

# Computing local velocity estimates of image sequences by constructive interference of global Fourier components

Babette K. Dellen and Florentin Wörgötter

Manuscript received January 30, 2007

B.K. Dellen and F.Wörgötter are with the Bernstein Center for Computational Neuroscience, Göttingen, Germany.

### Abstract

We propose a novel Fourier-based technique for the estimation of optic-flow fields from image sequences. In this method, the instantaneous velocities of local image points are inferred directly from the global 3D Fourier components of the image sequence by searching for those velocities for which the superposition of the corresponding Fourier gratings leads to constructive interference at the image point. Since the localization of velocities within the image space is derived from global signals, the technique does not suffer the aperture problem and is capable of capturing motion signals that take place on different spatiotemporal scales. We test the algorithm on both synthetic and real image sequences. The performance is evaluated by using error measurements, and the results are compared to standard techniques for optic-flow computation.

### Index Terms

Optic flow, global signal processing, uncertainty principle, aperture problem, Fourier transform.

## I. INTRODUCTION

Optic flow is one of the most important attributes for analyzing visual scenes. From the optic-flow field, information about the state of the observer, the 3D layout of the visual scene, and the attributes of independently moving objects can be inferred. Dense and accurate optic-flow fields are the prerequisite for many vision applications.

The computation of velocity is intrinsically connected to the representation of images in Fourier space. All the nonzero power associated with a translating 2D-pattern lies on a plane through the origin in Fourier space. The pattern velocity determines the orientation of the plane, and can be derived from the nonzero Fourier components by finding the velocity for which the constraint lines of the Fourier components intersect. However, usually the image sequence contains more than one object. Most optic-flow techniques try to resolve this problem by restricting their computations to a small spatiotemporal region of the image sequence [1], [2]. Given that this region is chosen sufficiently small, it will, in the ideal case, contain only the object of interest, which will then have an approximately constant velocity. There is a catch though: since the measurement of frequencies is based on spatiotemporal sampling, the velocity estimates become less accurate with decreasing window size. This phenomenon is known as the uncertainty principle in signal processing [3].

Several approaches have been made to facilitate image segmentation by representing images in the global Fourier domain. Vernon developed a Fourier-based method for segmenting images which are formed by an occluding foreground and occluded background for situations in which the object velocity is constant and normal to the principal ray of the image sensor [4]. Dellen et al. demonstrated that objects which are moving in opposing directions are separable in Fourier space and reconstructed the segmented image regions in real space by applying the inverse Fourier transform [5]. Global Fourier components have also been used to reduce motion estimation errors in a hybrid approach [6]. However, a solid bridge between spatiotemporal localization and velocity measurements of image points is still missing.

To fill this gap, we propose a novel algorithm for the estimation of optic flow which derives velocity estimates from the global Fourier components and reconstructs spatiotemporal position through constructive interference of the same Fourier components. Since the local velocity estimates are computed from global frequency measurements, the algorithm does not suffer the aperture problem which is caused by local windowing.

The paper is structured as followed: In Section II, the basic idea and the algorithmic framework are explained. In Section III, error measures for the performance evaluation of the algorithm are introduced. In Section IV, the results of the algorithm for synthetic and real image sequences are shown. The performance of the algorithm is quantified by error measurements for those sequences for which the true flow field is known and compared to other optic-flow algorithms. In Section V, the results are discussed and suggestions for future applications are given.

## II. ALGORITHMIC FRAMEWORK

The visual scene observed by human eyes or a camera can be represented as a three-dimensional discrete function of intensity values  $I(\mathbf{x}, t)$ , where  $\mathbf{x} = (x, y)$  defines the spatial dimensions and  $t$  the temporal dimension. By means of Fourier decomposition, the image sequence can be described as a superposition of translating gratings, such that

$$I(\mathbf{x}, t) = \sum_{\mathbf{k}, k_t} A(\mathbf{k}, k_t) \cos(\mathbf{k} \cdot \mathbf{x} - k_t t) + B(\mathbf{k}, k_t) \sin(\mathbf{k} \cdot \mathbf{x} - k_t t) \quad , \quad (1)$$

where  $\mathbf{k} = (k_x, k_y)$  is a wave vector with spatial frequencies  $k_x$  and  $k_y$ , and  $k_t$  is a temporal frequency. The amplitudes  $A(\mathbf{k}, k_t)$  and  $B(\mathbf{k}, k_t)$  depend on the spatial and temporal frequencies

of the gratings and represent the image sequence in 3D Fourier space. Each grating moves with a velocity  $\mathbf{v} = k_t \mathbf{k} / k^2$ , where  $k = |\mathbf{k}|$  is the absolute spatial frequency of the grating. The combined movements of the gratings contributing to a particular point  $(\mathbf{x}, t)$  of the image sequence determine the local velocity  $\mathbf{V}(\mathbf{x}, t)$  at this point. A schematic of this principle is given Fig. 1. Importantly, for non-transparent conditions, only intensities belonging to a single object can be represented at a local point  $(\mathbf{x}, t)$ . The contribution of a grating to the velocity of a point  $(\mathbf{x}, t)$  of the image sequence can be quantified by assigning a weight to every grating. For example, a grating which has a negative amplitude at a point of positive intensity is contributing destructively, while a grating of positive amplitude contributes constructively. Thus, weights depend not only on the amplitude of the grating at this point, but also on the intensity of the point itself, such that

$$w(\mathbf{x}, t, \mathbf{k}, k_t) = [A(\mathbf{k}, k_t) \cos(\mathbf{k} \cdot \mathbf{x} - k_t t) + B(\mathbf{k}, k_t) \sin(\mathbf{k} \cdot \mathbf{x} - k_t t)] \text{sign}[I(\mathbf{x}, t)] \quad , \quad (2)$$

where  $\text{sign}[\pm a] = \pm$  is the sign function. Hence, for each point  $(\mathbf{x}, t)$  of the image sequence, we obtain a set of weighted velocities  $\{\mathbf{v}(\mathbf{k}, k_t); w(\mathbf{x}, t, \mathbf{k}, k_t)\}$ . The set of weighted velocities can be used to estimate the local velocity  $\mathbf{V}(\mathbf{x}, t)$  of the image point since all gratings belonging to an object moving with a velocity  $\mathbf{v}$  fulfill the motion-constraint equation  $k_t = \mathbf{v} \cdot \mathbf{k}$  [7].

