Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT®

ACADEMIC PRESS

Computer Vision and Image Understanding 90 (2003) 129–152

Computer Vision and Image Understanding

www.elsevier.com/locate/cviu

Multiple motion analysis: in spatial or in spectral domain?

Weichuan Yu,[a,*] Gerald Sommer,[b] and Kostas Daniilidis[c]

[a] *Department of Diagnostic Radiology, Yale University, BML 332, P.O. Box 208042, New Haven, CT 06520-8042, USA*
[b] *Institute of Computer Science, Christian Albrechts University, Preusserstrasse 1-9, D-24105 Kiel, Germany*
[c] *GRASP Lab, University of Pennsylvania, 3401 Walnut Street, Suite 336C, Philadelphia, PA 19104-6228, USA*

Received 28 June 2001; accepted 16 January 2003

Abstract

In this paper, we compare the effects of multiple motions in spatial and spectral representations of an image sequence. We describe multiple motions in both domains and establish a comparison regarding their inherent properties when discretized. Though the spectral model provides us with an explicit description of both occlusion and transparency, it turns out that its resolution is very limited. We show that the spatial domain represented by the spatio-temporal derivatives has superior resolution properties and is thus more appropriate for the treatment of occlusion. We present an algorithm which based on an initial estimate of the number of motions uses the shift-and-subtract technique to localize occlusion boundaries and to track their movement in occlusion sequences. The same technique is used to distinguish occlusion from transparency and to decompose transparency scenes into multi-layers.

Keywords: Motion analysis; Multiple motion model; Image segmentation; Occlusion; Transparency

1. Introduction

The detection and estimation of multiple motions are challenging problems in the study of optical flow. Single motion estimation approaches are based on the

[*] Corresponding author. Fax: 1-203-737-4273.

E-mail addresses: weichuan@noodle.med.yale.edu (W. Yu), gs@ks.informatik.uni-kiel.de (G. Sommer), kostas@grasp.cis.upenn.edu (K. Daniilidis).

doi:10.1016/S1077-3142(03)00011-0

conservation of a response characterizing the local structure of images. For example, the well-known brightness change constraint equation (BCCE) [1]

$$I_x u + I_y v + I_t = 0 \tag{1}$$

is based on the conservation of intensity. Optical flow is denoted by (u, v) and the spatio-temporal derivatives by the triple (I_x, I_y, I_t). It is clear that such a constancy constraint models a single motion inside the neighborhood involved in estimation or inside the support of the filters producing the response to be conserved.

The detection of multiple motions can be addressed as segmentation problem. However, the optical flow-field segmentation problem is coupled with the estimation of the flow itself which is a well-known ill-posed problem. Such coupled estimation–segmentation problems have the chicken-and-egg dilemma inherent. If the flow were accurately given everywhere then we would be able to find the motion boundaries. But flow can be accurately estimated only if we know the motion boundaries and hence the exact neighborhood where the single motion assumption holds. A solid approach would be to formulate a global variation optimization problem where unknowns are both the flow and the region boundaries. Indeed, many approaches presented several variants of discontinuity preserving or anisotropic diffusion approaches [1–4].

In this paper, we are interested in the *local* detection and estimation of multiple motions. Local approaches can be divided in two groups depending on the domain they work: spatio-temporal and frequency domain.

1.1. Spatial approaches

Most of the spatial approaches are based on parametric models applied in large neighborhoods [5]. These face again the above-mentioned dilemma which they try to solve with successive iterations between segmentation and estimation. The most profound paradigm is expectation–maximization (EM), e.g. [6]. Bergen et al. [7] proposed an iterative method based on the shift-and-subtract strategy to estimate two motions. They subtracted pixels connected to one motion during the refinement of the parameters of the other motion and vice versa. Irani et al. [8] applied this iterative method in the temporal integration to blur out uninterested regions and to track objects even with non-consistent speeds.

In presenting explicit multiple motion models many researchers have made contributions. Wang and Adelson [9] represented multiple motions with a multi-layer model. The local motion estimation technique they used for estimation is still based on the BCCE. Fleet et al. [10] explicitly modeled an occlusion boundary in the spatial domain with a step function in both components of the optical flow field and used the steerability theory to detect the boundary. Black and Fleet further proposed to use the Bayesian framework to determine which pixels belong to the motion boundary regions and to choose an appropriate motion model (i.e., translational motion vs. occlusion motion) [11].

Black and Anandan [12] treated occlusion regions as *outliers* of the motion constraint and set lower weights to these regions in the estimation. The concept of an *outlier* represents exactly the relationship between the pixels near occlusion boundaries and the pixels with a single motion: The spatio-temporal partial derivatives of the pixels with a single motion form a plane in the derivative space and the derivatives of the pixels near occlusion boundaries deviate from this plane due to motion discontinuities. Based on this concept, probabilistic methods were proposed to model occlusion boundaries [13] and to estimate motions near occlusion boundaries [7,14]. Outliers were considered as noise in statistic methods (e.g. [15]) as well as in the Hough transform-based approaches (e.g. [16]).

1.2. Spectral approaches

Motion estimation was also addressed from the point of view of orientation analysis in the spectral domain. Adelson and Bergen [17] pointed out that motion is equivalent to spatio-temporal orientation and introduced a spatio-temporal energy model for single motion representation. This was the first optical flow algorithm based on the spectral analysis.

Bigün and coworkes [18,19] connected the orientation analysis with symmetry detection. They pointed out that a single motion can be described as a linear symmetric image, whose spectrum is a line passing through the origin in the frequency domain. They fixed the orientation of the spectral line by minimizing a moment measure in the frequency domain. Shizawa and Mase [20] proposed a simple superposition principle. Fleet and Langley [21,22] as well as Beauchemin and Barron [23] further analyzed the spectral structure of occlusion and transparency. Jähne used a 3D structure tensor [24] to detect symmetry and to estimate motion [25]. He further introduced an eigenvalue-based coherence measure to distinguish different kinds of motions such as single constant motion and motion discontinuities.

In the following, we will display that occlusion is equivalent to multiple planes with some distortions both in the derivative space and in the frequency domain; transparency can be described as multiple planes without distortions only in the spectral domain [23,26]. The corresponding motion parameters are determined by the normal vectors of these planes. Determining the precise orientation of two motion planes, however, remains a difficult task because the angle between two motion planes can be arbitrary. Bigün et al. [19], Shizawa et al. [20], and Jähne [25] used the principal axis analysis, which is also called principal component analysis (PCA), Karhunen-Loève transform (KLT), and Hotelling transform in different literature. It decomposes a signal into an orthogonal basis using eigenvector analysis or singular value decomposition (SVD). The problem of principal axis analysis is that it is only appropriate to detect one dominant orientation because the largest eigenvector is always orthogonal to the other eigenvectors, no matter what kind of structure the signal has. In other words, the orientation resolution of the principal axis analysis is not sufficient to solve a non-orthogonal multiple orientation problem.

1.3. Hybrid approaches

In order to determine the orientation of the motion plane in the frequency domain and avoid the discrete Fourier transform, Heeger sampled the spectrum of the image sequence with 12 Gabor filters in the spatial domain [27]. Similarly, to sample the spectrum for motion estimation [28], Xiong and Shafer used a basis of confluent hypergeometric functions [29].

Grzywacz and Yuille [30] further pointed out that the orientation uncertainty depends on the angular support of a filter which is the angle between two tangential lines of the support passing through the spectral origin. In orientation analysis, the filters at different frequencies are desired to have the same angular uncertainty, which is exactly the property of Gabor wavelets [31,32].

One main concern of Gabor/hypergeometric filter-based approaches is the enormous complexity of computation in sampling the spectral domain with fine resolution. Another concern is the positive skewness in the filter responses of the Gabor wavelets [30]. However, we do not address these points in this paper.

1.4. Our contribution

In this paper, we focus on the comparison between spatial and spectral models of multiple motions. We argue that though the spectral motion model describes both occlusion and transparency in a uniform manner, the spatial model is more appropriate for occlusion analysis because it provides finer resolution and requires less frames in the sequence than the spectral motion model. The shift-and-subtract technique is also applied to localize and to track motion boundaries.

This paper is constructed as follows: The following section studies occlusion and transparency in detail for a better understanding of multiple motions. We also compare motion models in spatial and in spectral domain. In Section 3 we introduce a general framework for multiple motion analysis. As the first step, we characterize motions with a spherical signature which is used to distinguish occlusion from transparency. This signature also provides us with a reasonable initial value close to the correct solution. In the estimation step, we mainly address the outlier issue in occlusion estimation. After that we use the shift-and-subtract technique to localize and to track occlusion boundaries. Section 4 shows some experiment examples. Finally, this paper is concluded in Section 5.

