Anita Sellent,
Martin Eisemann,
Marcus Magnor

Calculating Motion Fields from Images with Two Different Exposure Times

Technical Report / Computer Graphics Lab, TU Braunschweig ; 2008-5-6

Anita Sellent
sellent@cg.tu-bs.de
Computer Graphics Lab, TU Braunschweig

Martin Eisemann
eisemann@cg.tu-bs.de
Computer Graphics Lab, TU Braunschweig

Prof. Dr. Ing. Marcus Magnor
magnor@cg.tu-bs.de
Computer Graphics Lab, TU Braunschweig

Calculating Motion Fields from Images with Two Different Exposure Times

Technical Report 2008-5-6

May 29, 2008

Computer Graphics Lab, TU Braunschweig

1 Abstract

In this paper, we present an extension to optical flow for estimating dense 2D motion fields with occlusion information. While optical flow computation is based on two consecutive short-exposure images, we make also use of an additional intermediate long-exposure image. This motion-blurred intermediate image captures scene motion directly. We introduce an image formation model that relates the long-exposure image to both short-exposure images in terms of dense 2D motion and local occlusion/disocclusion. Based on this image formation model, we describe a practical algorithm that enables estimating the 2D motion field of occluding and occluded regions as well as determining the instant of occlusion. Results for synthetic and real scenes demonstrate the validity of the proposed approach.

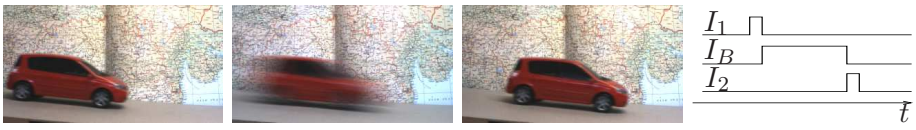
2 Introduction

The projection of 3D scene motion or camera motion onto the image plane results in dense 2D motion fields. Many computer vision applications are based on reliable motion field estimation, e.g., tracking, structure-from-motion, or image segmentation. A common approximation of the actual motion field is the optical flow [VP89]. Technically, the optical flow represents the propagation velocity of the image intensity gradient needed to account for the temporal change in the brightness pattern between two consecutive images. To numerically evaluate the local derivatives in space and time, optical flow algorithms work best with pinpoint-sharp images as input, i.e., with images depicting a moving scene at two discrete points in time. Motion-blurred images, in contrast, capture scene motion directly. Also, occlusion enters into motion-blurred image formation. Finally, sampling-theoretical considerations suggest that images depicting moving scenes should be recorded with finite exposure times to ensure adequate temporal prefiltering.

Based on these observations, we present a novel approach to estimating dense 2D motion fields, including occlusion information. In addition to two consecutive short-exposure images, our method requires an intermediate, long-exposure image, Fig. 1. We introduce an image formation model that, given the relative shutter release times, can be solved numerically for the underlying 2D motion field and occlusion information. In effect, we make use of the additional motion-blurred image to constrain the solution space of the reconstruction problem which is crucial exspecially for large motions.

3 Related work

The number of the articles published concerning the computation of the optical flow shows that this is still a very active field of research [BFB94, MB96, AN88]. Related to our work, Lim et al. [LAG05] and Brox et al. [BBPW04] propose relief to the limitation that optical flow techniques provide reliable results only if the disparity between consecutive images is not too large. Us-



I_1 I_B I_2

Figure 1: To determine the 2D motion field with accompanying occlusion information, we acquire an intermediate long-time exposure I_B in addition to two short-time exposures I_1, I_2 with the relative shutter release times shown at the right.

ing high-speed recordings or a scale-space approach, respectively, improved convergence can be obtained despite the ill-posedness of the underlying problem. Another problem in motion field computation arises from occlusion effects. Alvarez et al. [ADPS02] determine occluded regions by calculating forward and backward optical flow and checking for consistency. Areas with large discrepancies are considered occluded and are simply excluded from further computations. As a solution, Xiao et el. [XCS+06] propose interpolating motion into occluded areas from nearby regions by bilateral filtering.

