Multiple-Motion-Estimation by Block-matching using MRF

Ingo Stuke[1], Til Aach[1], Erhardt Barth[2], and Cicero Mota[2]

[1]Institute for Signal Processing
[2]Institute for Neuro- and Bioinformatics
University of Lübeck, Ratzeburger Allee 160, 23538 Lübeck, Germany

Abstract

This paper deals with the problem of estimating multiple motions at points where these motions are overlaid. We present a new approach that is based on block-matching and can deal with both transparent motions and occlusions. We derive a block-matching constraint for an arbitrary number of moving layers. We use this constraint to design a hierarchical algorithm that can distinguish between the occurrence of single, transparent, and occluded motions and can thus select the appropriate local motion model. The algorithm adapts to the amount of noise in the image sequence by use of a statistical confidence test. Robustness is further increased with a regularization scheme based on Markov Random Fields. Performance is demonstrated on image sequences synthesized from natural textures with high levels of additive dynamic noise and on real video sequences.

Keywords: Block-matching, multiple motions, transparent motions, occlusion, Markov Random Fields.

1. Introduction

Motion estimation is essential in a variety of image processing and computer vision tasks, like video coding, tracking, directional filtering and denoising, scene analysis, etc. Standard motion models, however, fail in case of transparent and occluded motions. In case of transparent motions, two or more motion vectors are observable at the same image location and time. As with a single motion, the estimation of multiple motions implies a one-to-many correspondence and is thus an ill-posed problem [8]. In consequence, algorithms for motion estimation have to incorporate some form of local regularization and to estimate the local motion parameters based on a local neighborhood. For a single motion, the algorithms can be classified in three main classes: differential, transform-based, and block-matching based methods.

A differential algorithm for two transparent motions was first proposed by Shizawa and Mase [17] and was later generalized for the case of N motions in [15] where an analytic solution based on so-called mixed motion parameters was presented. The introduction of the mixed motion parameters allows to linearize the transparent motion equation of Shizawa and Mase. They encode the motions unambiguously, but not explicitly. The motion vectors are obtained by

solving for the roots of a complex polynomial, whose coefficients can be expressed in terms of the mixed-motion parameters. A similar approach was used by Langley [14] for the case of two-motions and in the context of vision modeling. A phase-based solution for the estimation of two transparent overlaid motions was proposed by Vernon [23]. This method has been generalized for an arbitrary number of N motions in [21]. This generalization led to solutions for extracting the N motions at a given location and for separating the moving image layers. Szeliski et al. [22] proposed a layer extraction algorithm for transparency and reflections together with a multiple-motions technique that recovers the layers and their motions from the input images in the spatial domain.

Pingault et al. [16] used Vernon's solution in the spatial domain to derive a flow constraint for two transparent motions based on a Taylor-series expansion. Additionally, they regularized the motion vector fields using a B-Spline approach.

It is well known that the Fourier transform spectrum of an image undergoing rigid translations lies in a plane in the spatio-temporal frequency domain [25]. In case of transparent motions, each moving layer lies in such a plane and several filter-based methods that parameterize these planes have been developed [26, 18].

Bergen et al. [7] propose an area-regression approach for estimating two motions from only three frames. The approach uses an iterative algorithm to estimate one motion, performs a nulling operation to remove the intensity pattern giving rise to this first motion, an then solves for the second motion. Other approaches based on nulling filters and velocity-tuned mechanisms have been proposed in [10, 11].

Wang and Adelson [24] model an image region by a set of overlapping opaque layers. They initially compute single motions by using a least-square approach within local image patches. They then use K-means clustering to group motion estimates into regions of consistent affine motions, which then define the layers. This is different from our way of dealing with occluding motions since we assume that a transparent-layer model is valid in the vicinity of the occluding boundary, but not at the boundary.

Although both differential and transform-based methods are fast and perform well for small displacements, block-matching is known to perform better for large displacements and with higher levels of noise. It is thus a widely used method in various technical applications. To our knowledge,

the first block-matching algorithm for multiple-motion estimation was proposed in [20] based on some related results obtained in [7, 16, 23]. In this paper we first extend the algorithm in [20] within a stochastic framework such as to include a confidence test. A second extension proposed here is based on Markov random fields and greatly enhances robustness. The algorithm is derived from the phase-based solution for the Fourier-domain equations for transparent motions [23, 21]. The distortion caused by occluding regions is also analyzed and we show how to apply our algorithm to estimate motions at occlusions.

2. The block-matching constraint

The block-matching constraint will be derived from the phased-based method for multiple motion estimation [23, 21]. The image sequence is therefore modeled as an additive superposition of N independent moving layers. This model is transformed to the Fourier domain and the motion layers are successively eliminated by analytical methods. The remaining equations describe a non-linear coupling between the desired motion vectors and the observed image sequence. These equations are then transformed back to the spatial domain, where they define a multiple-motions block-matching constraint. This constraint actually describes how a particular image in the sequence results from N previous images that are individually warped according to the motion parameters and then superimposed.

2.1 The block-matching equation for N motions

In the spatial domain, we model N transparent motions as

$$f_k(\boldsymbol{x}) = f(\boldsymbol{x}, k) = g_1(\boldsymbol{x} - k\boldsymbol{v}_1) + g_2(\boldsymbol{x} - k\boldsymbol{v}_2) + \cdots \\ + g_N(\boldsymbol{x} - k\boldsymbol{v}_N), \quad k = 0, 1, \ldots \tag{1}$$

The above system of equations involves the observed images f_k for each time step k and spatial position $\boldsymbol{x}$, the unknown layers g_n and the motion vectors $\boldsymbol{v}_n$ for $n = 1, \ldots, N$, that we wish to determine, see [17].

In the Fourier domain, Equation (1) becomes

$$F_k(\omega) = \phi_1^k G_1(\omega) + \phi_2^k G_2(\omega) + \cdots + \phi_N^k G_N(\omega), \quad (2)$$

where $\phi_n = e^{-j\omega \cdot \boldsymbol{v}_n}$, $n = 1, \ldots, N$ are the phase shifts and $\omega = (\omega_x, \omega_y)$ are the frequency variables. Uppercase letters denote the Fourier transforms of the corresponding functions denoted with lower-case letters, e.g., F_k is the Fourier transform of f_k.

We simplify notation by setting $\Phi_k = (\phi_1^k, \ldots, \phi_N^k)^T$ and $\boldsymbol{G} = (G_1, \ldots, G_N)^T$ and obtain the following expression for the above system of equations:

$$F_k = \Phi_k \cdot \boldsymbol{G}. \tag{3}$$

The goal now is the elimination of the unknown vector $\boldsymbol{G}$ that contains the Fourier-transforms of the motion layers. The remaining equation then relates only to the observable

Fourier transform of the single images and the phase shifts, i.e., $F_0, \ldots, F_N$ and $\phi_1, \ldots, \phi_N$. Note that we need to use at least N past frames in which the motion vectors $\boldsymbol{v}_n$ are assumed to be constant. The polynomial

$$p(z) = (z - \phi_1) \cdots (z - \phi_N) = a_0 z^N + a_1 z^{N-1} + \cdots + a_N \tag{4}$$

with unknown coefficients $a_1, \ldots, a_N$ and $a_0 = 1$ allows for an analytical elimination of the unknown layers g_n. Since the roots of the polynomial are the phase terms in $\Phi_1 = (\phi_1, \ldots, \phi_N)$, we have:

$$a_0 \Phi_N + a_1 \Phi_{N-1} + \cdots + a_N \Phi_0 = \\ (p(\phi_1), \ldots, p(\phi_N)) = \boldsymbol{0}. \tag{5}$$