We estimate the local velocity by employing the following voting scheme: Each component velocity votes with its assigned weight for all local velocities which lie along the corresponding constraint line, yielding a map of votes

$$m(\mathbf{U}, \mathbf{x}, t) = \sum_{\mathbf{k}, k_t} w(\mathbf{x}, t, \mathbf{k}, k_t, \mathbf{x}, t) \exp[-(k_t - \mathbf{U} \cdot \mathbf{k})^2 / (\xi |\mathbf{k}|)^2] \quad , \quad (3)$$

where  $\xi$  defines the width of the Gaussian. If the velocity signal is sufficiently strong, the intensity map peaks at  $\mathbf{U} = \mathbf{V}(\mathbf{x}, t)$ , where  $\mathbf{V}(\mathbf{x}, t)$  is the local velocity of the point. The estimated velocity becomes

$$\mathbf{V}_e(\mathbf{x}, t) = \arg\{\max[m(\mathbf{U})]\} \quad . \quad (4)$$

#### A. Eliminating false velocity estimates

We measure the ‘‘confidence’’  $G(\mathbf{x}, t)$  of the velocity estimated at  $\mathbf{x}$  by computing the correlation coefficient of the map of votes with a Gaussian centered at  $\mathbf{V}_e(\mathbf{x}, t)$

$$G(\mathbf{x}, t) = C \{ \exp[-(\mathbf{U} - \mathbf{V}_e(\mathbf{x}, t))^2 / \sigma^2], m(\mathbf{U}, \mathbf{x}, t) \} \quad , \quad (5)$$

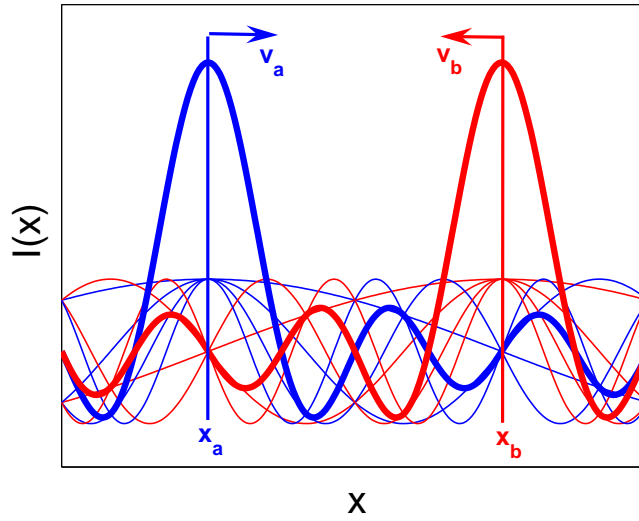


Fig. 1. Schematic: Object reconstruction through interference of global Fourier gratings. Two moving objects, depicted as red and blue arrows, are plotted together with a few of their Fourier gratings. When superposing the Fourier components of each object (thick blue and red lines), the Fourier components belonging to the blue object (thin blue lines) interfere constructively at the position of the blue object, but interfere destructively everywhere else. Since both objects are moving, all Fourier components belonging to the blue object move with a velocity  $v_a$ , while the Fourier components of the red objects move with a velocity  $v_b$ . Hence, when multiplying all Fourier components with their respective component velocity and computing their sum locally at  $x_a$ , we obtain the local velocity at  $x_a$ , namely  $V_a$ , since the contributions of the component velocities belonging to the red object sum to zero due to destructive interference. The same holds for the red object at  $x_b$ . By utilizing the principle of superposition, local information can be inferred from global signals, which, taken independently, seem to carry to information about position.

where  $\sigma$  is a parameter determining the width of the Gaussian. The function  $G$  provides a confidence measure for each estimated local velocity value. For example, if the velocity signal is too weak, no peak will emerge from the map of votes, and Eq. 4 will return a wrong velocity estimate. In this case, the confidence will be small and the velocity estimate can be discarded. We use a parameter  $\tau$  to threshold the confidence function.

However, in some cases, the intensity map exhibits a peak even if the true velocity signal of the image point is too weak to be distinguishable. Such false peaks might be caused by random interference effects of the Fourier gratings. It is therefore useful to also discard the velocity estimates belonging to image points which do not give a strong velocity signal. Velocity signals can be too weak for proper detection for two reasons: (i) the point is part of a large homogeneous

image area such that the low spatial frequencies are not reflecting the true velocity of the image point (undersampling). (ii) the intensity of the point is close or equal to the mean intensity of the image.

The first error source (i) can be essentially removed by preprocessing the image sequence with a velocity high-pass filter, cutting off all static gratings:

$$\Psi_v = \begin{cases} 1 & \text{if } |k_t/k| > \tau_v, \\ 0 & \text{else.} \end{cases}, \quad (6)$$

where  $\tau_v$  is a threshold. Usually, we choose the threshold  $\tau_v$  to be equal to the velocity resolution of the algorithm.

The second error source (ii) can be diminished by estimating only velocities for image points which are well above the mean intensity of the image, choosing

$$\Psi_I = \begin{cases} 1 & \text{if } |I(\mathbf{x}, t)| > \tau_I \max(|I(\mathbf{x}, t)|), \\ 0 & \text{else.} \end{cases}, \quad (7)$$

### III. ERROR MEASUREMENTS

When the true velocities of an image sequence is given, error measures can be computed to quantify the performance of the algorithm. In this paper, we compute two error measures, the absolute error and the angular error, which are presented together with the standard error. The absolute error is defined as the mean absolute difference of the estimated velocities  $\mathbf{V}_e$ ,

$$\bar{E}_a = (n \cdot m / \rho) \sum_{\mathbf{x}} E_a(\mathbf{x}) \theta[G(\mathbf{x}) - \tau] \cdot \Psi_v(\mathbf{x}) \cdot \Psi_I(\mathbf{x}), \quad (8)$$

with

$$E_a(\mathbf{x}) = |\mathbf{V}(\mathbf{x}) - \mathbf{V}_e(\mathbf{x})|, \quad (9)$$

and with a standard error

$$\Delta \bar{E}_a = (n \cdot m / \rho) \left[ \sum_{\mathbf{x}} (E_a(\mathbf{x}) - \bar{E}_a)^2 \theta[G(\mathbf{x}) - \tau] \cdot \Psi_v(\mathbf{x}) \cdot \Psi_I(\mathbf{x}) \right]^{-1/2}. \quad (10)$$

Here,  $\theta(a)$  is the step function, with  $\theta(a)$  equal to one if  $a > 0$  and zero otherwise. The functions  $\Psi_v$  and  $\Psi_I$  are only included if the image sequence is preprocessed with a high-pass velocity filter (see Section II.A). The parameters  $m$  and  $n$  are the sizes of the image in  $x$  and  $y$  direction, respectively.