There has been some previous work on calculating the optical flow from a single, long-time exposed image based on Fourier analysis [Rek96] or autocorrelation [PK03]. Though the algorithms perform well for detecting translational camera motion, the approach is unsuitable for general scene motion with several different object movements. A combination of short-time and long-time exposures has been used in the context of image restoration: Tico and Vehvilainen [TV06] model motion blur as a convolution with a spatially invariant point-spread-function which is estimated based on a statistical model of the image formation process. The hybrid camera of Ben-Esra and Nayar [BEN03] takes a long-time exposure of the scene, while a detector with a much lower spatial and higher temporal resolution takes a sequence of short-time exposures to detect the camera motion. In both cases, the blur kernel is reconstructed to de-blur (i.e., deconvolve) the long-time exposed image. For still image restauration, the assumption of a spatially invariant blur kernel is an adequate approximation. Unfortunately, this does not hold for the purpose of general motion estimation where the blur kernel may be different for every pixel. A motion de-blurring approach which takes moving objects into account is described by Raskar et al. [RAT06]. A modified shutter which opens and closes several times at irregular intervals is used to recover sharp images if the extent of 2D motion is known.

The motion-from-smear approach [CNM96] extends the underlying notion of image restoration and motion estimation. Two images with long exposure times are acquired immediately after each other. The local blur kernel, i.e., local motion is estimated by de-blurring the images to one common (unknown) reference image. Apart from being susceptible to reconstruction errors, applicability appears somewhat restricted because only motion-blurred images are used as input. This drawback also affects the method proposed by Tull and Katsaggelos [TK96] which is based on a sequence of slightly blurred images to calculate the optical flow.

The contribution of this paper is an image formation model that enables using the information provided by two short-time exposed images and an intermediate long-time exposure for calculating a dense motion field with associated occlusion timings.

4 Image formation model

In order to exploit the information provided by the long-time exposed image, we need an image formation model that relates the acquired images via 2D motion. Let $I_B : \Omega \to \mathbb{R}$ be the long-time exposed image while $I_1 : \Omega \to \mathbb{R}$ and $I_2 : \Omega \to \mathbb{R}$ are two short exposures taken at the beginning and the end of the exposure time of I_B, Fig. 1. We assume that the exposure time of I_1 and I_2 is sufficiently short so that these two images do not show any motion blur. All three images are brightness-adjusted and normalized such that in case of no motion, I_1, I_2 and I_B are identical. Note that we make here the customary assumption that scene surface appearance does not change over time. We are now going to develop our image formation model for three, increasingly general cases.

4.1 Without occlusion

For the simplest case, we consider a non-transparent, moving object in the scene which is neither occluded nor disoccluded during the entire exposure time. Take a point $\vec{x}$ in the image plane that is the projection of a scene surface point, Fig. 2a. For our discussion, we consider the fixed point $\vec{x}$ in the image plane and look at the projected surface points that pass through and do not follow the movement of a fixed pixel. At the instant of exposure of image I_1, the ray through $\vec{x}$ hits a point $y_1 \in \mathbb{R}^3$ on the scene surface. Likewise it hits a surface point y_2 at the instant of exposure of image I_2. We assume that for I_1 and I_2 the exposure time is sufficently short so that each point in the image plane corresponds to one scene point. During the time interval of the intermediate image I_B, the ray through $\vec{x}$ hits a contiguous set of scene points $Y \subset \mathbb{R}^3$. The assignment of image plane points to scene points is no longer a map between points, but each point is mapped to a subset of $\mathbb{R}^3$.

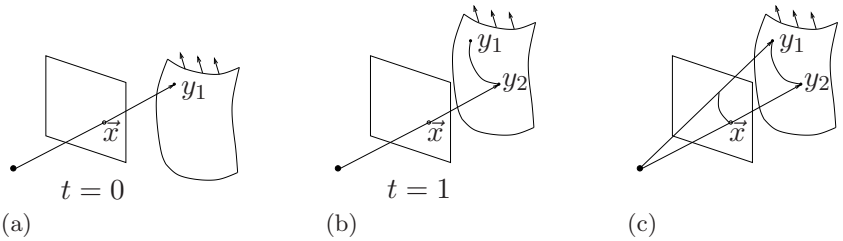


Figure 2: Image formation without occlusion: a contiguous path of scene surface points between y_1 and y_2 contributes to pixel $\vec{x}$ in the long-exposure image I_B. The projection of the path from y_1 to y_2 into the image plane forms a planar curve (c).