Therefore by inserting (3) in (5) we obtain

$$a_0 F_N + a_1 F_{N-1} + \cdots + a_N F_0 = \\ (\Phi_N + a_1 \Phi_{N-1} + \cdots + a_N \Phi_0) \cdot \boldsymbol{G} = \boldsymbol{0} \cdot \boldsymbol{G} = 0. \tag{6}$$

The coefficients of $p(z)$ are symmetric polynomials in terms of the roots $\phi_1, \ldots, \phi_N$:

$$\begin{aligned}
a_0 &= 1 \\
a_1 &= -\sum_{i=1}^{N} \phi_i \\
a_2 &= \sum_{i<l} \phi_i \phi_l \\
a_3 &= -\sum_{i<l<k} \phi_i \phi_l \phi_k \\
&\vdots \\
a_N &= (-1)^N \phi_1 \phi_2 \cdots \phi_N.
\end{aligned}$$

By transforming Equation (6) back into the spatial domain we obtain

$$e(f, \boldsymbol{x}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N) = \\ (-1)^N f_0(\boldsymbol{x} - \boldsymbol{v}_1 - \cdots - \boldsymbol{v}_N) + \cdots \\ - \sum_{i<l} f_{N-2}(\boldsymbol{x} - \boldsymbol{v}_i - \boldsymbol{v}_l) \\ + \sum_{i} f_{N-1}(\boldsymbol{x} - \boldsymbol{v}_i) - f_N(\boldsymbol{x}) = 0, \tag{7}$$

because the products of the phase terms lead to concatenated shifts in the spatial domain. Since each a_n is a sum of $\binom{N}{n}$ terms, the central part of Equation (7) has $\sum_{n=0}^{N} \binom{N}{n} = 2^N$ terms.

Equation (7) describes how the N-th image can be constructed form the N previous images by using the motion vectors. Therefore, this equation can be used as the basis for block-matching methods for an arbitrary number of motions that is theoretically unlimited. For a single motion, Equation (7) reduces to the classical block-matching constraint

$$e(f, \boldsymbol{x}, \boldsymbol{v}) = f_0(\boldsymbol{x} - \boldsymbol{v}) - f_1(\boldsymbol{x}) = 0 \tag{8}$$

while for two transparent motions, it becomes

$$e(f, \boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2) = f_0(\boldsymbol{x} - \boldsymbol{v}_1 - \boldsymbol{v}_2) - f_1(\boldsymbol{x} - \boldsymbol{v}_1)$$
$$- f_1(\boldsymbol{x} - \boldsymbol{v}_2) + f_2(\boldsymbol{x}) = 0. \quad (9)$$

2.2 Causality and refinement

The motions estimated according to Equation (7) are assigned to frame f_N. This implies that the estimated motion vectors correspond to that frame, whose argument in the block-matching constraint does not depend on the motion vectors. In other words, the above block-matching constraint uses N previous frames in order to estimate the motions for the image f_N and is thus a causal method. However, at any given time index k we could, in addition, refine older measurements by using the same constraint in a different way. As we shall see, simple substitutions allow to estimate the motions of any other frame of the remaining frames $f_0, \ldots, f_{N-1}$. Of course, this procedure has to be performed in a causal way. In order to estimate the motion vectors for f_0 we use the substitution $\boldsymbol{y} = \boldsymbol{x} - \boldsymbol{v}_1 - \cdots - \boldsymbol{v}_N$ and thus Equation (7) becomes

$$e(f, \boldsymbol{y}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N) =$$
$$(-1)^N f_0(\boldsymbol{y}) +$$
$$(-1)^{N-1} \sum_{i<l} f_1(\boldsymbol{y} + \boldsymbol{v}_i)$$
$$(-1)^{N-2} \sum_i f_2(\boldsymbol{y} + \boldsymbol{v}_i + \boldsymbol{v}_j) + \cdots \quad (10)$$
$$- f_N(\boldsymbol{y} + \boldsymbol{v}_1 + \cdots + \boldsymbol{v}_N) = 0.$$

Additional equations are obtained by using one of the following substitutions:

$$\begin{aligned}
\boldsymbol{y} &= \boldsymbol{x} - \boldsymbol{v}_i, & i &= 1, \ldots, N \\
\boldsymbol{y} &= \boldsymbol{x} - \boldsymbol{v}_i - \boldsymbol{v}_j, & i &< j \\
\boldsymbol{y} &= \boldsymbol{x} - \boldsymbol{v}_i - \boldsymbol{v}_j - \boldsymbol{v}_k, & i &< j < k \\
&\vdots \\
\boldsymbol{y} &= \boldsymbol{x} - \boldsymbol{v}_1 - \cdots - \boldsymbol{v}_N
\end{aligned} \quad (11)$$

Table 1 depicts all possible block-matching constraints that can be obtained for two motions by the above substitutions. Since Equation (7) has 2^N possible shifts we have $2^N - 1$ possible substitutions and thus 2^N different block-matching constraints. Note that Equation (7) defines $\binom{N}{k}$ occurrences of every image frame f_k. Therefore we have $\binom{N}{k}$ different possibilities to estimate the motions for f_k. These additional constraints can be used to improve the final block-matching result. For instance, let $\{e_1(f, \boldsymbol{y}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N), \ldots, e_l(f, \boldsymbol{y}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N)\}$ denote a set of l selected block-matching constraints. In this case, one example constraint that considers all the above equations is of the form

$$e(f, \boldsymbol{y}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N) = \sum_{i=1}^{l} e_i(f, \boldsymbol{y}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N) = 0. \quad (12)$$

An additional benefit is that the constraints can be used to estimate the motions for the first and last frames in a sequence. The results in this paper are all computed by using only the initial constraint in (7).

3. Hierarchical algorithm for transparency and occlusion

By using the block-matching constraints above, a number of different algorithms for the estimation of multiple motions can be derived. We here present a hierarchical algorithm based on a combination of statistical model discrimination and hierarchical decision making. First, a single-motion model is fitted to the sequence by exhaustive search. If the fit is poor, the single-motion hypothesis is rejected and the algorithm tries to fit two transparent motions. Otherwise, the single motion is estimated and used. If the assumption of two transparent motions must also be rejected, the algorithm tries to fit an occlusion model, which will be developed later in this section, and estimates the occluded motions. The method can be extended to deal with an arbitrary number of transparent motions. The image noise is modeled as additive white Gaussian noise, thus leading to a significance test that evaluates χ^2 test statistics.

3.1 The stochastic image-sequence model

Apart from distortions and occlusions the block-matching constraint may be violated due to noise. Additional information about the distribution of the noise would help to determine whether or not the error signals observed after block-matching can be explained by the noise model. Different motion types lead to different noise distributions of the error signals. This can be helpful for selecting the most likely motion model.

We model the observed image intensity at each spatial location and time step as

$$f_k(\boldsymbol{x}) = \bar{f}_k(\boldsymbol{x}) + \epsilon_k(\boldsymbol{x}), \quad \epsilon_k(\boldsymbol{x}) \sim \mathcal{N}(0, \sigma^2) \quad (13)$$

for $k = 0, 1, \ldots$.