The density  $\rho$  of the estimated optic flow field is defined as the total number of pixels for which an velocity estimate is computed divided by the total number of pixels,

$$\rho = \sum_{\mathbf{x}} \{\theta[G(\mathbf{x}) - \tau] \cdot \Psi_v(\mathbf{x}) \cdot \Psi_I(\mathbf{x})\} / (n \cdot m) \quad . \quad (11)$$

The angular error is defined as

$$\bar{E}_\phi = (n \cdot m / \rho) \sum_{\mathbf{x}} E_\phi(\mathbf{x}) \theta[G(\mathbf{x}) - \tau] \cdot \Psi_v(\mathbf{x}) \cdot \Psi_I(\mathbf{x}) \quad , \quad (12)$$

with

$$E_\phi(\mathbf{x}) = \arccos\{(|\mathbf{V}(\mathbf{x}) \cdot \mathbf{V}_e(\mathbf{x})|^2 + 1)^{-1/2} / [(|\mathbf{V}(\mathbf{x})|^2 + 1) \cdot (|\mathbf{V}_e(\mathbf{x})|^2 + 1)]^{-1/2}\} \quad , \quad (13)$$

and with a standard error

$$\Delta \bar{E}_\phi = (n \cdot m / \rho) \left[ \sum (E_\phi(\mathbf{x}) - \bar{E}_\phi)^2 \theta[G(\mathbf{x}) - \tau] \cdot \Psi_v(\mathbf{x}) \cdot \Psi_I(\mathbf{x}) \right]^{-1/2} \quad . \quad (14)$$

## IV. RESULTS

### A. Translating-square sequence

We illustrate the algorithm first on a sequence which contains a square, 10 pixels wide, which moves on a white background with a velocity of  $\mathbf{v} = [1, 1]$ . The translating-square sequence contains 24 frames, which is about the number of frames movies display within a second (Fig. 2A, left panel). The global Fourier transform is taken over the whole sequence, as explained previously (Section II), and the weights are computed for each point of frame 12 of the sequence. We compute the map of votes for velocities  $\mathbf{U}$ , where the component velocities in  $x$  and  $y$  direction are ranging from  $-3$  to  $3$  pixels/frame in steps of  $0.1$  pixels/frames. More detailed information about the implementation of the algorithm is given in the Appendix. The same parameters  $\xi = 0.3$  and  $\sigma = 0.6$  are used for all sequences. The map of votes for an image point belonging to the square (labeled a) is presented in the middle panel of Fig. 2A. A clearly distinguishable peak is visible in the map of votes, with a maximum positioned at  $\mathbf{U} = [1, 1]$ .

In contrast, point b in the surround of the square, which is a homogenous image region, returns a flat map of votes, and no clear peak can be distinguished (Fig. 2A, right panel). This is reflected in the confidence  $G(\mathbf{x})$  of the velocity estimate, presented in Fig. 2B, left panel. All points on the square have a high confidence, while points in the homogenous surround have not.

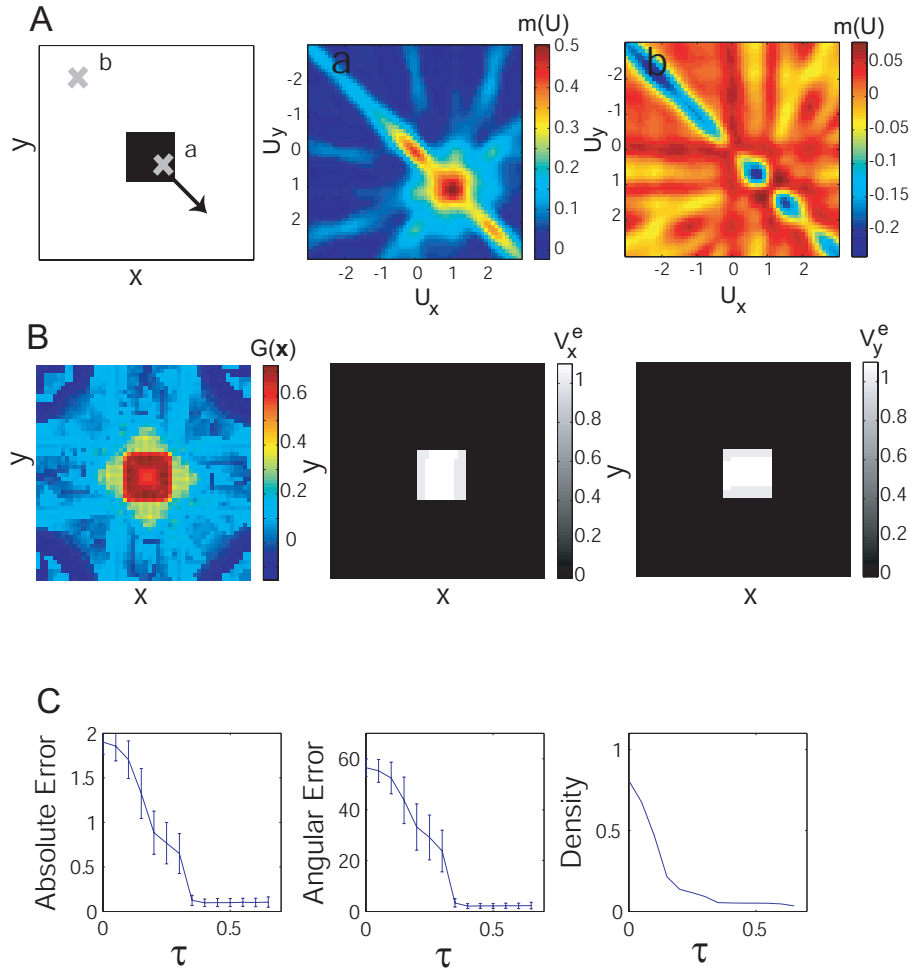


Fig. 2. Translating square. **A** Schematic (left panel). Map of votes for point  $a$  of the square, showing a clear peak (middle panel). Map of votes for point  $b$  in the homogeneous surround (right panel). **B** The confidence  $G(\mathbf{x})$  as a function of position (left panel). The estimated velocities in  $x$  direction (middle panel), and in  $y$  direction (right panel). **C** Absolute error  $\bar{E}_a$ , angular error  $\bar{E}_\phi$  and density  $\rho$  as a function of the threshold  $\tau$ .