If the surface moves in a continuous fashion, Y is a connected 3D path which can be parametrized over time t with y_1 and y_2 as its endpoints, Fig 2b. For image I_1, the first endpoint of the path, y_1, is projected to $\vec{x}$. The rest of the path Y is projected to a curve $\vec{p_1}(\vec{x},t)$ in the image plane of I_1. For image I_2, the second endpoint of the path, y_2, is projected to $\vec{x}$. The other points of Y form a curve $\vec{p_2}(\vec{x},t)$ in the image plane of I_2, Fig. 2c. Transforming the long exposure interval to $[0,1]$ we can write

$$I_B(\vec{x}) = \int_0^1 I_1(\vec{x} + \vec{p_1}(\vec{x},t))\ dt = \int_0^1 I_2(\vec{x} + \vec{p_2}(\vec{x},t))\ dt \tag{1}$$

with $\vec{p_1}(\vec{x},0) = \vec{0}$ and $\vec{p_2}(\vec{x},0) = \vec{0}$.

Expressing (1) in terms of convolutions, $\vec{p_1}(\vec{x},t)$ and $\vec{p_2}(\vec{x},t)$ parametrize spatially variable blur kernels for I_1 and I_2, respectively, which are non-zero along a plane curve.

Since both curves are parameterized in terms of acquisition time t, we don't have to integrate one image for the entire exposure time. Instead, we are free to switch between image I_1 and I_2 at any instant $s \in [0,1]$:

$$I_B(\vec{x}) = \int_0^s I_1(\vec{x} + \vec{p_1}(\vec{x},t))\ dt + \int_0^{1-s} I_2(\vec{x} + \vec{p_2}(\vec{x},t))\ dt \tag{2}$$

with $s \in [0,1]$ arbitrary.

We will exploit this flexibility to partition the integral as needed in the following sections. Note that by differentiating (2) with respect to s and applying the fundamental theorem of calculus, we obtain the brightness constancy assumption of traditional optical flow computation. However, (2) is more general because it incorporates the additional information provided by the motion-blurred image I_B. In the following, we assume that the exposure time of I_B is still sufficiently short to justify a linear model for the curves:

$$\vec{p_1}(\vec{x},t) = -t\ \vec{w}(\vec{x}) \quad \text{and} \quad \vec{p_2}(\vec{x},t) = t\ \vec{w}(\vec{x}) \tag{3}$$

where $\vec{w} : \Omega \to \mathbb{R}^2$, $\vec{w}(\vec{x}) = (u(\vec{x}), v(\vec{x}))^T$. While there are no fundamental problems with lifting this constraint, we found the linear motion assumption to be valid for all our test data, even for our rotational motion example, Fig. 4.

4.2 With occlusion

To incorporate occlusion effects, we can partition the integral in (2) and switch between $\vec{p_1}$ and $\vec{p_2}$ at the moment of occlusion/disocclusion. Consider a point $\vec{x}$ in the image plane which becomes occluded during the long exposure time of I_B. The set Y of 3D scene points is then divided into two parts, Fig 3. While the first part lies on the background surface and contains

y_1, the second part lies on the foreground surface and contains y_2. By projecting the divided path onto the image plane, we note that the background surface part of the path is occluded in I_2, and our image formation model becomes

$$I_B(\vec{x}) = \int_0^{t_0} I_1(\vec{x} - t\ \vec{w_a}(\vec{x}))\ dt + \int_0^{1-t_0} I_2(\vec{x} + t\ \vec{w_b}(\vec{x}))\ dt \qquad (4)$$

where $\vec{w_a}$ and $\vec{w_b}$ are the linear 2D curves corresponding to scene background and foreground, respectively, and $t_0 \in [0, 1]$ is the moment at which the scene point projected to $\vec{x}$ jumps from the background surface to the foreground surface.