2. Understanding multiple motions

2.1. Spatial observation of multiple motions

Both occlusion and transparency can be decomposed into multiple layers [9]. But decompositions are based on different principles. We illustrate this difference in Fig. 1, where occlusion is more local than transparency in the spatial domain: while occlusion involves a step-function at the occlusion boundary, transparency results from the overlapping of two motions in the entire window. This difference makes it difficult to describe both kinds of multiple motions using a unified model in the spatial domain. Concretely, while occlusion can be characterized as multiple planes in the derivative space [20] with some outliers caused by the incorrect derivatives near

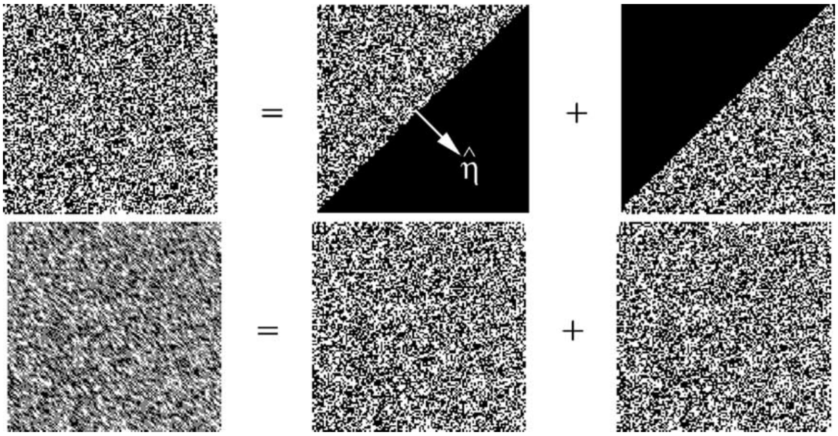


Fig. 1. Difference between occlusion and transparency. Here random dot regions represent motions and dark regions denote static areas. The occluding signal is moving with the speed $(1, 1)$ [pixel/frame] and the occluded signal with $(1, -1)$ and the speeds of transparent motions are $(1, 1)$ and $(1, -1)$ as well. *Top:* One frame of the occlusion sequence is decomposed into two layers by a Heaviside unit step-function (Eqs. (2) and (3)). There is motion discontinuity only at the boundary. The term $\hat{\eta}$ denotes the unit vector normal to the occluding boundary. *Bottom:* One frame of the transparency sequence is a simple superposition of two layers (Eq. (10)). Multiple motions exist in the entire window.

occlusion boundaries, transparency cannot be described as multiple planes in the derivative space because the *global* overlapping results in an erroneous estimation of derivatives in the entire window. This limitation was the motivation for a better model in the frequency domain.

2.2. Spectral analysis of occlusion

The spectrum of multiple motions was first analyzed by Fleet and Langley [22]. Assuming that an occlusion boundary is a characteristic function $\chi(\mathbf{x})$, they modeled the occlusion in the spatial domain as follows:

$$I(\mathbf{x}, t) = \chi(\mathbf{x} - \mathbf{v}_1 t) I_1(\mathbf{x} - \mathbf{v}_1 t) + [1 - \chi(\mathbf{x} - \mathbf{v}_1 t)] I_2(\mathbf{x} - \mathbf{v}_2 t), \tag{2}$$

where $\mathbf{x}$ denotes 2D spatial Cartesian coordinates, $I_1(\mathbf{x})$ is a 2D *occluding* signal moving with velocity $\mathbf{v}_1 = (u_1, v_1)^{\mathrm{T}}$ and $I_2(\mathbf{x})$ is a 2D *occluded* signal moving with velocity $\mathbf{v}_2 = (u_2, v_2)^{\mathrm{T}}$.

Beauchemin and Barron [23] were the first who formulated an exact model in the frequency domain. They modeled the occlusion in the spatial domain with a Heaviside unit step-function $U(\mathbf{x})$ for $\chi(\mathbf{x})$:

$$U(\mathbf{x}) = \begin{cases} 1, & \mathbf{x}^{\mathrm{T}} \hat{\eta} \geqslant 0, \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

with $\hat{\eta}$ denoting a unit vector normal to the occluding boundary. Eq. (2) reads then

$$I(\mathbf{x}, t) = U(\mathbf{x} - \mathbf{v}_1 t) I_1(\mathbf{x} - \mathbf{v}_1 t) + I_2(\mathbf{x} - \mathbf{v}_2 t) - U(\mathbf{x} - \mathbf{v}_1 t) I_2(\mathbf{x} - \mathbf{v}_2 t), \tag{4}$$

We denote the spatial frequency vector as $\mathbf{k} = (\omega_x, \omega_y)^{\mathrm{T}}$ and the temporal frequency as ω_{t}. Using the convolution and shift theorems we obtain the Fourier transform of Eq. (4) as

$$\begin{aligned}\tilde{I}(\mathbf{k},\omega_{\mathrm{t}}) &= \tilde{U}(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_1+\omega_{\mathrm{t}}) * \tilde{I}_1(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_1+\omega_{\mathrm{t}}) + \tilde{I}_2(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_2+\omega_{\mathrm{t}}) \\ &\quad - \tilde{U}(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_1+\omega_{\mathrm{t}}) * \tilde{I}_2(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_2+\omega_{\mathrm{t}}), \qquad (5)\end{aligned}$$

where $*$ means convolution and $\tilde{}$ denotes the Fourier transform of the corresponding signal. The Dirac function $\delta(\mathbf{k},\omega_{\mathrm{t}})$ is caused by the motion in the sequence. The spectrum of the Heaviside unit step-function is a 2D Dirac function located at the origin plus a 2D hyperbolic line

$$\tilde{U}(\mathbf{k}) = 2\pi\left[\pi\delta(|\mathbf{k}|) + \frac{\delta(\mathbf{k}^{\mathrm{T}}\hat{\eta}_{\perp})}{\mathrm{i}\mathbf{k}^{\mathrm{T}}\hat{\eta}}\right]. \qquad (6)$$

Here $\hat{\eta}_{\perp}$ denotes a unit vector perpendicular to $\hat{\eta}$. Taking the properties of the impulse function into account we obtain (see [26,33] for the lengthy derivation)

$$\begin{aligned}\tilde{I}(\mathbf{k},\omega_{\mathrm{t}}) &= \left\{\left[2\pi^2\delta(|\mathbf{k}|) + \frac{2\pi\delta(\mathbf{k}^{\mathrm{T}}\hat{\eta}_{\perp})}{\mathrm{i}\mathbf{k}^{\mathrm{T}}\hat{\eta}}\right] * \tilde{I}_1(\mathbf{k})\right\}\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_1+\omega_{\mathrm{t}}) + \tilde{I}_2(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_2+\omega_{\mathrm{t}}) \\ &\quad - \left[2\pi^2\delta(|\mathbf{k}|) + \frac{2\pi\delta(\mathbf{k}^{\mathrm{T}}\hat{\eta}_{\perp})}{\mathrm{i}\mathbf{k}^{\mathrm{T}}\hat{\eta}}\right]\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_1+\omega_{\mathrm{t}}) * \tilde{I}_2(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_2+\omega_{\mathrm{t}}) \\ &= [2\pi^2\tilde{I}_1(\mathbf{k}) + A(\mathbf{k})]\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_1+\omega_{\mathrm{t}}) + (1-2\pi^2)\tilde{I}_2(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_2+\omega_{\mathrm{t}}) + B(\mathbf{k},\omega_{\mathrm{t}}) \\ &\qquad (7)\end{aligned}$$

with

$$A(\mathbf{k}) = \frac{2\pi}{\mathrm{i}\mathbf{k}^{\mathrm{T}}\hat{\eta}}\delta(\mathbf{k}^{\mathrm{T}}\hat{\eta}_{\perp}) * \tilde{I}_1(\mathbf{k}), \qquad (8)$$

$$B(\mathbf{k},\omega_{\mathrm{t}}) = \frac{\mathrm{i}2\pi}{\mathbf{k}^{\mathrm{T}}\hat{\eta}}\delta(\mathbf{k}^{\mathrm{T}}\hat{\eta}_{\perp})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_1+\omega_{\mathrm{t}}) * \tilde{I}_2(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_2+\omega_{\mathrm{t}}). \qquad (9)$$

The first two terms of expression (7) are two oriented planes passing through the origin of the frequency space ($\mathbf{k}^{\mathrm{T}}\mathbf{v}_1+\omega_{\mathrm{t}} = 0$ and $\mathbf{k}^{\mathrm{T}}\mathbf{v}_2+\omega_{\mathrm{t}} = 0$). Their normal vectors, namely $(\mathbf{v}_1, 1)$ and $(\mathbf{v}_2, 1)$ contain the velocities. The second term is the exact spectrum of the occluded signal. The first term contains an additional distortion term $A(\mathbf{k})$ on the plane of the occluding spectrum. However, here we are only interested in the orientation of the plane and the term $A(\mathbf{k})$ does not disturb the orientation. Actually, $A(\mathbf{k})$ strengthens the spectral plane $\mathbf{k}^{\mathrm{T}}\mathbf{v}_1+\omega_{\mathrm{t}} = 0$. Therefore, we do not consider it as distortion. The main discriminating term is the third one, $B(\mathbf{k},\omega_{\mathrm{t}})$. It is a convolution between a 3D spectral line passing through the origin whose amplitude changes hyperbolically and the spectrum of the occluded signal, which lies on the spectral plane $\mathbf{k}^{\mathrm{T}}\mathbf{v}_2+\omega_{\mathrm{t}} = 0$. On the occluded plane, each Dirac component shifts the center of the 3D spectral line from origin to its position and weights this line with the Dirac amplitude. As the occluded plane is composed of a set of differently weighted and located Dirac components, we view the distortion as the sum of differently weighted spectral lines with their roots on the occluded plane and with their orientation formed by $\delta(\mathbf{k}^{\mathrm{T}}\hat{\eta}_{\perp})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_1 + \omega_{\mathrm{t}})$.