Therefore, from Equation (7) and the above noise model, we have

$$e(f, \boldsymbol{x}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N) = e(\bar{f}, \boldsymbol{x}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N) + \varepsilon_N(\boldsymbol{x}), \quad (14)$$

where

$$\varepsilon_N(\boldsymbol{x}) =$$
$$(-1)^N \epsilon_0(\boldsymbol{x} - \boldsymbol{v}_1 - \cdots - \boldsymbol{v}_N) + \cdots$$
$$- \sum_{i<l} \epsilon_{N-2}(\boldsymbol{x} - \boldsymbol{v}_i - \boldsymbol{v}_l) \quad (15)$$
$$+ \sum_i \epsilon_{N-1}(\boldsymbol{x} - \boldsymbol{v}_i) - \epsilon_N(\boldsymbol{x}).$$

There are 2^N terms in the right-hand side of the above equation. By assuming these terms to be statistically independent, we obtain

$$\varepsilon_N(\boldsymbol{x}) \sim \mathcal{N}(0, 2^N \sigma^2). \quad (16)$$

Block-matching constraint	Substitution
$f_0(\boldsymbol{x} - \boldsymbol{v}_1 - \boldsymbol{v}_2) - f_1(\boldsymbol{x} - \boldsymbol{v}_1) - f_1(\boldsymbol{x} - \boldsymbol{v}_2) + f_2(\boldsymbol{x}) = 0$	
$f_0(\boldsymbol{y} - \boldsymbol{v}_2) - f_1(\boldsymbol{y}) - f_1(\boldsymbol{y} + \boldsymbol{v}_1 - \boldsymbol{v}_2) + f_2(\boldsymbol{y} + \boldsymbol{v}_1) = 0$	$\boldsymbol{y} = \boldsymbol{x} - \boldsymbol{v}_1$
$f_0(\boldsymbol{y} - \boldsymbol{v}_1) - f_1(\boldsymbol{y} - \boldsymbol{v}_1 + \boldsymbol{v}_2) - f_1(\boldsymbol{y}) + f_2(\boldsymbol{y} + \boldsymbol{v}_2) = 0$	$\boldsymbol{y} = \boldsymbol{x} - \boldsymbol{v}_2$
$f_0(\boldsymbol{y}) - f_1(\boldsymbol{y} + \boldsymbol{v}_2) - f_1(\boldsymbol{y} + \boldsymbol{v}_1) + f_2(\boldsymbol{x} + \boldsymbol{v}_1 + \boldsymbol{v}_2) = 0$	$\boldsymbol{y} = \boldsymbol{x} - \boldsymbol{v}_1 - \boldsymbol{v}_2$

Table 1. Further block-matching constraints for two motions obtained by substitutions in the original constraint.

The hypothesis of noise independence fails when the arguments of the terms that involve the same image f_n in Equation (7) are equal. However, this cannot happen for less than three transparent motions. For four or more motions it may occur, e.g., that $\boldsymbol{v}_1 + \boldsymbol{v}_2 = \boldsymbol{v}_3 + \boldsymbol{v}_4$. This case can be detected during the search process and the variance can be adjusted accordingly. Hence, for a perfect match of the transparent motion model, the motion-compensated residual can be modeled as

$$e(f, \boldsymbol{x}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N) = \varepsilon_N(\boldsymbol{x}) \sim \mathcal{N}(0, 2^N \sigma^2). \qquad (17)$$

Consequently, the sum BM_N of squared differences over the block obeys the χ^2 distribution with $|\boldsymbol{B}|$ degrees of freedom, i.e.,

$$BM_N(\boldsymbol{x}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N) =$$
$$\frac{1}{2^N \sigma^2} \sum_{\boldsymbol{y} \in \boldsymbol{B}} e_N(f, \boldsymbol{y}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N)^2 \sim \chi^2(|\boldsymbol{B}|), \qquad (18)$$

where $\boldsymbol{B}$ is the set of pixels in the block under consideration and $|\boldsymbol{B}|$ is the number of elements in $\boldsymbol{B}$.

A block-matching algorithm can be designed such as to minimize the above expression. Other positive and strictly monotonic functions of the motion compensated residual could also be used as a criterion.

3.2 Single motion and two transparent motions

In the case of single motion the corresponding block-matching constraint is defined as the difference between the motion compensated image and the next image. Hence the function to be minimized is

$$BM_1(\boldsymbol{x}, \boldsymbol{v}) = \frac{1}{2\sigma^2} \sum_{\boldsymbol{y} \in \boldsymbol{B}} \left(f_0(\boldsymbol{y} - \boldsymbol{v}) - f_1(\boldsymbol{y}) \right)^2. \qquad (19)$$

Similarly, for two motions the expression

$$BM_2(\boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2) = \frac{1}{4\sigma^2} \sum_{\boldsymbol{y} \in \boldsymbol{B}} e(f, \boldsymbol{y}, \boldsymbol{v}_1, \boldsymbol{v}_2)^2 \qquad (20)$$

needs to be minimized with respect to $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$. If there is only one motion inside $\boldsymbol{B}$, i.e. $f_1(\boldsymbol{x}) = f_0(\boldsymbol{x} - \boldsymbol{v})$, the value $BM_1(\boldsymbol{x}, \boldsymbol{v})$ will even in the ideal noise-free case, be small for the correct motion vector $\boldsymbol{v}$. But if $\boldsymbol{B}$ includes

two motions, the value BM_1 will be different from zero for any vector $\boldsymbol{v}$, because one vector cannot compensate for two motions. Accordingly, in case of two transparent motions, $BM_2(\boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2)$ will be small if we insert the two correct motion vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$. Such a Gaussian model has been previously used in e.g. [2, 4], but may alternatively be replaced by Generalized Gaussian models [1, 19].

3.3 Occluded motions

In case of occluded motions, Equations (8) and (9) are no longer valid because Equation (1) defines an additive superposition that does not hold at occlusions. We model the occlusion of the layer g_2 by the occluding layer g_1 by

$$f_k(\boldsymbol{x}) = \gamma(\boldsymbol{x} - k\boldsymbol{v}_1)g_1(\boldsymbol{x} - k\boldsymbol{v}_1) + (1 - \gamma(\boldsymbol{x} - k\boldsymbol{v}_1))g_2(\boldsymbol{x} - k\boldsymbol{v}_2), \qquad (21)$$

with $\gamma = 1$ at positions where g_1 occludes g_2 and $\gamma = 0$ otherwise, see [12]. By evaluating the error criterion (9) for two transparent motions in combination with the above model for occlusions we obtain

$$e(f, \boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2) = \big(\gamma(\boldsymbol{x} - 2\boldsymbol{v}_1) - \gamma(\boldsymbol{x} - \boldsymbol{v}_1 - \boldsymbol{v}_2) \big) \big(g_2(\boldsymbol{x} - \boldsymbol{v}_1 - \boldsymbol{v}_2) - g_2(\boldsymbol{x} - 2\boldsymbol{v}_1) \big). \qquad (22)$$

Depending on the motion vectors, the difference between the γ-function terms on the right hand-side of the above equation may be different from zero. This will be the case in the vicinity of an occluding boundary. Therefore, if we intend to apply the block-matching error criterion for transparent motions to estimate two motions at the occluding boundary, the value of $BM_2(\boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2)$ will be high although $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are the correct motion vectors. The size of the region around the occluding boundary where this happens depends on the difference between the velocities. In fact, by replacing $\boldsymbol{y} = \boldsymbol{x} - 2\boldsymbol{v}_1$ in the right-hand side of the above equation, we find

$$e(f, \boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2) = \big(\gamma(\boldsymbol{y}) - \gamma(\boldsymbol{y} + \boldsymbol{v}_1 - \boldsymbol{v}_2) \big) \big(g_2(\boldsymbol{y} + \boldsymbol{v}_1 - \boldsymbol{v}_2) - g_2(\boldsymbol{y}) \big), \qquad (23)$$

which means that the distortion is restricted to a strip with a maximum width of $|\boldsymbol{v}_1 - \boldsymbol{v}_2|$. For the simplest case of a straight-line border, the strip is $|\boldsymbol{N} \cdot (\boldsymbol{v}_1 - \boldsymbol{v}_2)|$ wide, where $\boldsymbol{N}$ is the unit vector normal to the border. Due to this distortion, it is not guaranteed that the minimum of BM_2 yields the

correct motion vectors. The problem of estimating two motions at the occluding boundary can be reduced to the problem of transparent motions if we exclude the region of distortion from the calculation of the residual error. Therefore, however, we need to find the location of the occluding boundary. A more formal treatment of motions at the occluding boundary is given in [6, 5].