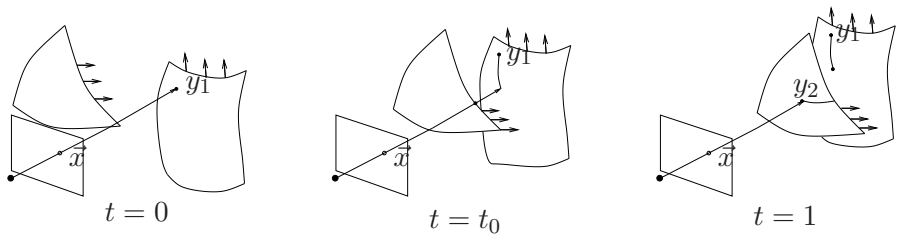


Figure 3: Image formation with occlusion: the path from y_1 to y_2 is divided into two separate parts which lie on different surfaces of the scene.

For a disoccluded scene point, the analogous consideration yields a similar equation with the roles of $\vec{w_a}$ and $\vec{w_b}$ exchanged, i.e., $\vec{w_a}$ is the linear 2D curve that correspondes to scene foreground and $\vec{w_b}$ corresponds to the scene background.

Note that we do not need to distinguish explicitly between occlusion and disocclusion but treat both cases in the same model. We assume here that a point is either occluded or disoccluded between images I_1 and I_2, but not both.

4.3 With Temporal Offset

For the most general case considered here, we want to allow for exposure gaps between the images I_1 and I_B as well as between I_B and I_2. Gaps between image exposures can occur, e.g., because of camera hardware constraints. Scene motion, of course, continues during such exposure gaps. To account for gaps, we include a temporal offset in (2) and obtain

$$\begin{aligned} I_B(\vec{x}) &= \int_0^{s} I_1(\vec{x} - \sigma_1\ \vec{w} - t\ \vec{w})\ dt\ +\ \int_0^{1-s} I_2(\vec{x} + \sigma_2\ \vec{w} + t\ \vec{w})\ dt \qquad (5) \\ &= \int_{\sigma_1}^{\sigma_1+s} I_1(\vec{x} - t\ \vec{w})\ dt\ +\ \int_{\sigma_2}^{\sigma_2+1-s} I_2(\vec{x} + t\ \vec{w})\ dt, \qquad (6) \end{aligned}$$

where we have omitted the dependency on $\vec{x}$ in $\vec{w}$ for brevity. σ_1 and σ_2 are the quotient of the length of the exposure gaps and the exposure time of I_B. Eqn. (4) transforms in the same way by replacing the limits of the interval. It is important to note that, for the general case, the paths considered in Sect. 4.1 do not contain the endpoints y_1 and y_2 anymore and occluding time detection is limited to the time recorded by I_B excluding the gap.

5 Numerical Solution

In order to determine the vertical and horizontal motion components at each point $\vec{x}$ in the image domain, we solve the time step equation (2). We first define a sequence $0 \leq s_1 < \ldots < s_N \leq 1$. Each of these s_i determines the part of the curve which is taken from I_1 while the rest of the curve is taken from I_2. By applying (2) for multiple instants, increased robustness to image noise and to deviations from the linear motion model is achieved. In this way we obtain a redundant system of N equations

$$min_{w \in \mathbb{R}^2} \sum_{i=1}^{N} \|F_i(w)\|_2 \tag{7}$$

where

$$F_i(w) = I_B - \int_0^{s_i} I_1(\vec{x} - t\ \vec{w})\ dt - \int_0^{1-s_i} I_2(\vec{x} + t\ \vec{w})\ dt \quad \text{for } i \in \{1, \ldots, N\}\ . \tag{8}$$

In order to increase robustness in the presence of intensity differences between the differently exposed images, we additionally include a differentiated version of the equations, enforcing the brightness constancy assumption:

$$F_{N+1}(w) = I_1(\vec{x} - t\ \vec{w}) - I_2(\vec{x} + t\ \vec{w}) \tag{9}$$

Taking into account the linear motion model, we evaluate equation (9) for $t = 0.5$ to obtain one more equation. This non-linear least squares problem can be solved numerically. We use a model-trust region implementation of the well-known Levenberg-Marquardt algorithm [DS83]. The path integral over the images is calculated using linear interpolation for the image functions I_1 and I_2. The derivatives of the function $F = (F_1, \ldots, F_{N+1})^T$ are determined numerically.