If the energy of the distortion term $B(\mathbf{k}, \omega_{\mathrm{t}})$ is very high, we will not be able to recognize these two planes. Fortunately, the energy of the distortion decreases very quickly after leaving the occluded spectral plane due to the hyperbolic property of $\mathrm{i}2\pi/\mathbf{k}^{\mathrm{T}}\hat{\eta}$. The amplitude of distortion is much less than that of the signal in most regions of the spectrum outside the occluding and the occluded spectral plane. Thus, we still can use the information of energy distribution to fit dominant spectral planes except for the regions at low frequencies, where the determination of the orientation of spectral planes is more susceptible to distortion than in high frequency regions. As a recipe, we may consider the spectrum only above a lower frequency bound to improve the robustness of motion estimation.

2.3. Spectral analysis of transparency

Transparency may be viewed as a special case of occlusion by simply substituting $\chi(\mathbf{x} - \mathbf{v}_1 t)$ with a real constant a ($a \in (0, 1)$) [23]

$$I(\mathbf{x}, t) = aI_1(\mathbf{x} - \mathbf{v}_1 t) + (1 - a)I_2(\mathbf{x} - \mathbf{v}_2 t). \tag{10}$$

The corresponding spectrum is then characterized by two oriented planes without distortion

$$\tilde{I}(\mathbf{k}, \omega_{\mathrm{t}}) = a\tilde{I}_1(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_1 + \omega_{\mathrm{t}}) + (1 - a)\tilde{I}_2(\mathbf{k})\delta(\mathbf{k}^{\mathrm{T}}\mathbf{v}_2 + \omega_{\mathrm{t}}). \tag{11}$$

2.4. Spectral model of multiple motions

Though in the case of occlusion there exists a distortion term, most of the energy is on the two spectral planes due to the hyperbolic nature of the distortion term. Thus, both occlusion and transparency are characterized as multiple spectral planes passing through the origin, and the corresponding motion speeds are described by the normal vectors of these planes.

This model can be viewed as a generalization of the spatio-temporal energy model of single motion [17,27]. At first sight it is very similar to the work of Shizawa and Mase [20], who assumed that multiple motions are additive superpositions of two single motions in the frequency domain or in the derivative space [20]. But there are two distinct points in our work:

- Shizawa and Mase proposed that multiple motions are characterized as multiple planes both in the (I_x, I_y, I_t)-space and in the frequency domain. We argue that the description is not feasible in the (I_x, I_y, I_t)-space in the case of transparency because the points in the (I_x, I_y, I_t)-space are mixtures of two component motions and they do not lie on either of the two component motion planes.
- At low frequencies multiple planes are disturbed by the distortion term of the occlusion according to our analysis. We have to truncate low frequency components in order to fit multiple planes robustly.

2.5. Comparison between spatial and spectral model

According to the analysis in the above section, the assumption of multiple planes in the spectral domain describes both occlusion and transparency, while the same assumption in the spatial domain describes only occlusion. When we do not have a priori knowledge about motions, we should stay in the spectral domain.

It should be noticed that, though we have a thorough analysis of multiple motions in the spectral domain, there exists a severe problem in obtaining the energy spectrum of the image sequence due to the block effect of the discrete Fourier transform (DFT). To avoid the block effect of DFT, we take a local Fourier transform (LFT), i.e., DFT windowed by a Gaussian. According to the convolution theorem, Gaussian windowed DFT of the image sequence is equivalent to the convolution between the spectrum of the image sequence and the Gaussian function. Hence, the spectrum is blurred after the LFT and the resolution of the spectral model decreases. To increase resolution we could enlarge the window for the application of the LFT. But unfortunately, the constant motion assumption in this enlarged neighborhood is more fragile. Meanwhile, using a larger window means including more frames in the estimation, but we can hardly assume that the motion is constant over a very large time interval. Thus, for occlusion analysis we prefer to stay in the spatial domain and treat the derivatives near occlusion boundaries as *outliers*.

3. Multiple motion analysis algorithm

In this section, we introduce a concrete algorithm for multiple motion analysis. This algorithm can be divided into the following three consecutive steps:

1. Motion Characterization—determine the number of motions and the domain of parameter estimation.
2. Motion Estimation—estimate motion parameters of occlusion and transparency.
3. Scene Analysis—track the movement of occlusion boundaries or decompose transparency frames into multi-layers.

3.1. Motion characterization

Given a motion sequence, how do we know whether there are multiple motions? If there exist multiple motions, how do we determine whether they are occlusion or transparency? These questions need to be answered in the motion characterization step. Here we apply a new *conic filter* [34] either on spatio-temporal derivative space of a local neighborhood or on the local Fourier transform to obtain a 3D orientation signature of the local neighborhood in the motion sequence. This *conic filter* is a Gaussian function defined in local spherical coordinates and its Cartesian support has the shape of a truncated cone with axis in radial direction and very small angular support. The reader is referred to [34] for details of the *conic filter*. Here we focus on the filter response of motion planes. A tilted motion plane passing through the origin

$(0,0,0)$ in the 3D Cartesian coordinate system with a unit normal vector $\mathbf{n} = (n_1, n_2, n_3)^{\mathrm{T}}$ reads

$$xn_1 + yn_2 + zn_3 = 0. \tag{12}$$

Here the triple (x,y,z) represents either the derivative coordinates (I_x, I_y, I_t) or the spectral coordinates $(\omega_x, \omega_y, \omega_{\mathrm{t}})$. After converting the Cartesian coordinates into spherical coordinates (i.e., $(x,y,z) \rightarrow (r,\theta,\phi)$ and $(n_1,n_2,n_3) \rightarrow (1,\theta_n,\phi_n)$), we obtain

$$r\cos(\phi)\cos(\phi_n)\cos(\theta - \theta_n) + r\sin(\phi)\sin(\phi_n) = 0, \tag{13}$$

where θ and ϕ denote the azimuth and elevation angle of the plane and (θ_n, ϕ_n) denote the angles of the normal vector. The variable r is integrated out after applying a set of *conic filters*. Thus, the tilted motion plane is converted into a harmonic curve in the spherical coordinates

$$\cos(\phi)\cos(\phi_n)\cos(\theta - \theta_n) + \sin(\phi)\sin(\phi_n) = 0. \tag{14}$$

Both Eq. (12) and Eq. (14) resemble the BCCE constraint (cf. Eq. (1)). Correspondingly, the motion parameters (u, v) are determined by the normal vector of the plane either in the Cartesian coordinates or in the spherical coordinates

$$\begin{aligned} u &= \frac{n_1}{n_3} = \cos(\theta_n)\cot(\phi_n), \\ v &= \frac{n_2}{n_3} = \sin(\theta_n)\cot(\phi_n). \end{aligned} \tag{15}$$

The angles (θ_n, ϕ_n) in the normal vector $\mathbf{n}$ are related to the maximal ϕ coordinate, ϕ_m, and the corresponding θ coordinate, θ_m, in the 3D orientation signature (cf. Fig. 2 and see Appendix A for derivation):

$$\begin{aligned} \theta_n &= \theta_m + 180°, \\ \phi_n &= 90° - \phi_m. \end{aligned} \tag{16}$$

The advantage of characterizing motions in the spherical coordinates is that each harmonic curve in the (θ, ϕ) space has two zero-crossing points on the θ axis with a

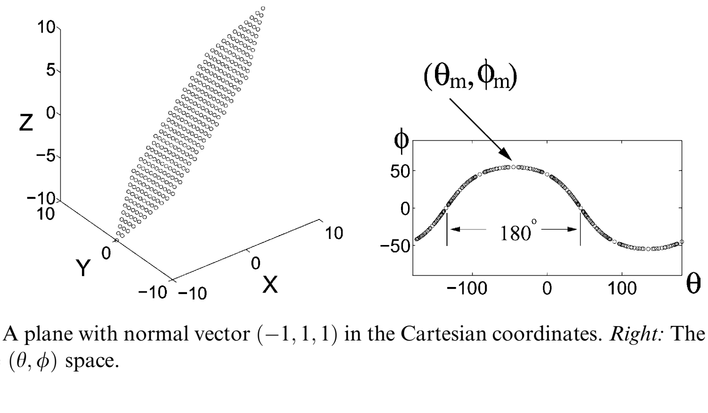


Fig. 2. *Left:* A plane with normal vector $(-1,1,1)$ in the Cartesian coordinates. *Right:* The corresponding curves in the (θ, ϕ) space.

distance of 180° and θ_m lies exactly in the middle of these two zero-crossing points (cf. Fig. 2). This extra geometric constraint is very useful in determining the number of motions automatically as well as in obtaining reasonable initial values of motion parameters. In this paper, we assume that there are at most two motions in an image sequence. The motion characterization step is then described as follows:

1.1. Apply the *conic filters* both on the spatio-temporal derivative space of a local neighborhood to obtain a spatial orientation signature and on the local Fourier transform to obtain a spectral orientation signature.

1.2. Fix a threshold parameter η and an energy threshold parameter λ_{e}. Add up the nonzero signature components in each orientation signature to obtain a total energy E_{tol}.

1.3. In each orientation signature, cluster the nonzero signature components near θ axis (i.e., $-\eta \leqslant \phi \leqslant \eta$) into the same group if their distance is less than 2η. If the centroids of two groups have a distance $\in [180^\circ - \eta, 180^\circ + \eta]$, these two groups form a group-pair.

1.4. For each group-pair of the orientation signature in the determined estimation domain, continue searching along the positive ϕ direction from their middle points (there are two middle points due to the circular angle distance along θ direction) and cluster the nonzero signature components with local maximal ϕ_m into different groups like in step 1.3. The weight-center of the vertical group gives us a guess of (θ_m, ϕ_m) and consequently an initialization of (θ_n, ϕ_n) (cf. Eq. (16)).