3.4   Selection of the adequate transparent-motion model

An obvious possibility to find the most adequate local motion model would be to use a discriminant functions or, for the simple cases of one or two motions, a likelihood ratio test. To this end, one would have to find the minimum block-matching values for all motion models before the test can be carried out.

Instead, we opt for a computationally more efficient significance test, which allows for a hierarchical estimation of the motion vectors. From the discussion in the previous section, $BM_N(\boldsymbol{x}, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N)$ is χ^2-distributed with $|\boldsymbol{B}|$ degrees of freedom. If we allow a percentage α of misclassification, we can derive a threshold T_N for BM_N as follows: let the null-hypothesis H_0 mean that the model of N transparent motion is correct. T_N is then determined by

$$\text{Prob}(BM_N > T_N | H_0) = \alpha. \qquad (24)$$

Thus, H_0 is rejected if $BM_N > T_N$. The correct threshold can be obtained by using tables of the χ^2 distribution.

3.5   The hierarchical algorithm

We will now show how to integrate the above considerations into a hierarchical algorithm that can deal with the above mentioned cases of single, transparent and occluded motions. The rationale is that we estimate the confidence for local motion models of increasing complexity and then estimate the motion parameters according to the adequate model. The hierarchical algorithm is described below and summarized in Algorithm 1. An extension to more than two motions is conceptually straightforward.

The algorithm first finds by full search the motion vector $\boldsymbol{v}$ that minimizes BM_1. It then tests whether or not this value is explainable by the underlying noise model: if $BM_1(\boldsymbol{x}, \boldsymbol{v}) < T_1$ one motion is assigned to the current location. Otherwise, it proceeds to find the motion vectors $\boldsymbol{v}_1, \boldsymbol{v}_2$ that minimize BM_2 and then tests for $BM_2(\boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2) < T_2$. If both motion models are rejected, the location is marked as belonging to an occlusion boundary. In the second iteration we determine motion vectors for the above marked locations only. The algorithm is then iterated at the marked locations and the size of the block is increased at each iteration to ensure that there are sufficient non-marked pixels in the block. The estimation of the motion vectors for the marked pixels is based on non-marked pixels only, because the marked pixels violate the assumption of one or two additive motions and

---

Algorithm 1 Hierarchical algorithm

1: Compute thresholds T_1 and T_2
2: **for all** pixels **do**
3:    Compute minimum value of BM_1 and the corresponding motion vector.
4:    **if** $BM_1 < T_1$ **then**
5:       Choose single-motion model
6:    **else**
7:       Compute the minimum value of BM_2 and the two motion vectors
8:       **if** $BM_2 < T_2$ **then**
9:          Choose model for two transparent motions
10:      **else**
11:         Mark pixel
12:      **end if**
13:   **end if**
14: **end for**
15: Increase block size and repeat lines 3 to 14 for all marked pixels. Ignore marked pixels inside the current block and recompute T_1 and T_2 according to the number of non-marked pixels in the block.

---

thus the minimization of either BM_1 or BM_2 would not render the correct motions. The iteration is repeated until motion vectors are found for all marked pixels or a maximum number of iterations is reached. For each marked pixel the thresholds have to be adapted according to the number of non-marked pixels in the block (degrees of freedom). This two-phase approach enables us to compute two motions at the occluding boundary by avoiding the non-zero terms on the right side of Equation (23).

4. Motion estimation using Markov random fields

The algorithms proposed in the previous sections are based on spatio-temporal relations of image intensity but do not consider spatial and temporal relationships between the motion vectors. This seems unsatisfactory because regions corresponding to moving objects tend to be of compact shape with smooth motion-vector fields. Sets of isolated moving points with non-smooth motion vector fields are unlikely to occur in natural images. Regularization of the motion vector fields is widely used for optical flow estimation, and has been proposed for multiple motions as well [21]. Since we here view motion estimation as making statistical observations, we choose to increase robustness against noise by using a stochastic framework based on Markov random fields (similar to how it was used in [3] for motion detection and in [4] for single motion estimation) in combination with the block-matching constraint. This approach has three major benefits: firstly, it allows to select the most probable motion model (the correct number of observed motion vectors) in the presence of noise; secondly, it ensures the spatio-temporal smoothness of the motion fields; thirdly, it simultaneously provides a segmentation of the images based on the local number of mo-

tions. In the following we will present a detailed estimation algorithm for up to two transparent motions. A generalization to more than two motions is straightforward.

4.1 Bayesian formulation for two motions

For each pixel $\boldsymbol{x}$ and time step k, we seek the motion vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ and a binary segmentation value $s \in \{0, 1\}$, which represents the number of observed motions at this particular pixel. The segmentation takes the value $s = 0$ in regions with a single motion and $s = 1$ in regions with two transparent motions. Static patterns are a special case of single motion with zero velocity. The aim is to estimate the tuple $\boldsymbol{u}_k(\boldsymbol{x}) = (\boldsymbol{v}_1(\boldsymbol{x}), \boldsymbol{v}_2(\boldsymbol{x}), s(\boldsymbol{x}))$ at each pixel using $N+1 = 3$ successive images. According to the maximum a posteriori principle we wish to estimate the most probable segmentation and motion vector fields for the current frame given the observations f_k, f_{k-1}, f_{k-2}. The estimated field $\boldsymbol{u}_k = \{\boldsymbol{u}_k(\boldsymbol{x})\}$ hence satisfies

$$\boldsymbol{u}_k = \arg\max_{\boldsymbol{u}} p(\boldsymbol{u}|f_k, f_{k-1}, f_{k-2}), \qquad (25)$$

where $p(\boldsymbol{u}|f_k, f_{k-1}, f_{k-2})$ is the posterior PDF for a tuple $\boldsymbol{u}$ given the observations f_k, f_{k-1}, f_{k-2}. Invoking Bayes' theorem, we rewrite the above relations as

$$\boldsymbol{u}_k = \arg\max_{\boldsymbol{u}} p(f_k, f_{k-1}, f_{k-2}|\boldsymbol{u}) \, p(\boldsymbol{u}). \qquad (26)$$

The prior pdf $p(\boldsymbol{u})$ ensures that this estimate is consistent with our smoothness expectations and the conditional PDF $p(f_k, f_{k-1}, f_{k-2}|\boldsymbol{u})$ describes the relationship between the observed images and the unknown motion fields.

4.2 The observation model

The segmentation result provides the number of observed motions at each pixel. Depending on this segmentation, we have to select the corresponding motion model to specify the likelihood $p(f_k, f_{k-1}, f_{k-2}|\boldsymbol{u})$. From Section 3.1, we know that the motion compensated difference (7) is $\mathcal{N}(0, 2^N \sigma^2)$-distributed. We wish to use BM_1 if the segmentation s indicates that there is only one motion, i.e. $s(\boldsymbol{x}) = 0$. Otherwise, we would switch to the model for two motions (BM_2).

The following constraint formalizes the above requirements:

$$
\begin{aligned}
p(f_k, & f_{k-1}, f_{k-2}|\boldsymbol{u}) \propto \\
& \prod_{\boldsymbol{x}} \Big[(1 - s(\boldsymbol{x}))(4\pi\sigma^2)^{-|\boldsymbol{B}|/2} e^{-BM_1(\boldsymbol{x}, \boldsymbol{v}_1(\boldsymbol{x}))} \\
& + s(\boldsymbol{x})(8\pi\sigma^2)^{-|\boldsymbol{B}|/2} e^{-BM_2(\boldsymbol{x}, \boldsymbol{v}_1(\boldsymbol{x}), \boldsymbol{v}_2(\boldsymbol{x}))} \Big].
\end{aligned}
\qquad (27)
$$

To extend the above equation to the case of more motions, one would replace the expression $(1 - s(\boldsymbol{x}))$ by a segmentation function $s_1(\boldsymbol{x})$, and $s(\boldsymbol{x})$ by a second segmentation function $s_2(\boldsymbol{x})$. Next, we will specify the prior $p(\boldsymbol{u})$, which completes the observation model.