Unfortunately, the best studied algorithms for solving non-linear least squares optimization provide only local convergence. But it is the global optimum that we are interested in. We therefore exploit the fact that scaling the image and blurring it with the motion blur kernel are commutative, as can be shown easily. This means that the time step equation (2) remains valid on each scale of a multiscale approach. We make use of this useful

characteristic in the first step of our algorithm. In order to attenuate the impact of local noise, we iterate the least-squares solution on each scale and smooth the preliminary velocities after each iteration.

6 Algorithm

The algorithm is designed in three steps general motion detection, conservative occlusion detection, and occlusion time estimation. In the first step of the algorithm we assume that there isn't any occlusion in the image. We calculate the least-squares solution for every pixel as described in Sect. 5. For occlusion-free regions, this first step already yields the motion field. For occluded pixels, however, this intial result must be refined.

In a second step we consider two error measures to conservatively detect possibly occluded regions of the images. The first measure to detect occlusion is the optimization residual calculated during the optimization in the first step, $r = \sum_{i=1}^{N+1} \|F_i(w)\|_2$. The second error measure results from the difference between I_B and the motion-blurred versions of images I_1 and I_2 according to (1),

$$\Delta(\vec{x}) = |I_B - \int_0^1 I_1(\vec{x} - t\ \vec{w})\ dt| + |I_B - \int_0^1 I_2(\vec{x} + t\ \vec{w})\ dt| \qquad (10)$$

If one of these criteria exceeds a preset threshold η_r or η_Δ, the corresponding pixel and its immediate four neighbors are marked as possibly occluded. The threshold can be chosen conservatively as points that are erroneously marked occluded still have their appropriate motion assigned. Their initial motion estimate is only disregarded in estimating foreground and background motion. As the time step equation (2) doesn't hold for occluded regions, we use (4), i.e., we have one equation instead of $N + 1$. In addition to the motion vectors for foreground and background, we are now interested also in the instant of occlusion/disocclusion t_0. We circumvent the problem of approximating five variables with one equation, namely $\vec{w_1}$, $\vec{w_2}$ and t_0, by interpolating the motion in the occluded regions from neighboring non-occluded regions. We first explore the neighborhood of a point for trusted motion and build two clusters for the motion of the occluded and the occluding surfaces. We then compare the value of the pixel under scrutiny to the histogram of the pixels which provide the estimate for the motion. The motion corresponding to the histogram whose correlation with the gray-value $I_1(\vec{x})$ is higher is assigned to w_1 while the remaining motion vector is assigned to w_2.

In the third step, we conduct a line search to determine the moment of occlusion t_0 by minimizing (4) and introducing an additional smoothness constraint on $t_0(\vec{x})$

$$min_{t_0\in[0,1]} |I_B(\vec{x}) - \int_0^{t_0} I_1(\vec{x} - t\,\vec{w_1})\,dt + \int_0^{1-t_0} I_2(\vec{x} + t\,\vec{w_2})\,dt| + \alpha\|\Delta t_0\| \quad (11)$$

with $\alpha \in \mathbb{R}^+$ a constant. The forward motion field of $\vec{x}$ is calculated by averaging the parameters of all motion curves that pass $\vec{x}$.

7 Experiments

We evaluated our algorithm for a number of different cases: synthetic test scenes with different kinds of motion, standard test scenes for motion field determination, scenes with large displacements and real world recordings of short- and long-exposure images.

For all experimental results we used $N = 5$ equally spaced instances $s_1, \ldots s_5$ and 3 smoothing iterations while employing a three-level image pyramide as a tradeoff between accuracy and calculation time. The parameter for the threshhold η_r and η_Δ to discriminate possibly occluded points was typically set to the 0.95 quantile, dependent on the amount of occlusion in the scene.

For our synthetic test scenes, Fig.4, we ray-traced many images at short time intervals and averaged the images to obtain the motion-blurred result I_B. The first and the last ray-traced image represent the short-time exposures I_1, I_2. For both translational as well as rotational movement, the motion field is recovered with high degree of accuracy. Note that due to our conservative occlusion detection approach, a number of pixels are falsely labeled occluded, while virtually none of the actually occluded pixels is labeled non-occluded. Since those pixels falsely labeled occluded move with the same velocity $\vec{w_a} = \vec{w_b}$ and (4) specializes to (2), i.e., the moment of occlusion is irrelevant for motion field reconstruction, the correct motion is still assigned.