1.5. For each (θ_n, ϕ_n), add up the nonzero signature components satisfying

$$|\cos(\phi)\cos(\phi_n)\cos(\theta - \theta_n) + \sin(\phi)\sin(\phi_n)| \leqslant \eta \tag{17}$$

to obtain a curve energy E_{cur}.

```
if $(E_{\mathrm{cur}}/E_{\mathrm{tol}}) > \lambda_{\mathrm{e}}$ /* enough energy in the curve */
    set the corresponding $(\theta_n, \phi_n)$ pair as reliable and remove those components
    satisfying the inequality (17) before treating the next $(\theta_n, \phi_n)$ pair;
else /* not enough energy in the curve */
    set the corresponding $(\theta_n, \phi_n)$ pair as unreliable.
```

1.6. Compare the number $\#s$ of reliable (θ_n, ϕ_n) pairs in the spatial signature and the same number $\#p$ in the spectral signature to determine the number of motions and to determine in which domain we should estimate motions.

```
if $\#s = \#p = 1$ /* single motion */
    exit and use single motion model for estimation;
else if $\#s = \#p = 2$ /* occlusion */
    use spatial orientation signature for estimation;
else if $\#s$ arbitrary and $\#p = 2$ /* transparency */
    use spectral orientation signature for estimation;
else /* unknown case */
    break the characterization and exit.
```

In the above algorithm, the group-clustering method in steps 1.3 and 1.4 is fragile in the following specific situations:

- If images have no pixels satisfying $|I_t| \leqslant \eta$, the orientation signature will have no energy within the threshold distance of the θ axis and step 1.3 will fail. The same is true for images whose signatures have no component near expected (θ_m, ϕ_m) location, though this is less likely to happen for real images.

- If two motions move in the same direction with different speeds or in the completely opposite direction (the later case may be considered as a special example of moving in the same direction since we can change the sign of the speed components), the motion planes in the (I_x, I_y, I_t)-space form a *bow tie* structure [35]. Correspondingly, the curves in the (θ, ϕ) space will have the same θ at $\phi = 0$. Besides, the extreme points coordinated with (θ_m, ϕ_m) in two curves may be very close to each other which makes the group-clustering difficult, even when we use the energy thresholding method to help the clustering. The difference between these two extreme points depends both on two motion speeds and on the angular resolution of the conic filter. A careful analysis of their quantitative relation still needs to be conducted.

The aforementioned fragility of group-clustering may be avoided if we can find an elegant curve-fitting technique to extract the parameters of multiple curves. But this point still remains to be studied.

3.2. Motion estimation

We use Eq. (14) based EM algorithm to estimate multiple motions. The EM algorithm consists of subsequent iterations of the expectation and maximization step until there is no significant difference in the parameter estimates. In the expectation step, the membership weights of points are updated by the new results of parameter estimation; in the maximization step, we use the usual maximum likelihood method to estimate parameters with the updated assignment of points to groups. The reader is referred to [36] for details about the EM algorithm.

Since the EM algorithm is an iterative method, it has no closed-form solution. Generally, we do not know the number of motions exactly. Unlike other implicit constraints [37–39], the motion characterization step helps to determine the number of motions explicitly. Moreover, convergence and robustness of the EM algorithm are very much dependent on the initial values. Using the orientation signature we can facilitate a good initial value close to the correct solution.

Here we mainly address the outlier issue in occlusion estimation. Most current probabilistic estimation algorithms including the EM algorithm still include the outliers in the occlusion estimation. This makes the estimation fragile, especially if the number of outliers is comparable to the number of pixels with a single motion, since probabilistic methods are purely based on statistics. Our motivation is to improve the quality of input data before extracting motion parameters. Concretely, we can improve the precision of occlusion estimation if we can purify multiple planes from outliers (i.e., distortions). The remaining question is how to detect these outliers. The following two facts are proven to be useful in detecting outliers: First, if we have occlusion in a window, the occlusion boundaries should lie somewhere *inside* this window, though we do not know their exact positions. Second, we observe the following relations in the spatio-temporal derivative space according to [25]:

- For a single constant translational motion, the three eigenvalues of the motion plane satisfy

$$\sigma_1 \geqslant \sigma_2 > \sigma_3 = 0. \tag{18}$$

- For a single constant motion having the aperture problem, the plane above degenerates into a line whose corresponding eigenvalues satisfy

$$\sigma_1 > \sigma_2 = \sigma_3 = 0. \tag{19}$$

- For more complex motions all eigenvalues are positive

$$\sigma_1 \geqslant \sigma_2 \geqslant \sigma_3 > 0. \tag{20}$$

We can judge if there are multiple motions from different combinations of eigenvalues even *without* knowing motion parameters. Moreover, the eigen-analysis in [25] can work in a very small window (e.g., a 5 × 5 × 3 window) which facilitates the detection of outliers. Thus, we use a multi-window strategy to eliminate outliers before estimation. We detect outlier regions using small windows and mark these regions as outliers. In a large window containing these small windows, the pixels outside outlier regions are then guaranteed to be *normal* pixels. Using only these *normal* pixels for estimation, we avoid the disturbance of outliers and therefore improve the precision of estimation results in the large window.

In practice, the eigenvalues may deviate from their standard values due to noise or derivative approximation error. Thus, instead of checking if $\sigma_3 = 0$, we set a threshold λ_{31} for multiple motion detection. If $\sigma_3 > \lambda_{31}\sigma_1$, we conclude that there are multiple motions. We may also check the aperture problem by defining another threshold λ_{21} between σ_2 and σ_1. Here we set $\lambda_{31} = \lambda_{21} = 0.2$.

It should be noticed that we also abandon some *normal* pixels by marking outliers with small windows. Therefore, we prefer to reduce the size of the small window so that this loss is as small as possible. On the other hand, in order to provide robust eigenvalue analysis we must have an adequate number of pixels in the small window. Taking into account that the occlusion boundaries are local in every image frame and that the motions are assumed to be piecewise-smooth, we solve this conflict by limiting the spatial size of the small window, but extending its temporal size to include pixels from other frames as well (e.g., from frames $(t_0 - 1)$ and $(t_0 + 1)$, where t_0 denotes the current frame). In order to verify that the *normal* pixels remaining are still adequate for estimation, we define a reliability measure which is a ratio between the number of *normal* pixels remaining and the total number of pixels in the large window

$$r_{\mathrm{m}} := \frac{\mathcal{N}_i}{\mathcal{N}_{\mathrm{all}}} \quad (i = 1, 2), \tag{21}$$

where $\mathcal{N}_1$ and $\mathcal{N}_2$ denote the number of remaining pixels of the occluding and occluded signal. If either of these two ratios is below a threshold, we have to enlarge the window to include more pixels for estimation.

3.3. Scene analysis

After obtaining multiple motion parameters in the boundary regions we further need to localize occlusion boundaries in one frame and track their movement. Fleet et al. [10,11] modeled an occlusion boundary explicitly as an edge in a local circular mask with six parameters, i.e., four motion parameters of both occluding and occluded signals, the orientation of this boundary, and the distance between the boundary and the center of the mask. This model is only suitable for a straight-line boundary.

The spectral model of occlusion boundary [23,26] assumes implicitly that the boundary is an edge (Eq. (3)). If the boundary has other contours, the term $U(\mathbf{x})$ in Eq. (3) has to be changed. Consequently, the spectrum of $U(\mathbf{x})$ changes in Eq. (6) as well. Since the distribution of boundary deformations has not been studied yet, we cannot propose an explicit model in the spectral domain to describe all possible boundaries.

Instead of using an explicit boundary model to localize motion boundaries, we apply the shift-and-subtract technique based on the spatial coherence of the image sequence [7,6]. Assume we have three successive frames I_{t-1}, I_t, and I_{t+1}. We first shift the frame I_{t-1} with two estimated speeds $\mathbf{v}_1$ and $\mathbf{v}_2$ to form the shifted frames $I_{t-1}(\mathbf{x}+\mathbf{v}_1)$ and $I_{t-1}(\mathbf{x}+\mathbf{v}_2)$. Then we calculate two difference images $\Delta I_{t,1}$ and $\Delta I_{t,2}$

$$\begin{aligned} \Delta I_{t,1}(\mathbf{x}) &= I_t(\mathbf{x}) - I_{t-1}(\mathbf{x}+\mathbf{v}_1), \\ \Delta I_{t,2}(\mathbf{x}) &= I_t(\mathbf{x}) - I_{t-1}(\mathbf{x}+\mathbf{v}_2). \end{aligned} \tag{22}$$

If the speeds are properly estimated, we will find one region with zero intensity in each one of $\Delta I_{t,1}$ and $\Delta I_{t,2}$ in case of occlusion. These two regions are complementary in coordinates (Fig. 3). Their intersection indicates the location of boundaries B_t. Thus, we extract the boundary information in a simple way without using an explicit model.

By repeating the same process on frames I_t and I_{t+1}, we obtain the shifted boundaries B_{t+1} and therefore track the movement of occlusion boundaries. Since the occlusion boundaries move consistently with the occluding signal, we solve the foreground/background ambiguity [11] as well.

The shift-and-subtract technique distinguishes occlusion from transparency, as there is no zero region in either $\Delta I_{t,1}$ or $\Delta I_{t,2}$ in case of transparency. Furthermore, we can use this technique to decompose transparency scenes into their multi-layer representations [9,40] (Fig. 9).