4.3 Spatial smoothness

The specification of the joint density $p(\boldsymbol{u})$ of all tuples $\boldsymbol{u}(\boldsymbol{x})$ should make the expected motion fields more likely than others. The Markov assumption simplifies the specification by definition of only local statistical dependencies of the tuples. Invoking the Hammersley-Clifford theorem, we can write $p(\boldsymbol{u})$ as a Gibbs density function:

$$p(\boldsymbol{u}) = \frac{1}{Z} e^{-\lambda E(\boldsymbol{u})}, \qquad (28)$$

with Z being a normalization constant. The parameter λ controls the influence of the smoothing. The energy $E(\boldsymbol{u})$ should therefore be small in case of locally smooth vector fields and segmentations. Due to the Markovian assumption, $E(\boldsymbol{u})$ can be divided into two local energy terms $E_L(\boldsymbol{x}, \boldsymbol{u})$ according to

$$E(\boldsymbol{u}) = \sum_{\boldsymbol{x}} E_L(\boldsymbol{x}, \boldsymbol{u}). \qquad (29)$$

The local energy terms $E_L(\boldsymbol{x}, \boldsymbol{u})$ depend on the motion vectors, the segmentation values at pixel $\boldsymbol{x}$ and at pixels in the neighborhood $N_{\boldsymbol{x}}$. In our case, the neighborhood $N_{\boldsymbol{x}}$ contains the eight pixels that are adjacent to pixel $\boldsymbol{x}$. To capture both the smoothness of the velocities and the segmentation, we divide the local energy terms into two parts. The term E_{L_s} measures the smoothness of the segmentation and E_{L_v} the smoothness of the motion fields, such that

$$E_L(\boldsymbol{x}, \boldsymbol{u}) = E_{L_s}(\boldsymbol{x}, \boldsymbol{u}) + E_{L_v}(\boldsymbol{x}, \boldsymbol{u}). \qquad (30)$$

To obtain locally smooth motion vector fields we penalize differences between adjacent motion vectors at two neighboring positions $\boldsymbol{x}$ and $\boldsymbol{y}$. In case where we have two motion vectors at both locations, we denote with the same index those vectors, which are more similar, i.e. $\boldsymbol{v}_1(\boldsymbol{x})$ is closer to $\boldsymbol{v}_1(\boldsymbol{y})$ than to $\boldsymbol{v}_2(\boldsymbol{y})$. This becomes problematic in cases where the number of motions varies in a given neighborhood, for instance due to object boundaries. In such cases, again based on the assumption of transparent layers, we apply the smoothness constraint to the single motion and only one of the adjacent two motions. The one motion is chosen as the one closest to the single motion and denoted with $\boldsymbol{v}_1$. With this convention, the local smoothness term is

$$
\begin{aligned}
E_{L_v}(\boldsymbol{x}, \boldsymbol{u}) = \sum_{\boldsymbol{y} \in N_{\boldsymbol{x}}} \big(& \|\boldsymbol{v}_1(\boldsymbol{x}) - \boldsymbol{v}_1(\boldsymbol{y})\|^2 \\
& + s(\boldsymbol{x})s(\boldsymbol{y})\|\boldsymbol{v}_2(\boldsymbol{x}) - \boldsymbol{v}_2(\boldsymbol{y})\|^2 \big).
\end{aligned}
\qquad (31)
$$

The above requirement is now controlled by the term $s(\boldsymbol{x})s(\boldsymbol{y})$.

The function E_{L_s} is specified in the same way as in [3] where it was used for motion segmentation. The procedure consists of counting the number of pixels inside the neighborhood $N_{\boldsymbol{x}}$ that have the same segmentation value $s(\boldsymbol{x})$. The
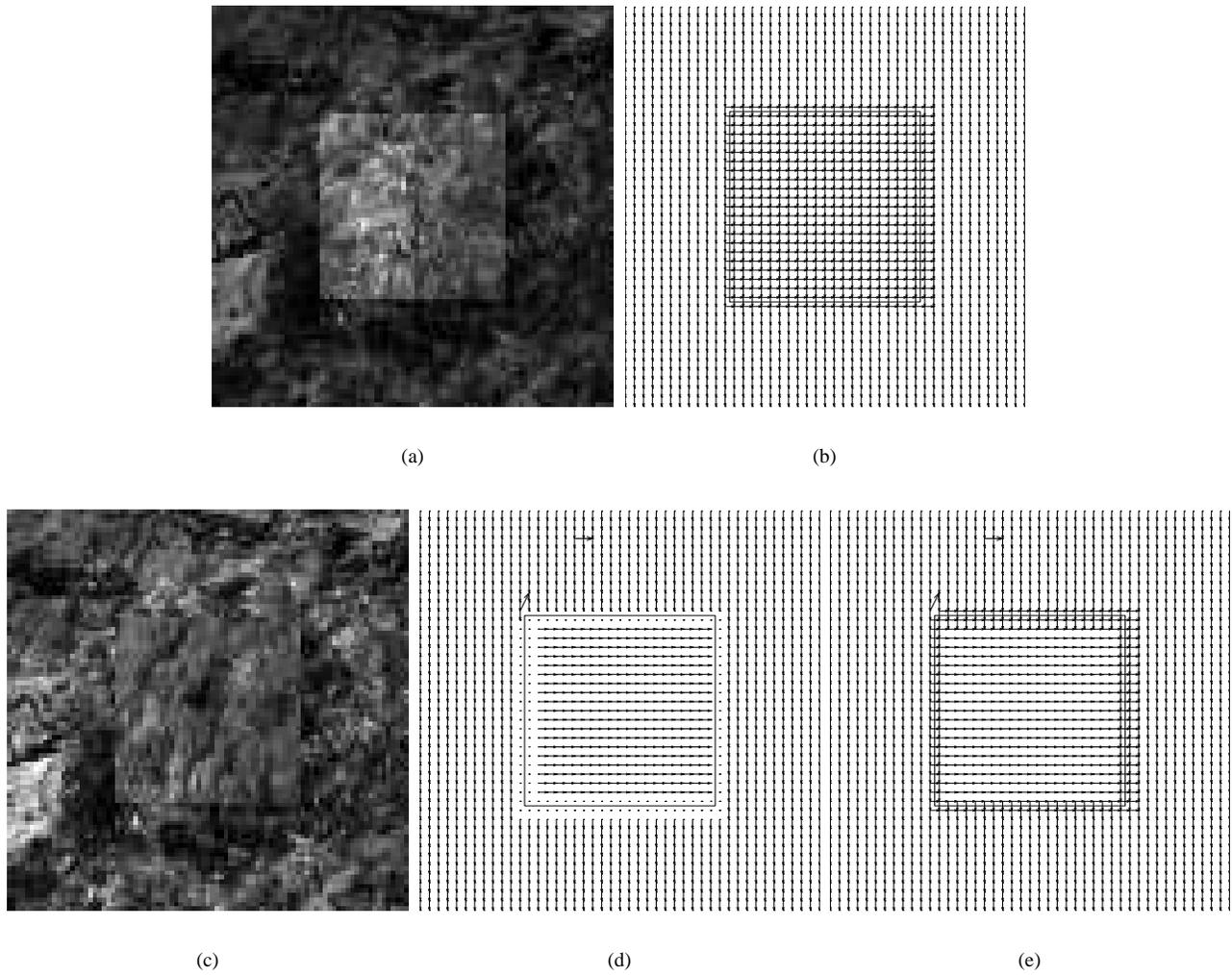
Figure 1. Results for transparent and occluded motions. Image (a) shows a frame of the transparent-motion test sequence and (b) the estimated motion vectors. Image (c) depicts a frame of an occlusion test sequence, (d) and (e) the estimated vectors after the first and second phase, respectively.