We used the standard test data of [BSL$^+$07] to compare the results of our algorithm to the results obtained by other algorithms. With the help of the ground truth motion field we created the intermediate, motion-blurred image, Fig. 5, assuming that the foreground always moves faster than the background. By comparing our motion field results to the ground-truth, we are able to calculate the mean angular error, Table 1. Results for Brox et al. [BBPW04] are included for comparison and further results obtained with other algorithms can be found in [BSL$^+$07]. Fig.5 presents the residual error in motion magnitude for our approach. Note that the ground-truth at points that become occluded is undefined, i.e. set to zero, so reliable data for the image creation is not available there.

To test our approach also on real-world scenes including displacements larger than 15 pixels, we took consecutive video frames of fast motions and

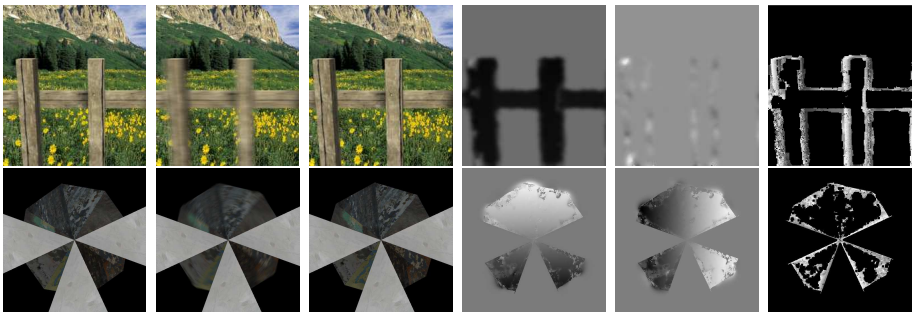


Figure 4: Results of our algorithm for raytraced images of two synthetic scenes from left to right: I_1, the motion-blurred image I_B, I_2, the horizontal motion component, the vertical motion component and grayscale coded moment of occlusion.

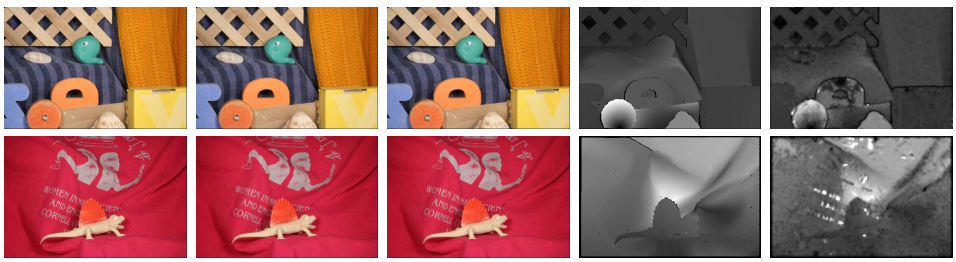


Figure 5: Results for standard test scenes *Seashell* and *Dimetrodon* of [BSL+07]; the motion blurred images I_B were created synthetically by warping. From left to right: I_1, I_B, I_2, the ground truth motion field (magnitude, grayscale coded) and the motion magnitude of our method

reconstructed the dense warp field semi-automatically. We applied the warp field to the first video frame to create the intermediate, motion-blurred image and the second short-exposure image, Fig. 6. We also computed the mean angular error for these cases, Table 1. Results for our implementations of the optical flow algorithms by Horn/Schunk [HS81] and Brox et al. [BBPW04] are included for comparison. Fig.6 presents the residual error in motion magnitude for our approach and Brox et al. [BBPW04]. The error of the optical flow computation is increased by the undetected motion in the background where the color changes only slightly.