Thresholding  $G(\mathbf{x})$  with  $\tau = 0.4$ , returns only velocity estimates which have a high confidence. The  $x$  and  $y$  components of the estimated velocities are given in Fig 2B, middle and right panel, respectively. The algorithm gives estimates for points lying on the square. Remarkably, the estimated velocities of points belonging to the edge of the square do not suffer the aperture problem, since the Fourier transformation is entirely global. However, velocity estimates for points in the homogenous surround cannot be given. This merely demonstrates the ambiguity of the vision problem, not a insufficiency of the algorithm, since homogenous regions can be

interpreted as moving with an arbitrary velocity. In other words, a homogeneous background contains all possible velocities.

To quantify the performance of the algorithm, we compute the absolute error and the angular error as a function of  $\tau$ , together with the standard error (Fig. 2C, left and middle panel, respectively). In Fig. 2C, right panel, the density of the sequence is plotted as function of  $\tau$ . The error measures decrease with increasing threshold and settle at low values close to the resolution of the map of votes of 0.1 pixels/frame.

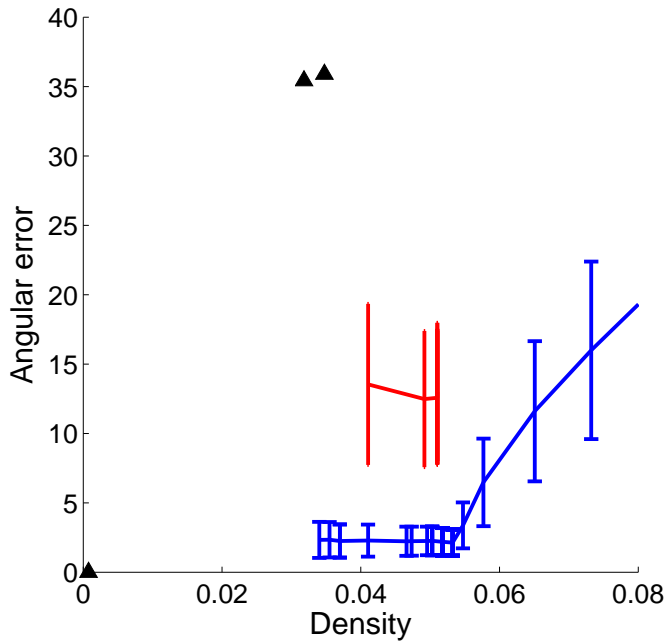


Fig. 3. Performance of optic-flow techniques for the translating-square sequence: Lucas and Kanade (black triangles), Gautama and Van Hulle (red), and our approach (blue).

We compare the performance of the algorithm with the outcome of other optic-flow techniques, a phase-based approach by Gautama and Van Hulle [9] and a differential technique by Lucas and Kanade [8] (Fig. 3). For our algorithm, we achieved decreasing densities by increasing the threshold  $\tau$ . For the phase-based approach, we use parameters as in [9], namely  $N_{min} = 7$  and  $\tau_l$  ranging from 0.001 to 0.02 in steps of 0.002. The choice of the parameters however does not significantly alter the result of the algorithm. The angular error is above the angular error of our global approach. The angular error of the Lucas and Kanade algorithm for a window

size of  $5 \times 5$  pixels<sup>2</sup> reaches a lower value than our algorithms, however, the density is very low. Different densities were obtained by thresholding the absolute gradient squared, leading to maximum confidence values at the corners of the square. At higher densities though, the performance of the Lucas and Kanade algorithm is less good than our approach. The large error at higher densities for the Lucas and Kanade algorithm is due to the aperture problem, which produces wrong velocities at the edge of the square. At very low densities, only the corners of the square are left, which are not afflicted with the aperture problem due to their higher intrinsic dimensionality [10]. The optic-flow algorithm by Gautama and Van Hulle produces good velocity directions at the edges because the method uses various spatial filters. The angular error is above our approach and below the angular error of Lucas and Kanade.

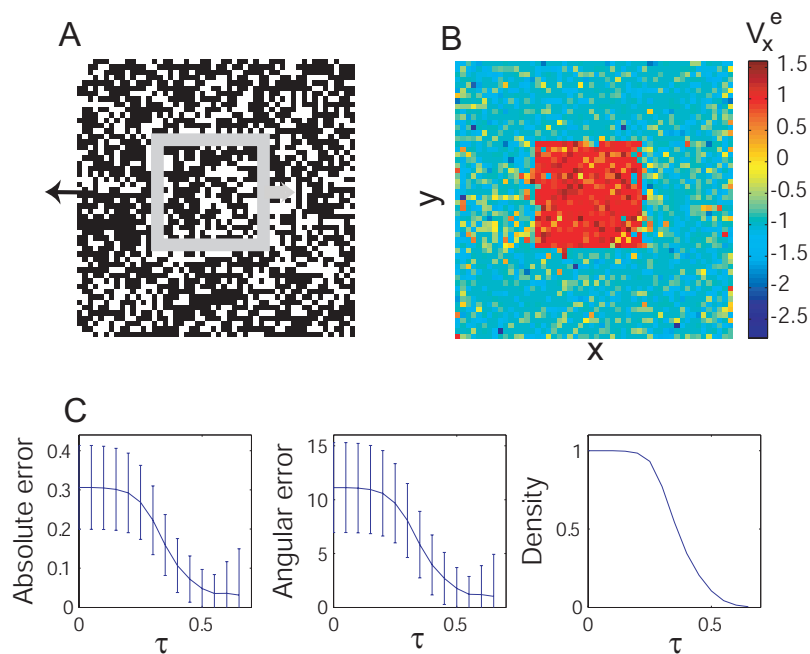


Fig. 4. Camouflaged square. **A** Schematic of the sequence. **B** Estimated  $x$  velocities. **C** Absolute error  $\bar{E}_a$ , angular error  $\bar{E}_\phi$  and density  $\rho$  as a function of the threshold  $\tau$ .