4. Experiment

4.1. Synthetic analysis

First let us confirm the comparisons between the spatial and the spectral model. In Fig. 4, we display the orientation signatures of both occlusion and transparency sequence shown in Fig. 1. These orientation signatures are obtained by projecting the spherical representation of the derivatives or the energy spectrum of the image sequences on to the spherical angular space (see [34] for detail). Since both occlusion and transparency sequence have the same motion parameters, we expect that their orientation signatures have the same curves. A comparison between two rows shows that the spectral model can treat both occlusion and transparency, while the spatial

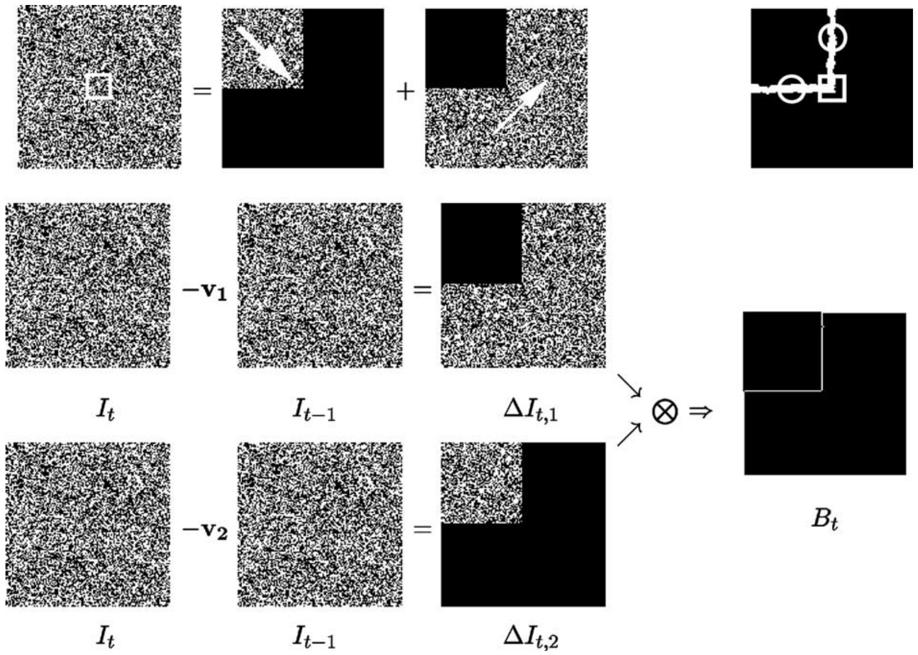
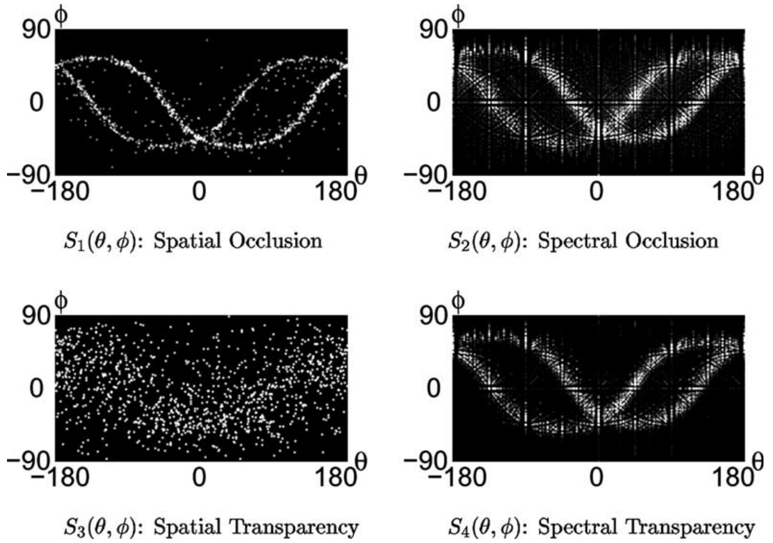


Fig. 3. *Row 1, Left:* One frame from a occlusion sequence. It is composed of one occluding signal moving right-down and one occluded signal moving right-up. *Row 1, Right:* Marked occlusion regions after eigenvalue analysis. While the explicit model [10] can describe the straight-line boundary parts marked with circles, it cannot describe the boundary corner marked with the square window. *Row 2:* Shift-and-subtract with the occluding speed. *Row 3:* Shift-and-subtract with the occluded speed. *Between Rows 2 and 3, Right:* The localized occlusion boundaries. Here we do not consider the border problem in the subtraction step.

model treats only occlusion. In the spectral signatures, we observe distortions outside two main curves in $S_2(\theta, \phi)$, while in $S_4(\theta, \phi)$ these distortions disappear. Besides, a comparison between $S_2(\theta, \phi)$ and $S_1(\theta, \phi)$ confirms that the spectral model has coarser resolution than the spatial model since the spectrum is blurred by LFT. We also apply the EM algorithm both on the spatial orientation signature and on the spectral orientation signature to make a quantitative comparison. The results in Table 1 indicate that the spatial EM algorithm provides more accurate results than the spectral EM algorithm and needs less iterations. Taking into account that we have to use a large window (here the window size is 32 × 32 × 32) to obtain the orientation signature in the spectral domain and that the constant motion assumption is easily violated in such a large window, we prefer to use the spatial model for occlusion analysis.

Next let us see the precision improvement after eliminating outliers in occlusion estimation. In Fig. 5, we show the result of outlier detection in the occlusion sequence (Fig. 1). We further show the orientation signatures before and after eliminating outliers. After eliminating outliers the curves in the (θ, ϕ) space are clearer. Consequently, the estimation results using the EM algorithm are better (Tables 1



$S_1(\theta, \phi)$: Spatial Occlusion

$S_2(\theta, \phi)$: Spectral Occlusion

$S_3(\theta, \phi)$: Spatial Transparency

$S_4(\theta, \phi)$: Spectral Transparency

Fig. 4. Orientation signatures of occlusion and transparency sequences in Fig. 1. *Top left:* The orientation signature of the occlusion sequence in the spatial domain. We use a 33 × 33 window in the derivative space to obtain this signature. *Top right:* The signature of the occlusion sequence in the spectral domain. We use a 32 × 32 × 32 window to obtain this signature. *Bottom left:* The signature of the transparency sequence in the spatial domain. The distribution of points is nearly random. *Bottom right:* The signature of the transparency sequence in the spectral domain.

Table 1
Occlusion estimation in spatial and spectral domain

Model	Eliminating outlier	Iteration	Occluding	Occluded
Spatial	Before	1	$(0.980, 0.997)$	$(0.963, -0.974)$
	After	1	$(0.994, 0.997)$	$(0.978, -0.988)$
Spectral	Not available	2	$(0.966, 1.002)$	$(1.007, -1.026)$

Occlusion estimation with initial values $(u_{10}, v_{10}) = (0.9, 1.1)$ and $(u_{20}, v_{20}) = (0.9, -1.1)$ and tolerance parameter $\sigma_r = 0, 1$. These initial values are calculated using the extreme points in the orientation signatures in Fig. 4.

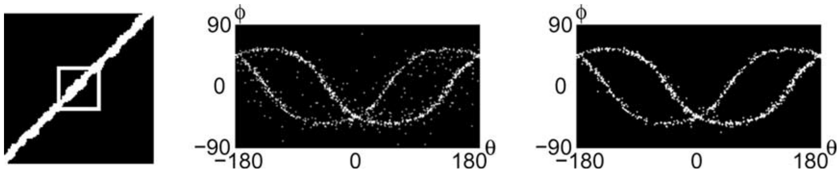


Fig. 5. *Left:* Marked outliers in the random dot occlusion sequence in Fig. 1 after eigenvalue analysis using a 5 × 5 × 3 window. The white box here shows the estimation window across the occlusion boundary. *Middle:* Orientation signature of 3D data in the (I_x, I_y, I_t) space before eliminating outliers. *Right:* Orientation signature after eliminating outliers. The two curves are clearly distinguishable. See Tables 1 and 2 for estimation results.

Table 2
The effect of eliminating outliers in occlusion estimation

Window size	Eliminating outliers	Occluding speed	Occluded speed
33×33	Before	$(0.986, 0.999)$	$(0.986, -0.988)$
	After	$(0.998, 0.999)$	$(0.990, -0.995)$
17×17	Before	$(0.880, 0.971)$	$(0.859, -0.869)$
	After	$(0.988, 1.013)$	$(0.993, -0.998)$

Estimation results before and after eliminating outliers with different window sizes. For comparison we apply the EM algorithm with the same parameters and initial values before and after eliminating outliers: $\sigma_r = 0, 1$, $(u_{10}, v_{10}) = (0.8, 0.3)$, and $(u_{20}, v_{20}) = (1.2, -0.1)$.

and 2). To analyze the effect of window size in the estimation, we reduce the estimation window from 33×33 to 17×17. In the 17×17 window, the number of outliers is more susceptible to be comparable to the number of *normal* pixels. As a result, the disturbance of outliers increases strongly. In contrast, if we eliminate outliers before estimation, we still can obtain reasonable results.