Finally, we tested our algorithm also on completely recorded, real-world images. We externally triggered a PointGrey Flea2 camera and programmed it such that it alters exposure time and gain between successive frames. This way, we were able to acquire I_1, I_B and I_2 with a fixed time gap between the images. The recorded images and the estimated motion field are shown

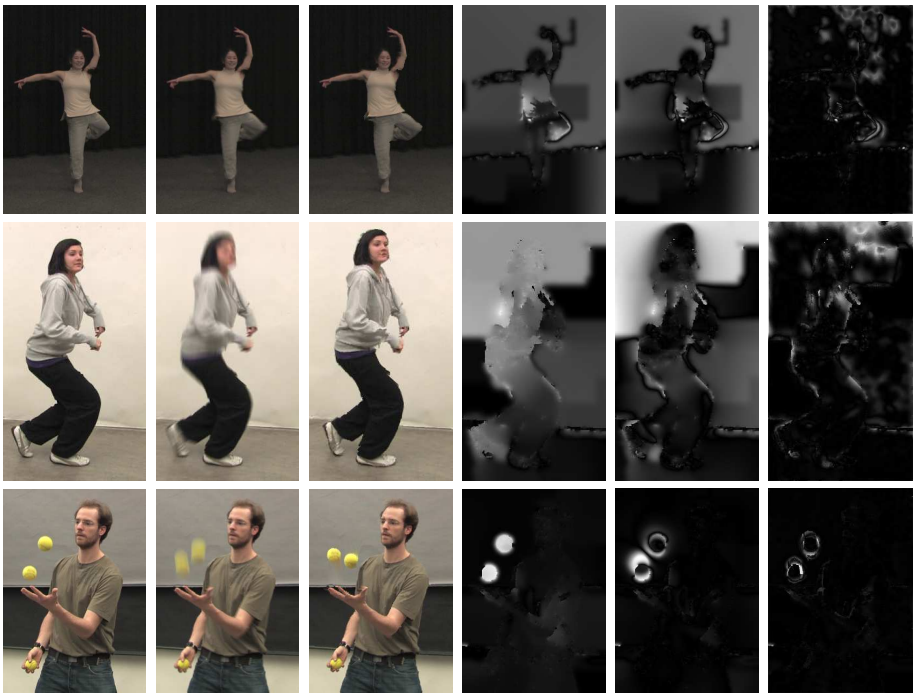


Figure 6: Results for large displacement test scenes *Ballet*, *Dancer* and *Juggler*; the motion blurred images I_B were created synthetically by warping. From left to right: I_1, I_B, I_2, the ground truth motion field (magnitude, grayscale coded), the difference in motion magnitude between ground truth and optical flow [BBPW04] and the difference in motion magnitude between ground truth and our method

in Fig. 7.

We found that the algorithm performs well also for non-linear motion, e.g., the rotation, Fig. 4. The algorithm is able to estimate large motion also in areas of only slight color change, as can be seen in the background of Fig. 6. Finally, there exists a "healing" smoothing effect for pixels erroneously labeled occluded: in Fig. 7, the saturated pixels in the overexposed highlights on the mirror balls are not in correspondence with the image formation model. Due to the high error measure, these pixels are labeled as occluded. However, the local interpolation approach for occluded pixels assigns consistent 2D motion even to such pixels.

8 Conclusions

In this paper, we extended the image formation model underlying optical flow to include also motion-blurred image information. The long exposure

Table 1: Mean angular error for the standard test scenes ond the large displacement test scenes (cf. Fig 5 and Fig 6): of our algorithm (with $N = 5$, 3 iterations per scale), an implementation of Brox et al. [BBPW04] and Horn/Schunck [HS81] (parameters optimized). Note that the error of classical optical flow for test scenes *Ballet* and *Dancer* is increased by the undetected motion of the relative uniform background.

Test Scene	our method	Brox et al.	Horn/ Schunck
Seashell	6.7249°	11.09°	-
Dimetrodon	6.2954°	10.99°	-
Ballet	11.4873°	50.3173°	51.5462°
Dancer	7.8967°	30.2181°	52.6645°
Juggler	9.5747°	16.2839°	13.9422°