The translating-square sequence presented in this paper differs in the number of frames relative

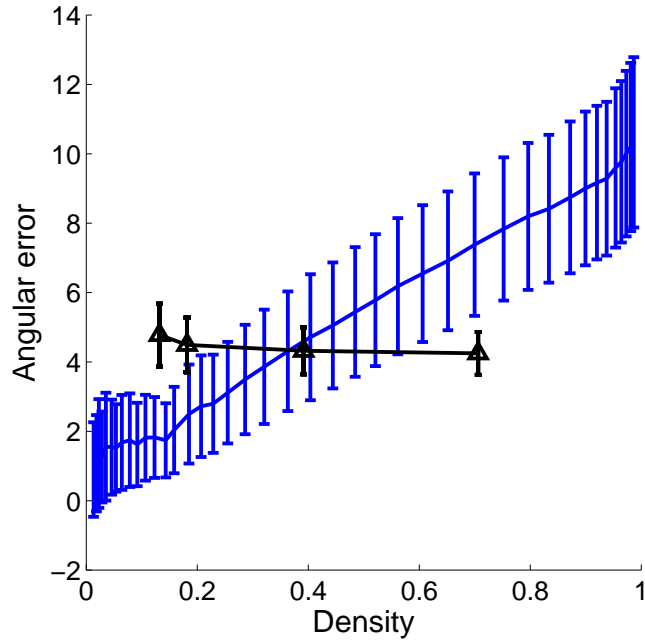


Fig. 5. Performance of optic-flow techniques for the camouflaged-square sequence: Lucas and Kanade (black) and our approach (blue).

to the size of the square from the Square2 sequence used by Barron et al. [1]. The Square2 sequence contains 20 frames. The  $40 \times 40$  pixels<sup>2</sup> large square moves with a velocity of  $\mathbf{v} = (-4/3, -4/3)$  pixels/frame, thus, the center of the square is mainly composed of static gratings. It is therefore less suited for demonstrating the characteristic properties of our global approach. However, our algorithm returns good velocity estimates for the edges of the square with a low angular error approaching 1 deg for low densities. The angular errors of other techniques computed for the Square2 sequence range from 49.03 deg to 0.07 deg [1]. The interested reader is directed to the Appendix, where the results of the algorithm for the Square2 sequence are shown. Ranking of techniques based on the angular error is difficult, since the performance of an algorithm can vary significantly for different image sequences. Also, the quality of an optic-flow algorithm is not only characterized by its angular error, but also by the density of the resulting flow field and whether velocities can be computed for the “interesting” parts of the image, e.g., independently moving objects.

### B. Camouflaged-square sequence

Here, a square composed of random dots moves with a velocity of  $\mathbf{v} = (1, 0)$  in front of a random-dot background which moves in the opposite direction with  $\mathbf{v} = (-1, 0)$ . Thus, the random-dot square is only defined by its motion relative to the background. A schematic of the sequence is given in Fig. 4A. The estimated velocities (unthresholded) are presented in Fig. 4B. The motion-defined square pops out with sharp borders and is clearly distinguishable from the background. The error measurements show decreasing error with increasing threshold settling at the velocity resolution of the map of votes (Fig. 4C).

We again compare the performance of the algorithm with the outcome of the algorithms by Gautama and Van Hulle [9] and Lucas and Kanade [8], using the same parameters as before. The performance of the algorithm by Gautama and van Hulle for this particular sequence is poor with a mean angular error of  $\bar{E}_\phi = 56 \pm 219$  deg, which could not be improved by changing parameters. The algorithm by Lucas and Kanade gives good results with a performance which is largely independent of the density and is thus less accurate at low densities than our approach (Fig. 5). The error of the Lucas and Kanade algorithm is due to poor spatiotemporal localization of velocities at the boundaries of the motion-defined camouflaged square.

### C. Translating- and Diverging-Tree Sequence

The translating- and diverging-tree sequences are artificial sequences generated from a real image of a tree (Fig. 6A). In the translating-tree sequence, the camera moves to the right, while filming a picture of a tree, which is slanted with respect to the fronto-parallel plane. The velocities are parallel to the  $x$ -axis and range from 1.73 to 2.26 pixels/frame. The estimated flow field for the raw translating-tree sequence for  $\tau = 0.55$  is shown in Fig. 6B, left panel. In order to selectively compute estimates at edge-like image regions, we preprocess the image sequence with a high-pass Butterworth spatial frequency filter with a cut-off value of  $D_0 = 0.1$ . The resulting optic-flow field for  $\tau = 0.55$  is shown in Fig. 6B, middle panel. In another example, we preprocess the sequence with a high-pass velocity filter with  $\tau_v = 0.1$  and cut off all intensity values close to the mean intensity with  $\tau_I = 0.1$ , as described in Section 2. The resulting optic-flow field for  $\tau = 0.55$  is shown in Fig. 6B, right panel.

In the diverging tree sequence, the camera recedes from the tree image, causing the optic-flow field to diverge from the center of expansion which is located in the middle of the image. The

speeds range from 1.29 to 1.86 pixels/frame. The estimated flow field for the raw sequence, the high-pass spatial frequency filtered, and the high-pass velocity filtered sequences are shown in Fig. 6C, from right to the left, respectively. The parameters are the same as for the translating tree sequence.

The performance of the algorithm for the translating- and diverging-tree sequence is evaluated in Fig. 7, A and B, respectively. The angular error for the raw, the high-pass spatial frequency filtered, and the high-pass velocity filtered translating tree sequence are plotted in blue, red, and black, respectively, as a function of the density of the estimated flow field (Fig. 7A). Decreasing densities are obtained by increasing the threshold  $\tau$ . At low densities, the algorithm produces the lowest error for the high-pass velocity filtered sequence. The angular error of the high-pass spatial frequency filtered sequence is above the angular error of the high-pass velocity filtered sequence and below the angular error of the raw sequence.

The angular error for the raw, the high-pass spatially filtered, and the high-pass velocity filtered diverging tree sequence are plotted in blue, red, and black, respectively, as a function of the density of the estimated flow field (Fig. 7B). Here, best results are obtained for the high-pass spatial frequency filtered image sequence. The angular error of the high-pass velocity filtered sequence is above the angular error of high-pass spatial frequency filtered sequence and below the angular error of the raw sequence. The error statistics for the example of the translating and diverging tree sequence demonstrate that preprocessing can improve optic-flow estimation significantly. Angular errors have been measured for a variety of optic-flow techniques for the both the translating and diverging tree sequence. The angular errors of the different techniques range from 17.66 – 0.73 deg. The tables can be looked up in [1].

#### D. Real sequences

We estimated the optic-flow fields for four different real image sequences, the SRI-trees sequence, the Rubic-cube sequence, the Hamburg taxi sequence, and the walking sequence. All sequences were preprocessed with a high-pass velocity filter and the intensity values were thresholded with  $\tau_I = 0.1$ .

