India, 1998.
- [11] P. Nesi, “Variational Approach to Optical Flow Estimation Managing Discontinuities,” *Image and Vision Computing*, Vol. 11, pp. 419-439, 1993.
- [12] J.Y.A. Wang and E.H. Adelson, “Representing Moving Images Images with Layers,” *IEEE Trans. Image Processing*, Vol. 3, pp. 625-638, 1994.
- [13] D. Cremers and S. Soatto, “Motion Competition: A Variational Approach to Piecewise Parametric Motion Segmentation,” *Int. J. Computer Vision*, Vol. 62, pp. 249-265, 2005.
- [14] D. Mumford and J. Shah, “Optimal Approximation by Piecewise Smooth Functions and Associated Variational Problems,” *Comm. Pure and Applied Mathematics*, Vol. 42, pp. 577-685, 1989.
- [15] L. Ambrosio and V.M. Tortorelli, “Approximation of Functionals Depending on Jumps by Elliptic Functionals via Γ -Convergence,” *Comm. Pure and Applied Mathematics*, Vol. 43, pp. 999-1036, 1990.
- [16] L.A. Vese and T.F. Chan, “A Multiphase Level Set Framework for Image Segmentation Using the Mumford and Shah Model,” *Int. J. Computer Vision*, Vol. 50, pp. 271-293, 2002.
- [17] N. Paragios and R. Deriche, “Geodesic Active Regions and Level Set Methods for Motion Estimation and Tracking,” *Computer Vision and Image Understanding*, Vol. 97, pp. 259-282, 2005.

- [18] T. Amiaz and N. Kiryati, “Dense Discontinuous Optical Flow via Contour-based Segmentation”, *Proc. IEEE International Conference on Image Processing (ICIP’2005)*, Vol. III, pp. 1264-1267, Genova, Italy, 2005.
- [19] S. Osher and J.A. Sethian, “Fronts Propagating with Curvature Dependent Speed: Algorithms based on Hamilton-Jacobi Formulation,” *J. Computational Physics*, Vol. 79, pp. 12-49, 1988.
- [20] A. Dervieux and F. Thomasset, “A Finite Element Method for the Simulation of Rayleigh-Taylor Instability,” *Lecture Notes in Mathematics*, Vol. 771, pp. 145-158, 1979.
- [21] G.D. Borshukov, G. Bozdagi, Y. Altunbasak and A.M. Tekalp, “Motion Segmentation by Multistage Affine Classification,” *IEEE Trans. Image Processing*, Vol. 6, pp. 1591-1594, 1997.
- [22] M. Irani and S. Peleg, “Motion Analysis for Image Enhancement: Resolution, Occlusion and Transparency,” *J. Visual Comm. Image Representation*, Vol. 4, pp. 324-335, 1993.
- [23] D.R.K. Brownrigg, “The Weighted Median Filter,” *Commun. ACM*, Vol. 27, pp. 807-818, 1984.
- [24] B. Galvin, B. McCane, K. Novins, D. Mason and S. Mills, “Recovering Motion Fields: An Evaluation of Eight Optical Flow Algorithms,” *Proc. British Machine Vision Conference*, Southampton, UK, 1998.
- [25] J. Weickert and C. Schnörr, “Variational Optic Flow Computation with a Spatiotemporal Smoothness Constraint,” *J. Math. Imaging and Vision*, Vol. 14, pp. 245-255, 2001.

- [26] H. Kollnig, H.-H. Nagel and M. Otte, “Association of Motion Verbs with Vehicle Movements Extracted from Dense Optical Flow Fields,” *Lecture Notes in Computer Science*, Vol. 801, pp. 338-347, 1994.
- [27] L. Bar, S. Rochel and N. Kiryati, “TAU-DANCE: Tel-Aviv University Multiview and Omnidirectional Video Dance Library,” VIA - Vision and Image Analysis Laboratory, School of Electrical Engineering, Tel Aviv University, January 2005.
- [28] R.J. Andrews and B.C. Lovell, “Color Optical Flow,” *Proc. Workshop Digital Image Computing*, Brisbane, pp. 135-139, 2003.
- [29] A. Bruhn, J. Weickert, T. Kohlberger, and C. Schnörr, “Discontinuity-Preserving Computation of Variational Optic Flow in Real-Time,” *Proc. Scale-Space 2005: Lecture Notes in Computer Science*, Vol. 3459, pp. 279-290, 2005.
- [30] H. Spies, B. Jähne and J.L. Barron, “Range Flow Estimation,” *Computer Vision and Image Understanding*, Vol. 85, pp. 209-231, 2002.