4.2. Real occlusion analysis

Fig. 6 shows the example of the well-known flower garden occlusion sequence in which a left moving trunk covers the left moving flower bed and houses. We first estimate single motion using the least square method. After obtaining the partial derivatives using a $5 \times 5 \times 5$ kernel, we build three derivative column vectors I_x, I_y, and I_t from a local 21×21 window. Then, we form a matrix $A_{441\times 2} = [I_x \; I_y]$ and estimate the optical flow using $(u, v) = -[A^{\star} I_t]^{\mathrm{T}}$, where $A^{\star}$ denotes the pseudo inverse of A. At the occlusion boundaries the results are not correct, as shown in row 2 of Fig. 6. We apply the framework introduced in Section 3 to estimate multiple motions. During the estimation of multiple motions, the window size is increased to 31×31 to contain more pixels. Each time when we obtain two estimated speeds in a local window, we assign the larger one to one layer and assign the smaller one to the other layer, as displayed in row 2. In row 3, we apply the shift-and-subtract technique. Before and after shifting, there is no difference inside the regions with the aperture problem (such as trunk and the sky). As a result, we only observe the boundaries of the trunk in the difference image $\Delta I_{t,1}$. In fact, the difference images $\Delta I_{t,1}$ and $\Delta I_{t,2}$ can be viewed as the result of occlusion segmentation. We further localize occlusion boundaries from $\Delta I_{t,1}$ and $\Delta I_{t,2}$ (row 3). The boundaries are not well connected since the nonzero regions in $\Delta I_{t,1}$ are discrete due to the aperture problem.

In this example, the Matlab source code runs on a Pentium III PC (933 MHz CPU and 512 MB memory) under LINUX environment. The least square method needs about 34 minutes to estimate a single motion in an entire image, while the EM algorithm needs about 35 minutes to estimate multiple motions. Considering that the occlusion region is only a very small fraction of the entire image (less than 5% in area), the EM algorithm needs much more computation than the least square method. The cost of shift-and-subtract technique is negligible compared to the cost of EM algorithm.

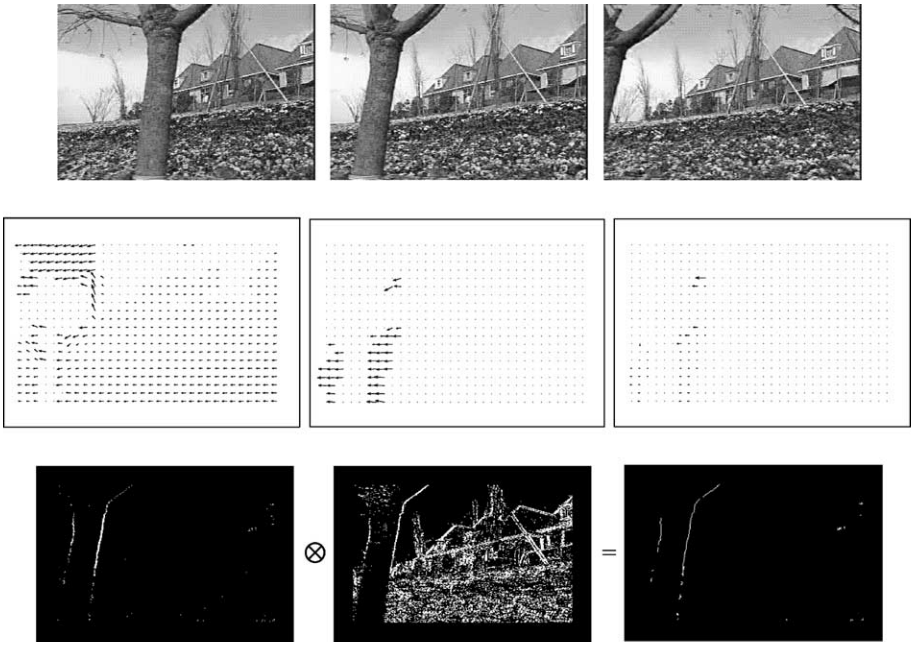


Fig. 6. *Row 1:* The 17th, 32th, and 48th frames of the flower garden sequence. Each frame has 240 × 352 pixels. Here we consider the 32th frame as the central frame. *Row 2, Left:* Single motion estimation results using the least square method (window size: 21 × 21). At motion boundaries the results are not correct. *Row 2, Middle and right:* Optical flow applying the spatial EM algorithm (window size: 31 × 31). *Row 3:* Detection and localization of motion boundaries. *Row 3, Left:* Difference image $\Delta I_{t,1}$ (cf. Eq. (22)). *Row 3, Middle:* Difference image $\Delta I_{t,2}$. *Row 3, Right:* Detected motion boundaries.

Characterizing a flower garden image pixel by pixel with a 21 × 21 window would need about 8.6 hours using spatial conic filtering. To reduce computation time, we characterize motions sparsely by changing the shift-interval of the 21 × 21 window from 1 to 5 pixels. The motion characterization time is reduced by a factor of 25. As a result, we need only about 21 minutes to characterize the motions in a flower garden image. This sparse characterization is based on the assumption that all pixels in the central 5 × 5 block of the 21 × 21 window are in the same motion class (single motion vs. multiple motions) simultaneously.

The computation time will be further reduced when we implement the algorithm in C source code. But we cannot yet give an exact answer that how much time we can save using C code. In addition, a parallel implementation of the characterization step and even of the estimation step is also considerable to accelerate the computation since the conic filtering and the motion estimation are highly local operations.

Figs. 7 and 8 display another occlusion example in which a right moving box is covering a left moving picture. The image is rich in texture so that we do not face the aperture problem. We first estimate motions with the single motion model. The results at the occlusion boundaries are incorrect. Then we apply our framework

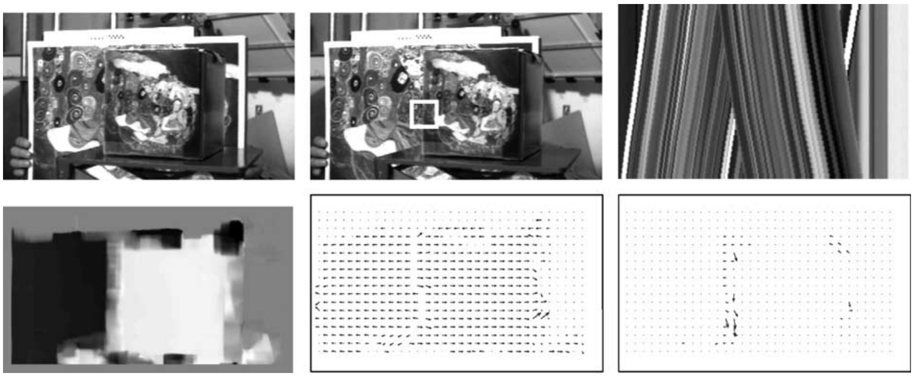


Fig. 7. *Top left and middle:* The first and 16th frame of the occlusion sequence. Each frame has 200 × 350 pixels. The white window in the 16th frame is centered at (122,137). *Top right:* The epipolar slice of the sequence along row 122. The first frame is at the top of the slice. The occlusion is characterized as two overlapping structures. (These three images are reprinted from *Yu et al. "Oriented Structure of the Occlusion Distortion: Is It Reliable?" IEEE Trans. Pattern Anal. Mach. Intell. 24(9) (2002) 1286–1290* (© 2002 IEEE) with the permission of IEEE). *Bottom left:* The result of single motion estimation using the least square method with window size 21 × 21. Since the vertical speed components are almost zero in this sequence, we show only horizontal speed components, using black color for negative speed (moving to the left) and white color for positive speed (moving to the right). *Bottom middle and right:* Spatial EM estimation results in the 16th frame.

and show estimation results in row 2 of Fig. 7. To evaluate the effect of eliminating *outliers* we apply the EM algorithm vertically along the occlusion boundary before and after eliminating *outliers*. It is a little bit difficult to compare the precision of estimation results since the ground truth is unknown. But the box as well as the picture have purely translational motions and there is no depth difference in the box region or in the picture region. Thus, there is almost no speed difference among pixels on each side of the boundary and we can use the estimation results within a large (31 × 31) window, where there are much more *normal* pixels than *outliers*, as ground truth. In the results with a small (15 × 15) window we observe the improvement after eliminating *outliers* clearly. In the window centered at (160,137) the results are not reasonable because there are only *four* pixels of the occluded signal remaining after eliminating *outliers*. This example demonstrates vividly the necessity of introducing reliability measure (Eq. (21)). By using the shift-and-subtract technique, we further localize the occluding boundary, which is displayed as intersection of zero regions in $\Delta I_{t,1}$ and $\Delta I_{t,2}$, as shown in row 3 of Fig. 8. This shift-and-subtract technique works also for boundaries with complex contours like the corners of the right moving box.

In this example, the BCCE-based least square method needs about 27 minutes to estimate single motion. It takes 17 minutes to characterize motions using conic filtering in the spatial domain. In the multiple motion region (the narrow band around the box), the spatial EM algorithm needs only 4 minutes to finish the estimation. The fast convergence of the EM algorithm is due to the fact that there is almost no speed difference on either side of the occlusion boundary.