resulting number is then subtracted from the maximum number of pixels with equal segmentation values (which is eight). The local segmentation energy is defined by

$$E_{L_s}(\boldsymbol{x}, \boldsymbol{u}) = 8 - w_{N_{\boldsymbol{x}}}, \qquad (32)$$

where $w_{N_{\boldsymbol{x}}}$ denotes the number of pixels in $N_{\boldsymbol{x}}$ having the same segmentation value as the pixel $\boldsymbol{x}$. This energy term is minimal if all pixels inside the neighborhood are of the same motion type as the considered pixel. Obviously, if the number $w_{N_{\boldsymbol{x}}}(s(\boldsymbol{x}))$ increases, the probability for the pixels $\boldsymbol{x}$ to be classified as having the common motion type will also increase.

4.4 The optimization algorithm

The function to be maximized in Equation (26) is defined as the product of (27) and (28) with corresponding energies given by (31) and (32), respectively. By use of the negative logarithm, its maximization is equivalent to the minimization of

$$\begin{aligned}
C(f_2, f_1, f_0 | \boldsymbol{u}) = \sum_{\boldsymbol{x}} \Big[&\big(1 - s(\boldsymbol{x})\big) BM_1(\boldsymbol{x}, \boldsymbol{v}_1(\boldsymbol{x})) \\
&+ s(\boldsymbol{x}) BM_2(\boldsymbol{x}, \boldsymbol{v}_1(\boldsymbol{x}), \boldsymbol{v}_2(\boldsymbol{x})) \Big] \\
&+ \lambda E(\boldsymbol{u}) + \log(\sqrt{2}) |\boldsymbol{B}||s|,
\end{aligned} \qquad (33)$$

where $|s| = \sum_{\boldsymbol{x}} s(\boldsymbol{x})$. The constants that do not influence the minimization have been dropped.

We minimize this criterion by using deterministic relaxation of the ICM-type [9] although this procedure does not necessarily converge to the global minimum of the functional.

An annealing algorithm which is able to find the global minimum has been proposed in [13] but is not used here because it is too slow. For each frame, the optimization routine starts with the result obtained for the previous frame as an initial guess. In case of sequences with fast motions, this might not be a good guess. One possibility to overcome this problem is to start with a motion-predicted initial guess. As long the motion does not change abruptly such a prediction is close to the actual motion vectors and provides temporal regularization. For some applications, however, an explicit temporal smoothing could improve the results. We outline the necessary modifications of the algorithm below.

4.5 Temporal Smoothness

The estimate u_{k-1} is available when u_k is estimated. For simplicity, successive images and u_{k-1} are modeled as being conditionally independent, i.e.,

$$p(f_k, f_{k-1}, f_{k-2}, u_{k-1}|u) = \\ p(f_k, f_{k-1}, f_{k-2}|u)\, p(u_{k-1}|u). \tag{34}$$

The MAP-estimate is then given by

$$u_k = \arg \max_{u} p(f_k, f_{k-1}, f_{k-2}|u)\, p(u_{k-1}|u)\, p(u). \tag{35}$$

The new estimation criterion includes an additional component $p(u_{k-1}|u)$ that captures the relation between the previous motion fields u_{k-1} and the current ones.

To specify $p(u_{k-1}|u)$, we again use a Gaussian model and make two simplifications: firstly, each vector $v^k = v^k(x)$ depends (implicitly) only on its predecessor $v^{k-1}(x+v^k)$ along the motion trajectory, thus making it more likely that both vectors in the likelihood-term belong to the same object; secondly, we assume conditional statistical independence. The likelihood now simplifies to

$$p(u_{k-1}|u) = \prod_{x} p(v_1^{k-1}(x + v_1^k)|v_1^k)p(v_1^{k-1}(x + v_2^k)|v_2^k),$$

$$\tag{36}$$

with

$$p(v_i^{k-1}(x + v_i^k)|v_i^k) = \\ \frac{1}{Z_T} \exp(-\lambda_T \|(v_i^{k-1}(x + v_i^k) - v_i^k\|^2) \tag{37}$$

for i=1,2, where Z_T is a normalization constant, and λ_T a weighting factor. The above term can be included in the optimization algorithm defined by Equation (33).

5. Results

5.1 Results for the confidence-based hierarchical algorithm

In Figure (1) examples of transparent and occluded motions are given. Image (a) shows the center frame of an images sequence containing areas with only one and with two transparent motions. The area with two transparent motions can be identified as the brighter box-shaped part in the image. One layer is moving with a velocity of one pixel per frame to the right and the other layer with one pixel per frame downwards. The estimated motion vectors for every other pixel are depicted in (b). The rectangle in (b) marks the outline of the area with two motions. Note that in both areas the motions are correctly estimated. For this example we used a block-size of 5×5 pixels. Image (c) shows the center frame of an occlusion test sequence and the images (d) and (e) the results after the first and second phase of the algorithm respectively. Again, motion vectors are plotted only for every other pixel. The motions for both regions are very well detected except for a few outliers. After the second phase, two motions are estimated in an area around the occluding boundary, where no motion could be computed in the first phase. Window sizes of 5×5 and 9×9 were used for first and second phase, respectively. In both examples Gaussian distributed noise was added to the sequences resulting in a signal-to-noise ratio of 35 dB. For the significance test we set $\alpha = 0.001$. Image (a) in Figure 2 shows the setup for a real occlusion example. A toy train that carries a texture image is moving to the right in front of a leftward moving background texture. Both movements are approximately one pixel per frame. The rectangle in (a) marks the region for which we show the estimated motions vectors. Image (b) depicts this selected region and also shows the outline of the region where no motions could be estimated after the first phase of the algorithm (due to the occluding boundary). The size of this region roughly matches the block-size of 7×7 pixels. The estimated motion vectors obtained after the first phase are shown in (c). Note that, except for a view outliers, the estimates are accurate. The final motion vectors (after the second phase) are divided for better visualization and shown in the two panels (d) and (e). Image (d) shows the leftward motions and image (e) the rightward motions. Note that both motions in the region around the occluding boundary are accurately estimated. For the second phase we used a block-size of 18×18 pixels. In the first phase we set $\alpha = 0.001$ for one motion and $\alpha = 0.05$ for two motions (since we did not expect transparent motions). In the second phase we had $\alpha = 0.001$ for both cases.