In the SRI Sequence, a camera moves parallel to the ground plane along the  $x$ -axis in front of several trees. The velocities are as large as two pixels/frame. The sequence contains 20 frames. The color-coded velocities in  $x$  direction of frame 10 for  $\tau = 0.5$ ,  $\tau_v = 0.1$ , and  $\tau_I = 0.1$  are

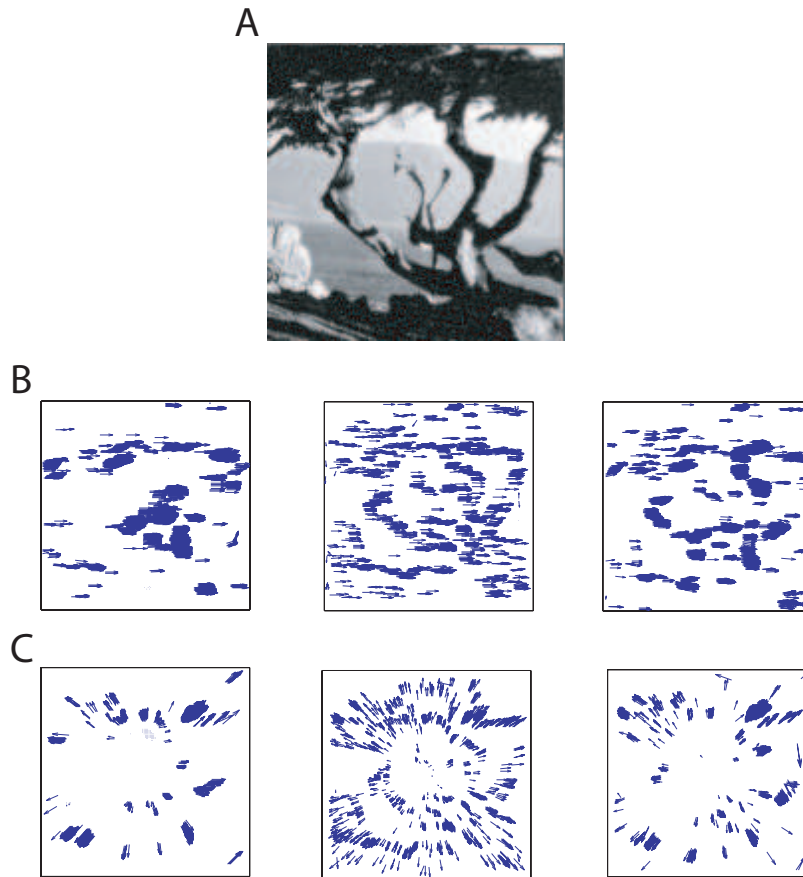


Fig. 6. Translating- and diverging tree sequence. **A** Snapshot of the original sequence. **B** Estimated flow for  $\tau = 0.55$  for the raw, the high-pass spatial frequency filtered, and the high-pass velocity filtered translating-tree sequence, left, middle, and right panel, respectively. **C** Estimated flow for  $\tau = 0.55$  for the raw, the high-pass spatially filtered, and the high-pass velocity filtered diverging tree sequence, left, middle, and right panel, respectively.

shown in Fig. 8A, right panel. Image points which are below threshold are set to zero. The tree in the foreground is clearly separable from the foreground and moving with a speed of about 2 pixels/frame and could be to large parts segmented from the background despite the poor resolution, the amount of occlusion, and the low contrast of the image sequence.

In the Rubic-cube sequence, a Rubic's cube is rotating counterclockwise on a turntable. The rotation induces velocities of the cube between 0.2 and 0.5 pixels/frame. The velocities on the turntable range from 1.2 to 1.4 pixels/frame. This sequence contains 20 frames. The optic-flow field at frame 10 for  $\tau = 0.5$ ,  $\tau_v = 0.1$ , and  $\tau_I = 0.1$  is shown in Fig. 8B, middle panel. An

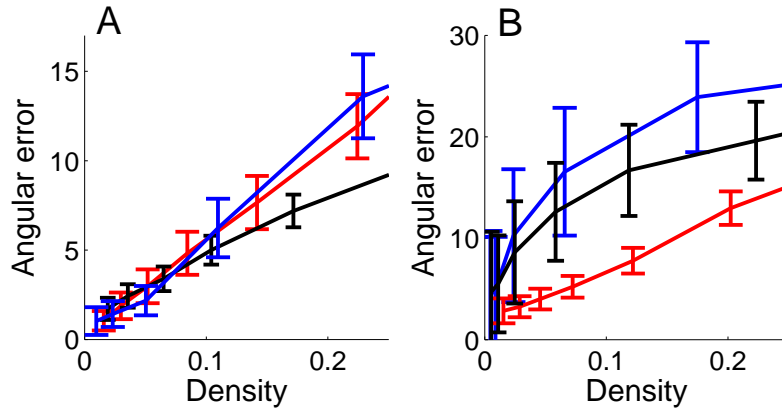


Fig. 7. **A** Translating tree sequence: Angular error  $\bar{E}_\phi$  as a function of image density  $\rho$  for the raw sequence (blue), the high-pass spatial frequency filtered sequence with  $D_0 = 0.1$  cycles/deg (red), the high-pass velocity filtered sequence with  $\tau_v = 0.1$  pixels/frame and  $\tau_I = 0.1$  (black). **B** Diverging tree sequence: Angular error  $\bar{E}_\phi$  as a function of image density  $\rho$  for the raw sequence (blue), the high-pass spatial frequency filtered sequence with  $D_0 = 0.1$  cycles/deg (red), the high-pass velocity filtered sequence with  $\tau_v = 0.1$  pixels/frame and  $\tau_I = 0.1$  (black).

enlarged part of the optic-flow field is given in Fig. 8B, right panel. The optic-flow field is scaled by a factor 2 for reasons of better display. This sequence demonstrates that our method is capable of treating rotational motion.

In the Hamburg taxi sequence, a street scene is shown with four moving objects: (i) a taxi turning the corner (ii) a car in the lower left, driving from the left to the right (iii) a van in the lower right driving from the right to the left, and (iv) a person walking in upper left. Image speeds of the four moving objects are approximately 1.0, 3.0, 3.0, and 0.3 pixels/frame, respectively. The sequence contains 20 frames. The optic-flow field at frame 10 for  $\tau = 0.5$ ,  $\tau_v = 0.1$ , and  $\tau_I = 0.1$  is shown in Fig. 8C, right panel. An enlarged part of the optic-flow field is given in Fig. 8C, right panel. The optic-flow field is scaled by a factor 2 for reasons of better display. by means of manual segmentation, we computed the average velocity for each moving object: (i)  $\bar{\mathbf{V}}_e = (-0.7, -0.7)$ , (ii)  $\bar{\mathbf{V}}_e = (2.7, 0.5)$ , (iii)  $\bar{\mathbf{V}}_e = (-2.6, -0.3)$ , and (iv)  $\bar{\mathbf{V}}_e = (-0.3, 0)$

pixels/frame. Fig. 8C demonstrates that all four objects are detected successfully by our method.