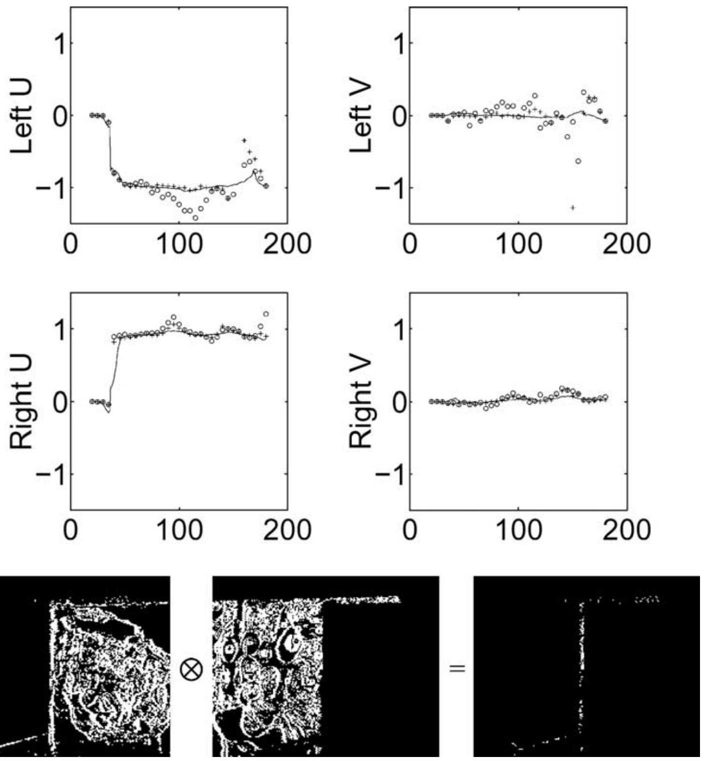


Fig. 8. *Rows 1 and 2:* Spatial EM results along column 137 using a 15 × 15 window. We use the results with a 31 × 31 window as the ground truth and plot them with solid lines. We plot the results before eliminating *outliers* with circles and the results after eliminating *outliers* with crosses. For comparison we plot different speed components separately. For clarity we sample the results with an interval of 5 pixels along column 137. *Row 3:* The result after shift-and-subtract. For clarity we enlarge the occlusion boundary region.

4.3. Real transparency analysis

In Fig. 9 we show a real transparency sequence in order to compare the spatial and spectral multiple motion models. It contains a right moving portrait and a mirrored left moving "muesli" package. We display the results of the BCCE-based least square method using 21 × 21 window in row 1. After the motion characterization step, the spectral EM algorithm with window size 32 × 32 × 32 is chosen for transparency estimation. For comparison, we apply the spatial EM algorithm (with window size 31 × 31) as well. The spatial EM algorithm is not able to estimate transparent motions correctly, while the spectral EM approach works well. The optical flow in the spectral EM approach is sparse due to the fact that in some regions of the package we do not have adequate texture information and the corresponding eiqenvalue-ratio σ_2/σ_1 is below the threshold $\lambda_{21} = 0.2$ (cf. Section 3.2). For a robust performance we ignore these regions in estimation.

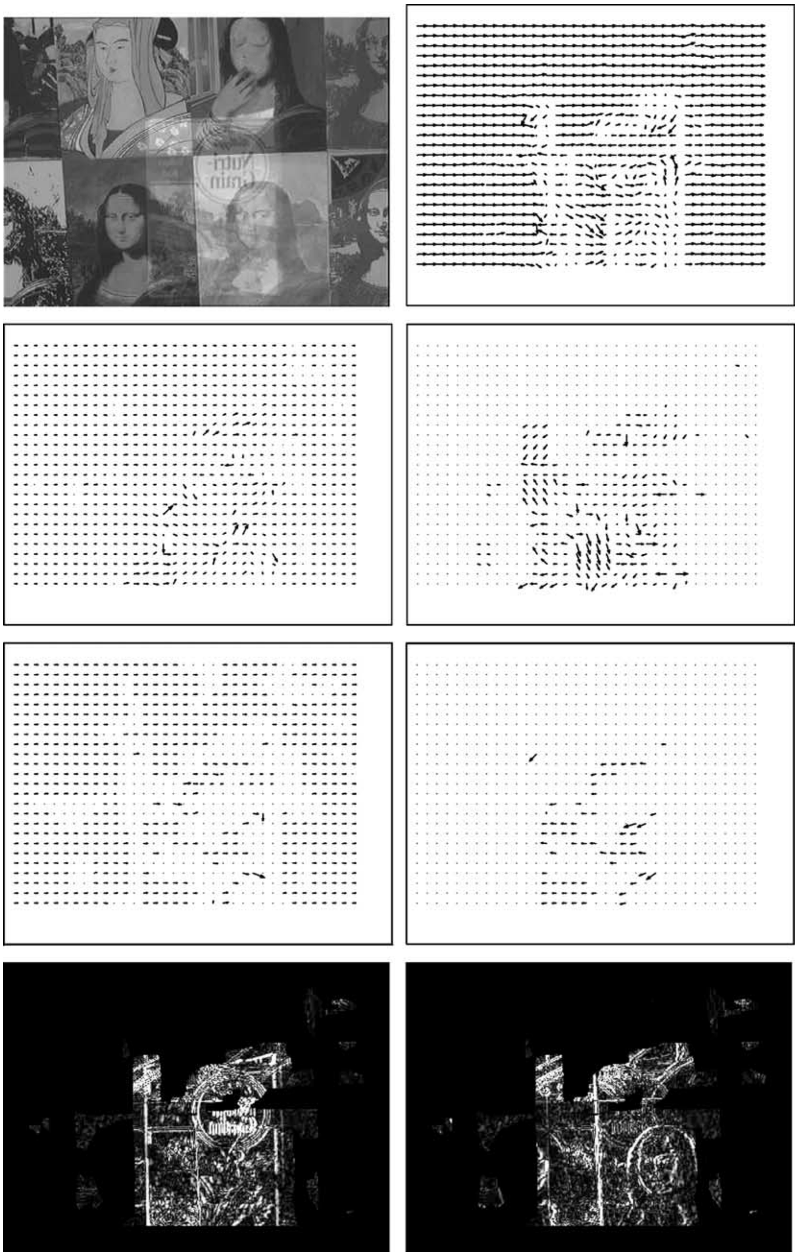


Fig. 9. Comparison of spatial- and spectral-EM algorithms on real transparency sequence. *Row 1, Left:* The 16th frame of the image sequence with 288 × 384 pixels. *Row 1, Right:* Estimation results using the single motion model in the 16th frame. *Row 2:* Optical flow of the spatial EM approach. The estimation results are not correct in the transparent region. *Row 3:* Optical flow of the spectral EM approach. *Bottom:* Decomposition of the transparency scene into two layers using the spectral EM results and the shift-and-subtract technique. *Bottom left:* Difference image $\Delta I_{t,1}$. *Bottom right:* Difference image $\Delta I_{t,2}$.

After obtaining the motion parameters, we further decompose the transparency scene into multi-layers with the shift-and-subtract technique. The results are shown in difference images $\Delta I_{t,1}$ and $\Delta I_{t,2}$. The layers are not in line with the package shape since some regions of the package suffer under the aperture problem.

In this transparency example, the single motion estimation step needs about 45 minutes, while the multiple motion estimation in the spatial domain needs one hour. The spectral motion estimation needs only 13 minutes since we ignore many regions by using λ_{21}. The motion characterization step in the spatial–temporal derivative space with a 21×21 window needs about 28 minutes if we shift the window in a step of 5 pixels. Characterizing motions in the spectral domain with a $32 \times 32 \times 32$ window, however, needs much longer (135 minutes) since we have to extract the local energy spectrum in each $32 \times 32 \times 32$ block.

5. Conclusion

In this paper, we compare the estimation and analysis of multiple motions in the spatial domain and in the spectral domain. While the spectral motion model describes both occlusion and transparency in a uniform manner and it is rigorously correct, the resolution limitation of frequency-based techniques makes the spectral model less attractive. Thus, we prefer to stay in the spatial domain for occlusion analysis.

In order to localize occlusion boundaries and to track their movement, we utilized the spatial coherence inside the frame and applied the shift-and-subtract technique. While we did not use an explicit local model of the boundary region, we still obtained the desired information about the occlusion boundaries. Furthermore, multiple motions can be segmented very efficiently by combining estimation techniques and spatial coherence [6]: The region with the same motion parameters can be figured out by calculating the difference between two frames with estimated speeds.

The spatial coherence information is also a key cue to distinguish occlusion and transparency in the spatial domain. Actually, it is not difficult to distinguish occlusion from transparency in the frequency domain. For example, we can look at a set of estimation results by shifting the observing window and observe their variation. Since occlusion is more local than transparency, the number of motions changes from two to one after the observing window has crossed occlusion boundaries, while in case of transparency the number of motions remains the same. We may also observe the relative ratio between data points outside the motion planes and those on the motion planes [26]. This ratio is much larger in case of occlusion than in case of transparency since all energy of the transparency lies on the dominant planes. However, in the frequency domain we cannot localize motion boundaries due to the well known uncertainty principle: The spectrum of the observing window provides us with no localization information inside the window. Therefore, we must go back to the spatial domain to detect and localize motion boundaries, where the coherence information plays a very important role in image segmentation and scene analysis. The shift-and-subtract technique and the recently introduced *normalized cut* approach [41,42] emphasize this point vividly.

Acknowledgments

We appreciate the anonymous reviewers' detailed comments which help us to improve and clarify this paper. We thank H. Farid, G. Birkelbach, M. Michaelis, S. Beauchemin, and M. Felsberg for their helpful suggestions and discussions. The financial support provided to the first author by German Academic Exchange Service (DAAD) and by NIH Grant R01 HL 44803-06 to J.S. Duncan is gratefully acknowledged. The second author is supported by Deutsche Forschungsgemeinschaft (DFG) Grant 320/1-3. The third author is grateful for the following support: NSF-IIS-0083209, NSF-EIA-0218691, NSF-IIS-0121293, NSF-EIA-9703220, a DARPA/ITO/NGI subcontract to UNC, and a Penn Research Foundation grant.