5.2 Results obtained with the Markov-random-fields extension

Figure (3) demonstrates the performance of the Markov Random Field approach. The test sequence is the same as the one used for testing the hierarchical algorithm but with a signal-to-noise ratio of only 17 dB. We initialized the algorithm with one motion and zero velocity everywhere. Image (a) shows one frame of the test sequence, image (b) the estimated motion vectors (plotted for every other pixel) for the MRF algorithm after three iterations, and (c) the estimated motions obtained by using the hierarchical algorithm. Again the square frame marks the outline of the area with two motions. For both algorithms we used a block-size of 3×3
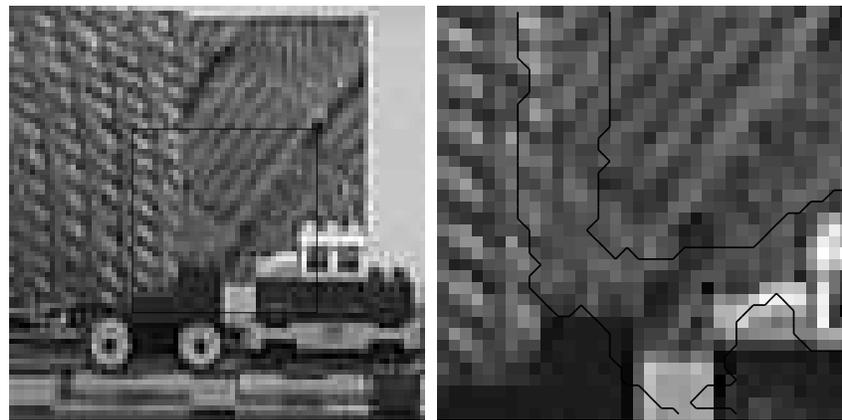
(a)

(b)
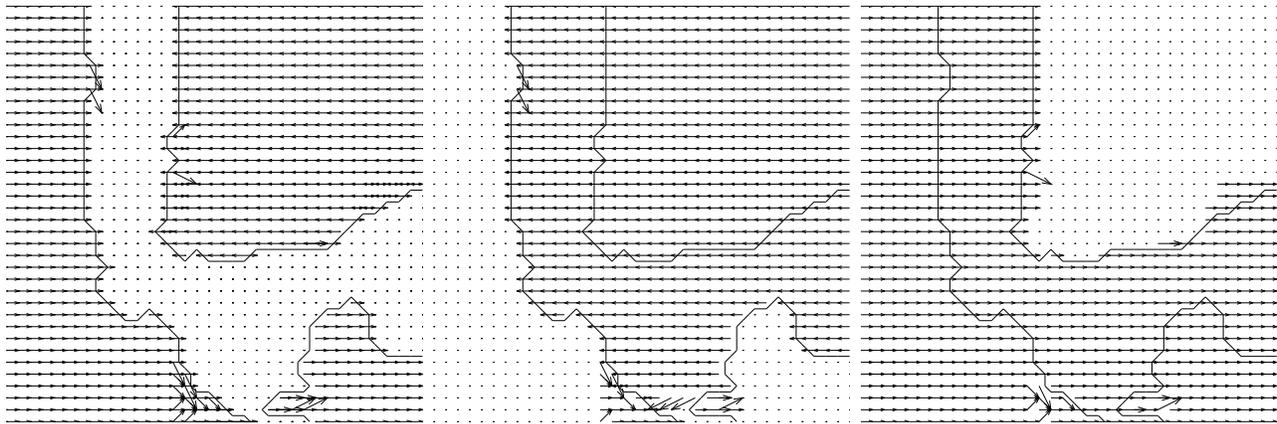
(c)

(d)

(e)

Figure 2. Results for real videos with occluding objects. Image (a) depicts a single frame of the sequence, (b) the area for which the motion vectors are shown, (c) the motion vectors after the first phase, (d) and (e) the estimated motions after the second phase, divided into two images for better visualization. See text for details.

pixels. The parameter $\lambda = 1$ was used for the MRF approach and $\alpha = 0.001$ for the hierarchical algorithm. Note that the MRF algorithm has considerably less outliers than the hierarchical algorithm. In some small regions of the MRF-based output the motions remain zero as initialized. The results obtained eight frames later are depicted in the second row of figure (3). Again we observe less outliers for the MRF algorithm and now the regions with zero velocities disappeared.

6. Conclusions

We first derived a block-matching constraint for an arbitrary number of transparent overlaid motions. To estimate N motions, $N + 1$ images and 2^N blocks are needed. We then analyzed how the block-matching constraint behaves near the

occluding boundary in case of occluded motions. Based on these theoretical results we developed a hierarchical algorithm, which enables the estimation of single, multiple transparent, and occluded motions. The estimation of occluded motions is performed in a second phase by excluding the pixels near the occluding boundary found in the first phase. The hierarchical algorithm has been tested with both real video sequences and synthetic images corrupted by additive noise. It performs well for a SNR down to 30 dB which is a typical value for low-end cameras. Nevertheless, for some applications, e.g. sequences resulting from medical imagery, a more robust algorithm may be needed. We therefore derived a regularized version of the block-matching algorithm for transparent motions by using Markov Random Fields and have shown that this extension significantly increases robustness.
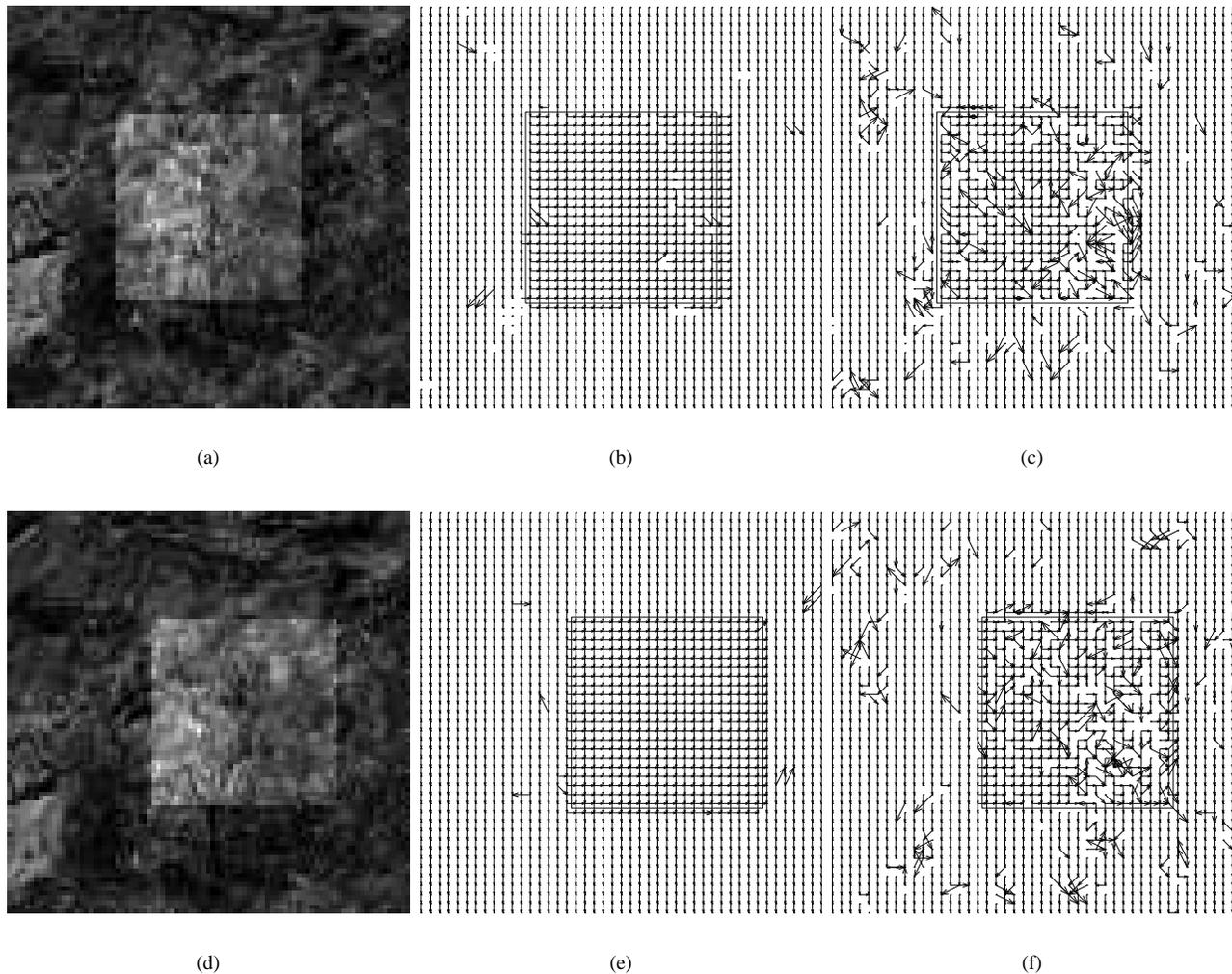
Figure 3. Results for transparent motions obtained with MRFs: Image (a) shows a frame of the transparent motion sequence with a SNR of 17 dB, (b) the estimated velocities for the MRF-approach and (c) the results for the hierarchical algorithm. The second row show the estimated motion vectors eight frames later.