In the walking sequence, a woman walks from the left to the right. The sequence contains 40 frames. The optic-flow field for frame 20 for  $\tau = 0.35$ ,  $\tau_v = 0.2$ , and  $\tau_I = 0.1$  is shown in Fig. 8D, right panel. An enlarged part of the optic-flow field is given in Fig. 8D, right panel. The optic-flow field is scaled by a factor 2 for reasons of better display. Using the global method, velocities could be assigned even though the optic-flow field of the walking woman is changing rapidly with time and space.

In summary, we found that quantitatively the global approach computes velocity estimates with an error which is in the error range of standard optic-flow techniques, as demonstrated for the translating square, the camouflaged square, and the translating and diverging tree sequences. The results of the algorithm for the real image sequences could only be evaluated qualitatively, since no ground truth is available. The algorithm is capable to segment a scene into foreground and background (SRI-tree sequence), to compute rotational motion (Rubic-cube sequence), to detect independently moving objects (taxi sequence), and to compute the complex optic-flow field of a walking person.

## V. CONCLUSION

We proposed a novel algorithm for the computation of optic flow which is based on constructive interference of the global Fourier components in an image sequence. Since local information is reconstructed through superposition of global Fourier gratings and not by means of local windowing, the method is neither constrained by the uncertainty principle of signal processing nor does it suffer from the aperture problem. As a consequence, accurate velocity estimates can even be assigned to extended edges, as demonstrated for the translating-square sequence. Despite the global integrations, local spatiotemporal information is retained, which has the consequence that even sharp motion-defined boundaries can be recovered (camouflaged-square sequence). We evaluated the performance for different synthetic image sequences, including the translating- and diverging tree sequences, and found an angular error close to the velocity resolution of the algorithm, here 0.1 pixels/frame, an adjustable parameter of the algorithm. For the real image sequences, the performance of the algorithm could only be evaluated qualitatively. The approach is successful in motion-based scene segmentation, computing rotational optic flow, recovering independently moving objects, and even in finding the complex optic flow of a walking person.

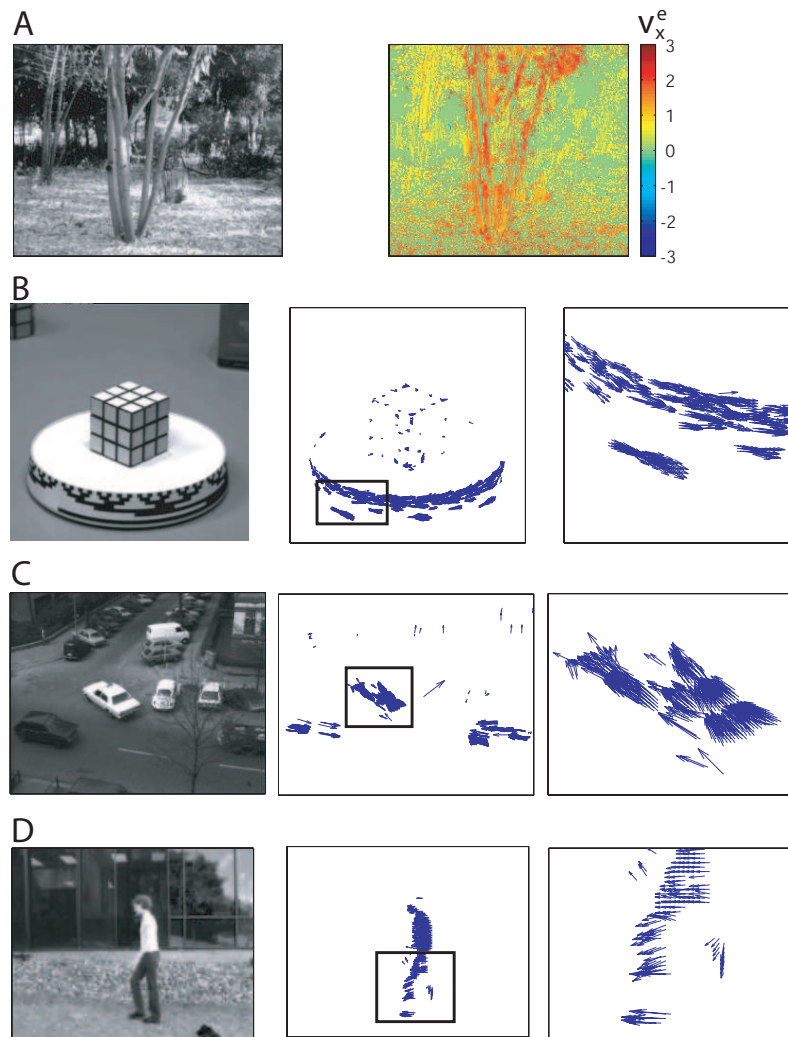


Fig. 8. Real image sequences. **A** SRI-trees. Frame 10 of the original sequence, left panel. The color-coded velocities in  $x$  direction of frame 10 for  $\tau = 0.5$ ,  $\tau_v = 0.1$ , and  $\tau_I = 0.1$ , right panel. **B** Rubic cube. Frame 10 of the original sequence, left panel. Estimated optic-flow field at frame 10 for  $\tau = 0.5$ ,  $\tau_v = 0.1$ , and  $\tau_I = 0.1$ , middle panel. Enlarged area of the optic-flow field (see inset, middle panel) is given in the right panel. **C** Hamburg taxi. Frame 10 of original sequence, left panel. Estimated optic-flow field at frame 10 for  $\tau = 0.5$ ,  $\tau_v = 0.1$ , and  $\tau_I = 0.1$ , right panel. An enlarged part of the optic-flow field (see inset, middle panel) is given in the right panel. **D** Walking. Frame 20 of the original sequence, left panel. Estimated optic-flow field for frame 20 for  $\tau = 0.35$ ,  $\tau_v = 0.2$ , and  $\tau_I = 0.1$ , middle panel. An enlarged part of the optic-flow field (see inset, middle panel) is given in the right panel.