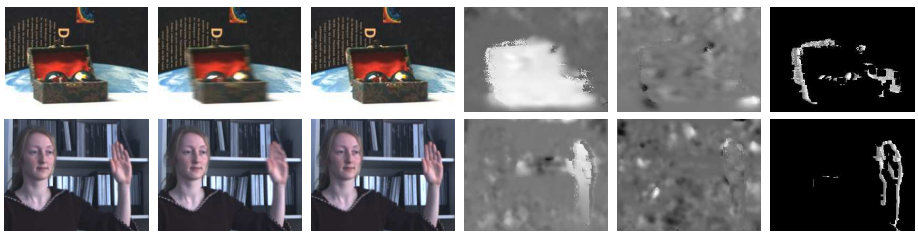


Figure 7: Real-world images acquired with a camera programmed to change gain and exposure time after each frame (from left to right): I_1,I_B,I_2, horizontal motion component, vertical motion component calculated with our algorithm and grayscale-coded occlusion time.

time image contains additional information about scene motion that is not recorded in any of the short-time exposures. The proposed algorithm provides dense motion field estimates in conjunction with occlusion information. The obtained results encourage us to exploit the redundancy of the developed system of equations further to take into account more complex motion models as well as changes in illumination.

References

[ADPS02] L. Alvarez, R. Deriche, T. Papadopoulo, and J. Sanchez. Symmetrical Dense Optical Flow Estimation with Occlusions Detection. *Proc. ECCV*, pages 721–735, 2002.

[AN88] JK Aggarwal and N. Nandhakumar. On the computation of motion from sequences of images-A review. *Proceedings of the IEEE*, 76(8):917–935, 1988.

[BBPW04] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. *Proc. ECCV*, pages 25–36, 2004.

[BEN03] M. Ben-Ezra and Shree K. Nayar. Motion deblurring using hybrid imaging. In *Proc. IEEE Computer Vision and Pattern Recognition*, volume I:657-664, June 2003.

[BFB94] JL Barron, DJ Fleet, and SS Beauchemin. Performance of optical flow techniques. *Int. J. of Computer Vision*, 12(1):43–77, 1994.

[BSL+07] S. Baker, D. Scharstein, JP Lewis, S. Roth, M.J. Black, and R. Szeliski. A Database and Evaluation Methodology for Optical Flow. *Proc. ICCV*, 2007.

[CNM96] Wei-Ge Chen, N. Nandhakumar, and Worthy N. Martin. Image motion estimation from motion smear-a new computational model. *IEEE T-PAMI*, 18(4):412–425, Apr. 1996.

[DS83] J.E. Dennis and R.B. Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. Prentice-Hall Englewood Cliffs, NJ, 1983.

[HS81] B.K.P. Horn and B.G. Schunck. Determining Optical Flow. *Artificial Intelligence*, 17(1-3):185–203, 1981.

[LAG05] S. Lim, J.G. Apostolopoulos, and A.E. Gamal. Optical flow estimation using temporally oversampled video. *IEEE Transactions on Image Processing*, 14(8):1074– 1087, Aug. 2005.

[MB96] A. Mitiche and P. Bouthemy. Computation and analysis of image motion: A synopsis of current problems and methods. *Int. J. of Computer Vision*, 19(1):29–55, 1996.

[PK03] T.L. Pao and M.D. Kuo. Estimation of the point spread function of a motion-blurred object from autocorrelation. *Proceedings of SPIE*, 2501:1226, 2003.

[RAT06] R. Raskar, A. Agrawal, and J. Tumblin. Coded exposure photography: motion deblurring using fluttered shutter. *ACM Transactions on Graphics*, 25(3):795–804, 2006.

[Rek96] Ioannis M. Rekleitis. Optical flow recognition from the power spectrum of a single blurred image. In *Int. Conf. on Image Processing*. IEEE, Sep. 1996.

[TK96] D. Tull and A. Katsaggelos. Regularized blur-assisted displacement field estimation. *Int. Conf. on Image Processing*, 1996.

[TV06] Marius Tico and Markku Vehvilainen. Bayesian estimation of motion blur point spread function from differently exposed image frames. In *Proc. of the 14th European Signal Processing Conference (EUSIPCO)*, Florence, Italy, September 2006.

[VP89] A. Verri and T. Poggio. Motion field and optical flow: qualitative properties. *IEEE T-PAMI*, 11(5):490–498, 1989.

[XCS+06] J. Xiao, H. Cheng, H. Sawhney, C. Rao, and M. Isnardi. Bilateral filtering-based optical flow estimation with occlusion detection. *Proc. ECCV*, 2006.