7. Acknowledgment

8. References

[1] T. Aach. *Bayes-Methoden zur Bildsegmentierung, Änderungsdetektion und Verschiebungsvektorschätzung*. Fortschrittberichte VDI Reihe 10, Nr. 261, VDI Verlag, Düsseldorf, 1993. Dissertation, RWTH Aachen.

[2] T. Aach and A. Kaup. Disparity-based segmentation of stereoscopic foreground/background imagesequences. *IEEE Transactions on Communications*, 42(2):673–679, 1994.

[3] T. Aach and A. Kaup. Bayesian algorithms for adaptive change detection in image sequences using Markov random fields. *Signal Processing: Image Communication*, 7(2):147–160, 1995.

[4] T. Aach and D. Kunz. Bayesian motion estimation for temporally recursive noise reduction in x-ray fluoroscopy. *Philips Journal of Research*, 51(2):231–251, 1998.

[5] E. Barth, I. Stuke, T. Aach, and C. Mota. Spatio-temporal motion estimation for transparency and occlusion. In *Proceedings of the IEEE International Conference on Image Processing*, volume III, pages 69–72, September 2003.

[6] E. Barth, I. Stuke, and C. Mota. Analysis of motion and curvature in image sequences. In *Proceedings of the 5th IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 206–210, 2002.

[7] J. Bergen, P. Burt, R. Hingorani, and S. Peleg. A three-frame algorithm for estimating two-component image motion. *PAMI*, 14(9):886–896, September 1992.

[8] M. A. Bertero, T. Poggio, and V. Torre. Ill-posed problems in early vision. *Proceedings of IEEE*, 76(8):869–889, 1988.

[9] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Stat. Soc.*, 48(3):259–302, 1986.

[10] T. Darrell and E. Simoncelli. On the use of "nulling" filters to separate transparent motions. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, New York, 1993. IEEE Computer Society Press.

[11] T. Darrell and E. Simoncelli. Separation of transparent motion into layers using velocity-tuned mechanisms. In *European Conf on Computer Vision*. Springer Verlag, 1994.

[12] D. J. Fleet and K. Langley. Computational analysis of non-fourier motion. *Vision Research*, 34(22):3057–3079, 1995.

[13] S. Geman and D. Geman. Stochastic relaxation, gibbs distribution, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analyzes and Machine Intelligence*, 6(6):721–741, 1984.

[14] K. Langley. Computational models of coherent and transparent plaid motion. *Vision Research*, 39 (1):87–108, 1999.

[15] C. Mota, I. Stuke, and E. Barth. Analytic solutions for multiple motions. In *Proceedings of the International Conference on Image Processing*, pages 917–920, 2001.

[16] E. Pingault, M Bruno and D. Pellerin. A robust multiscale B-spline function decomposition for estimating motion transparency. *IEEE Transactions on Image Processing*, 12(11):1416–1426, November 2003.

[17] M. Shizawa and K. Mase. Simultaneous multiple optical flow estimation. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 274–8, Atlantic City, June 1990.

[18] E. P. Simoncelli. Distributed representation and analysis of visual motion. Technical Report 209, MIT Media Laboratory, Cambridge, MA, 1993.

[19] C. Stiller. Object-based estimation of dense motion fields. *IEEE Transactions on Image Processing*, 6(2):1111–1117, 1997.

[20] I. Stuke, T. Aach, E. Barth, and C. Mota. Estimation of multiple motions by block matching. In W. Dosch and R. Y. Lee, editors, *Proceedings of the ACIS Fourth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD'03)*, pages 358–362, October 2003.

[21] I. Stuke, T. Aach, C. Mota, and E. Barth. Estimation of multiple motions: regularization and performance evaluation. In B. Vasudev, R. Hsing, A. Tescher, and T. Ebrahimi, editors, *SPIE/IST Electronic Imaging 2003*, volume 5022. SPIE, 2003.

[22] R. Szeliski, S. Avidan, and P. Anandan. Layer extraction from multiple images containing reflections and transparency. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 246–253, "Hilton Head Island, South Carolina", 2000.

[23] D. Vernon. Decoupling Fourier components of dynamic image sequences: a theory of signal separation, image segmentation and optical flow estimation. In *ECCV*, pages 68–85. Springer Verlag, 1998.

[24] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *The IEEE Transactions on Image Processing*, 3(5):625–638, 1994.

[25] A. B. Watson and A. J. Ahumada. Look at motion in the frequency domain. *J. K. Tsotos,editor, Motion: perception and representation*, pages 1–10, 1983.

[26] W. Yu, K. Daniilidis, S. Beauchemin, and G. Sommer. Detection and characterization of multiple motion points. In *IEEE Conf. Computer Vision and Pattern Recognition*, volume I, pages 171–177, June 1999. Fort Collins, CO.

Figure 4: Ingo Stuke

Ingo Stuke is currently a scientist at the Institute for Signal processing, University of Luebeck, Germany. His main research interests are on the field of motion and orientation estimation for image processing and computer vision applications. From 1995 to 2001 he studied computer science at the University of Luebeck.

Figure 5: Til Aach

Til Aach received his diploma and Doctoral degrees, both in EE, from Aachen University of Technology (RWTH) in 1987 and 1993, respectively. From 1993 to 1998, he was with Philips Research Labs, Aachen, Germany, where he was responsible for several projects in X-Ray imaging and processing. In 1996, he was also an independent lecturer with the University of Magdeburg, Germany. In 1998, he was appointed a full professor and director of the Institute for Signal Processing, University of Luebeck. His research interests are in medical and industrial image processing, signal processing, pattern recognition, and computer vision. He has authored or co-authored over 120 papers, and received several awards. He is an Associate Editor of the IEEE Transactions on Image Processing, and was Technical Program Co-Chair for the Southwest Symposium on Image Analysis and Interpretation (SSIAI) in 2000, 2002 and 2004. Til Aach is a Senior Member of IEEE.

Figure 6: Erhardt Barth

Erhardt Barth Dr. Barth is currently a scientist at the Institute for Neuro- and Bioinformatics, University of Luebeck, Germany. His main interests are in the areas of human and computer vision. He studied electrical and communications engineering at the University of Brasov and at the Technical University of Munich (Ph.D. 1994). He has been with the Universities of Melbourne and Munich, the Institute for Advanced Study in Berlin, the NASA Vision Science and Technology Group in California, and the Institute for Signal Processing in Luebeck. In May

2000 he received a Schloessmann Award.

Figure 7: Cicero Mota

Cicero Mota Cicero Mota is on leave from University of Amazonas, Brazil and is currently with the Institute for Neuro- and Bioinformatics, University of Luebeck, Germany. His research interests are in computer vision and information theory. Dr. Mota studied mathematics at the University of Amazonas and at the Institute for Pure and Applied Mathematics – IMPA (Ph.D. 1999) in Brazil. He has been with the Institute for Signal Processing, University of Luebeck (visiting scientist) and with the Intistute for Apllied Physics, University of Frankfurt, both in Germany. His research has been suported by the Brazilian agencies CNPq and CAPES, and by the german agencies DAAD and DFG.