Many methods for the computation of optic flow have been suggested, e.g., differential methods [8], [11], [12], region-based matching [13], [14], energy-based [15], and phase-based techniques

[9], [16], [17]. Other techniques include the “Skin and Bones” algorithm and the “Total Least Squares-Based Optic Flow” [18], [19]. By comparing the performance of different methods for optic-flow computation, Barron *et al.* [1] found that the first-order, local differential techniques by Lucas and Kanade [8] and the phase-based approach by Fleet and Jepson [17] were most reliable [1]. Galvin *et al.* claimed that the algorithm by Lucas and Kanade [8] gave the best results [2]. However, all of these methods compute the velocity of image points from the intensity pattern of local space-time windows around the points. Then, basic measurements such as spatiotemporal derivatives or local correlation surfaces are extracted and integrated to estimate the optic-flow field. As a consequence of the windowing operation, these methods are afflicted with the uncertainty principle and the aperture problem. For this reason, the algorithm by Lucas and Kanade returns erroneous velocity estimates at the bounding edge of the square in the translating-square sequence. The phase-based approach by Gautama and van Hulle performed well in this case, however, it failed to compute an optic-flow field for the camouflaged square sequences, where accurate spatiotemporal localization of velocity estimates is important. Our global approach gave good results for both of sequences, demonstrating that accurate velocity estimates can be assigned without loss of spatiotemporal precision if using constructing interference of global signals. However, it is difficult to rank different techniques because the performance of the algorithms varies dependent on the image sequence and on the measure used to quantify success, i.e., a low angular error can be utterly meaningless if the image points for which the velocity estimates have been computed are not of interest for solving a particular task.

Our method opens a new perspective on the computation of optic flow and on the interference of local measurements from global signals in general. For example, constructive interference could be also utilized for disparity estimation. The concrete implementation of the algorithms contains several toeholds at which the method could be improved, i.e, transparent motions could be detected and segmented by using more elaborated methods for reading out the velocity from the map of votes.

#### ACKNOWLEDGMENT

The work has received support from the EU Project Drivscio under Contract No. 016276-2 and the EU Project PACO-PLUS under Contract No. 027657.

## REFERENCES

- [1] J.L. Barron, D.J. Fleet, S. Beauchemin, and T. Burkitt, "Performance of optical flow techniques," *Int. J. Comput. Vis.*, vol. 12, no. 1, pp. 43-77, 1994.
- [2] B. Galvin, B. McCane, K. Novins, and S. mills, "Recovering motion fields: An evaluation of eight optical flow algorithms," in *Proc. British Machine Vision Conf.*, 1998, pp. 195-204.
- [3] R. Wilson, G.H. Granlund, "The uncertainty principle in signal processing," *IEEE Trans. Pat. An. Mach. Intel.*, vol. 6, 1984.
- [4] D. Vernon, "Decoupling Fourier components of dynamic image sequences: a theory of signal separation, image segmentation, and optical flow estimation," *Lect. Not. in Comp. Sci*, Springer Berlin/Heidelberg, Computer Vision – ECCV'98, pp. 69, 1998.
- [5] B.K. Dellen, J.W. Clark, and R. Wessel, "The brain's view of the natural world in motion: Computing structure from function using directional Fourier transformations," *Int. J. Mod. Phys. B. Special Issue*, in press.
- [6] A. Briassouli, A. Narendra, "Spatial and Fourier error minimization for motion estimation and segmentation." *Proceedings of the 18th International Conference on Pattern Recognition, ICPR 2006*, August 20-24, Hong Kong, 2006.
- [7] E.H. Adelson and J.R. Bergen, "Spatiotemporal energy models for the perception of motion," *J. Opt. Soc. Am. A*, vol. 2, pp. 284-299, 1985.
- [8] B.D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision. *Proc. DARPA IU Workshop*, pp. 121-130, 1981.
- [9] T. Gautama and M. Van Hulle, "A phase-based approach to the estimation of the optical flow field using spatial filtering," *IEEE Trans. Neural. Net.*, vol. 13, pp. 1127-1136, 2003.
- [10] N. Krüger and M. Felsberg, "A continuous formulation of intrinsic dimension," *Proceedings of the British Machine Vision Conference*, 2003.
- [11] B.K.P Horn and B.G. Schunck, "Determining optic flow," *AI*, vol. 17, pp. 185-204, 1981.
- [12] H.H. Nagel, "Displacement vectors derived from second-order intensity variations in image sequences," *CGIP*, vol. 21, pp. 85-117, 1987.
- [13] P. Anandan, "A computational framework and an algorithm for the measurement of visual motion," *Int. J. Comp. Vision*, vol. 2, pp. 283-310, 1989.
- [14] A. Singh, "An estimation-theoretic framework for image-flow computation," *Proc. IEEE ICCV*, Osaka, pp. 168-177, 1990.
- [15] D.J. Heeger, "Model for the extraction of image flow," *J. Opt. Soc. Am. A*, vol. 4, pp. 1455-1471, 1987.
- [16] A.M. Waxman and K. Wohn, "Contour evolution, neighbourhood deformation and global image flow: Planar surfaces in motion," *Int. J. Rob. Res.*, vol. 4, pp. 95-108, 1985.
- [17] D.J. Fleet and A.D. Jepson, "Computation of component image velocity from local phase information," *Int. J. Comp. Vision*, vol. 5, pp. 77-104, 1990.
- [18] S.X. Ju, M.J. Black, and A.D. Jepson, "Skin and bones: Multi-layer, locally affine, optical flow and regularization with transparency," *Proc. IEEE Comput. Vision Pattern Recognition Conf.*, San Francisco, CA, pp. 307-314, 1996.
- [19] A. Bab-Hadiashir and D. Suter, "Robust optic flow computation," *Int. J. Comp. Vis.*, vol. 29, pp. 59-77, 1998.

**Babette Dellen** received the Diplom in physics from the University of Cologne, Germany, and the Ph.D. in philosophy in physics from Washington University in Saint Louis, USA. She is currently working as a postdoctoral researcher at the Bernstein Center of Computational Neuroscience in Göttingen, Germany, in the group of Prof. F. Wörgötter. Her research interests include computer vision, image processing, computational neuroscience, nonlinear dynamics, and biological modeling.

**Florentin Wörgötter** studied biology at the University Düsseldorf and received his Ph.D. 1988 from the Universitätsklinikum Essen. He obtained his Habilitation in Neuroscience 1993 from the “Medizinische Fakultät der Ruhr-Universität Bonn”. He also conducted research as a visiting postdoctoral scientist in the USA (1988 to 1990), in China (1994), and in Sweden (1997). From 2000 to 2005, he was with the Department of Psychology of the University of Stirling as a Full Professor. Today, he is Professor for Computational Neuroscience at the Bernstein Center for Computational Neuroscience in Göttingen.