Appendix A. The relation between (θ_n, ϕ_n) and (θ_m, ϕ_m)

In Fig. 10, we represent all possible unit vectors on a 3D plane with a circle. The normal vector **n** is perpendicular to all vectors on this plane, including the vector $\mathbf{m}_1$ (pointing to the extreme coordinates (θ_m, ϕ_m)) and $\mathbf{m}_2$ (pointing to the point $(\theta_m - 90°, 0)$). Obviously, $\mathbf{m}_2$ is also perpendicular to $\mathbf{m}_1$. Since $\mathbf{m}_2$ lies in the horizontal XY plane, the dotted plane containing **n** and $\mathbf{m}_1$ is then perpendicular to the XY plane. In this vertical plane, we have

$$\phi_n + 90° + \phi_m = 180°. \tag{A.1}$$

This vertical plane always divides the circle equally as it passes through the origin. Taking into account that angles in the θ direction are periodic we have

$$|\theta_n - \theta_m| = 180°.$$

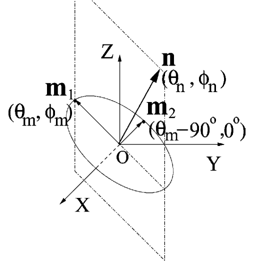


Fig. 10. The relation between (θ_n, ϕ_n) and (θ_m, ϕ_m). The circle contains all possible unit vectors on a 3D plane. The dotted plane containing the normal vectors **n** and $\mathbf{m}_1$ is a vertical plane perpendicular to the XY plane.

Without affecting the calculation of the velocity we simply take

$$\theta_n - \theta_m = 180^\circ. \tag{A.2}$$

Then we obtain Eq. (16).

References

[1] B.K.P. Horn, Robot Vision, MIT Press, 1986.

[2] H.H. Nagel, W. Enkelmann, An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences, IEEE Trans. Pattern Anal. Mach. Intell. 8 (1986) 565–593.

[3] M. Proesmans, L.J. VanGool, E. Pauwels, A. Oosterlinck, Determination of optical flow and its discontinuities using non-linear diffusion, in: J.O. Eklundh (Ed.), Proceedings of the Third European Conference on Computer Vision, Stockholm, Sweden, May 2–6, Lecture Notes in Computer Science, vol. 801, Springer, Berlin, 1994, pp. 295–304.

[4] J. Weickert, C. Schnörr, Räumlich-zeitliche Berechnung des optischen Flusses mit nicht-linearen flußabhängigen Glattheitstermen, in: DAGM Symposium Mustererkennung, Bonn, Germany, September 15–17, 1999, pp. 317–324.

[5] S.F. Wu, J. Kittler, A gradient-based method for general motion estimation and segmentation, J. Vis. Commun. Image Represent. 4 (1993) 25–38.

[6] Y. Weiss, E.H. Adelson, A unified mixture framework for motion segmentation: incorporating spatial coherence and estimating the number of models, in: IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, June 18–20, 1996, pp. 321–326.

[7] J.R. Bergen, P.J. Burt, R. Hingorani, S. Peleg, A three-frame algorithm for estimating two-component image motion, IEEE Trans. Pattern Anal. Mach. Intell. 14 (9) (1992) 886–895.

[8] M. Irani, B. Rousso, S. Peleg, Computing occluding and transparent motions, Int. J. Comput. Vis. 12 (1994) 5–16.

[9] J.Y.A. Wang, E.H. Adelson, Layered representation for motion analysis, in: IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, June 15–17, 1993, pp. 361–366.

[10] D.J. Fleet, M.J. Black, Y. Yacoob, A.D. Jepson, Design and use of linear models for image motion analysis, Int. J. Comput. Vis. 36 (3) (2000) 171–193.

[11] M.J. Black, D.J. Fleet, Probabilistic detection and tracking of motion boundaries, Int. J. Comput. Vis. 38 (3) (2000) 231–245.

[12] M.J. Black, P. Anandan, The robust estimation of multiple motions: parametric and piecewise-smooth flow fields, Comput. Vis. Image Understand. 63 (1) (1996) 75–104.

[13] A. Jepson, M.J. Black, Mixture models for optical flow computation, in: IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, June 15–17, 1993, pp. 760–761.

[14] P. Bouthemy, A maximum likelihood framework for determining moving edges, IEEE Trans. Pattern Anal. Mach. Intell. 11 (1989) 499–511.

[15] B.G. Schunck, Image flow segmentation and estimation by constraint line clustering, IEEE Trans. Pattern Anal. Mach. Intell. 11 (10) (1989) 1010–1027.

[16] M. Bober, J. Kittler, Estimation of complex multimodal motion: an approach based on robust statistics and Hough transform, Image Vis. Comput. 12 (10) (1994) 661–668.

[17] E.H. Adelson, J.R. Bergen, Spatiotemporal energy models for the perception of motion, J. Opt. Soc. Am. 1 (2) (1985) 284–299.

[18] J. Bigün, G.H. Granlund, Optimal orientation detection of linear symmetry, in: Proceedings of the International Conference on Computer Vision, London, UK, June 8–11, 1987, pp. 433–438.

[19] J. Bigün, G.H. Granlund, J. Wiklund, Multidimensional orientation estimation with application to texture analysis and optical flow, IEEE Trans. Pattern Anal. Mach. Intell. 13 (8) (1991) 775–790.

[20] M. Shizawa, K. Mase, A unified computational theory for motion transparency and motion boundaries based on eigenenergy analysis, in: IEEE Conference on Computer Vision and Pattern Recognition, Maui, Hawaii, June 3–6, 1991, pp. 289–295.

[21] D.J. Fleet, Measurement of Image Velocity, Kluwer Academic Publishers, 1992.

[22] D.J. Fleet, K. Langley, Computational analysis of non-Fourier motion, Vis. Res. 34 (1994) 3057–3079.

[23] S.S. Beauchemin, J.L. Barron, The frequency structure of 1D occluding image signals, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 200–206.

[24] H. Knutsson, Filtering and reconstruction in image processing, PhD thesis, Department of Electrical Engineering, Linkoeping University, Linkoeping, Sweden, 1982, Dissertation No. 88.

[25] B. Jähne, Spatio-Temporal Image Processing, Springer, Berlin, 1993.

[26] W. Yu, K. Daniilidis, S. Beauchemin, G. Sommer, Detection and characterization of multiple motion points, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. I, Fort Collins, CO, June 23–25, 1999, pp. 171–177.

[27] D.J. Heeger, Optical flow using spatiotemporal filters, Int. J. Comput. Vis. 1 (4) (1987) 279–302.

[28] Y. Xiong, S.A. Shafer, Moment and hypergeometric filters for high precision computation of focus, stereo and optical flow, Int. J. Comput. Vis. 24 (l) (1997) 25–59.

[29] A.F. Nikiforov, V.B. Uvarov, Special Functions of Mathematical Physics, Birkhaeuser, 1988.

[30] N.M. Grzywacz, A.L. Yuille, A model for the estimate of local image velocity by cells in the visual cortex, Proc. Roy. Soc. Lond. B 239 (1990) 129–161.

[31] T.S. Lee, Image representation using 2D Gabor wavelets, IEEE Trans. Pattern Anal. Mach. Intell. 18 (10) (1996) 959–971.

[32] R.P. Würtz, Object recognition robust under translations, deformations, and changes in background, IEEE Trans. Pattern Anal. Mach. Intell. 19 (7) (1997) 769–774.

[33] S.S. Beauchemin, J.L. Barren, On the Fourier properties of discontinuous visual motion, J. Math. Imag. Vis. 13 (3) (2000) 155–172.

[34] W. Yu, G. Sommer, K. Daniilidis, 3D-orientation signatures with conic kernel filtering for multiple motion analysis, Image Vis. Comput. (2003), in press.

[35] M.S. Langer, R. Mann, Dimensional analysis of image motion, in: Proceedings of the International Conference on Computer Vision, vol. I, Vancouver, Canada, July 7–14, 2001, pp. 155–162.

[36] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. R. Statist. Soc. B 39 (1977) 1–38.

[37] S. Ayer, H.S. Sawhney, Layered representation of motion video using robust maximum-likelihood estimation of mixture models and MDL encoding, in: Proceedings of the International Conference on Computer Vision, Boston, MA, June 20–23, 1995, pp. 777–784.

[38] H. Gu, Y. Shirai, M. Asada, MDL-based segmentation and motion modeling in a long image sequence of scene with multiple independently moving objects, IEEE Trans. Pattern Anal. Mach. Intell. 18 (l) (1996) 58–64.

[39] Y. Weiss, Smoothness in layers: motion segmentation using nonparametric mixture estimation, in: IEEE Conference on Computer Vision and Pattern Recognition, Puerto Rico, June 17–19, 1997, pp. 520–526.

[40] R. Szeliski, S. Avidan, P. Anandan, Layer extraction from multiple images containing reflections and transparency, in: IEEE Conference on Computer Vision and Pattern Recognition, vol. I, Hilton Head Island, SC, June 13–15, 2000, pp. 246–253.

[41] J. Shi, J. Malik, Normalized cuts and image segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition, Puerto Rico, June 17–19, 1997, pp. 731–737.

[42] J. Shi, J. Malik, Motion segmentation using normalized cuts, in: Proceedings of the International Conference on Computer Vision, Bombay, India, January 4–7, 1998, pp. 1154–1160.