J Math Imaging Vis (2011) 40: 285–304
DOI 10.1007/s10851-011-0260-8

Variational Multi-Valued Velocity Field Estimation for Transparent Sequences

Alonso Ramírez-Manzanares · Mariano Rivera · Pierre Kornprobst · François Lauze

Published online: 4 February 2011

Abstract Motion estimation in sequences with transparencies is an important problem in robotics and medical imaging applications. In this work we propose a variational approach for estimating multi-valued velocity fields in transparent sequences. Starting from existing local motion estimators, we derive a variational model for integrating in space and time such a local information in order to obtain a robust estimation of the multi-valued velocity field. With this approach, we can indeed estimate multi-valued velocity fields which are not necessarily piecewise constant on a layer—each layer can evolve according to a non-parametric optical flow. We show how our approach outperforms existing methods; and we illustrate its capabilities on challenging experiments on both synthetic and real sequences.

Keywords Transparent optical flow · Image regularization · Multiple motions · RDK sequences

A. Ramírez-Manzanares (✉)
Department of Mathematics, University of Guanajuato, AP 402,
Guanajuato, Gto., C.P. 36000, Mexico
e-mail: alram@cimat.mx

M. Rivera
Centro de Investigacion en Matematicas A.C., AP 402,
Guanajuato, Gto., 36000, Mexico

P. Kornprobst
INRIA, Odyssée Lab., 2004 route des Lucioles BP 93,
06902 Sophia Antipolis, France

F. Lauze
Institute of Computer Science, University of Copenhagen,
Universitetsparken 5, 2100 Kbh Ø, Denmark

1 Introduction

There exists a very wide literature on apparent motion estimation, also called Optical Flow (OF), due to the number of applications that require motion estimation and the complexity of the task. Motion estimation methods rely on conservation of a function of the recorded signal, generally luminance, or some of its derivatives across time and on spatial or spatiotemporal regularity constraints.

We first introduce the main notation: let $\Omega \subset \mathbb{R}^2$, then the function $f : (\mathbf{x}, t) \in \Omega \times \{0, \dots, T\} \to \mathbb{R}$ denotes the greyscale sequence or "image sequence", defined as a volume over space $\mathbf{x} = (x_1, x_2)$ and time t, and $\mathbf{u} = (u^1(\mathbf{x}, t), u^2(\mathbf{x}, t))^T$ denotes "its" motion field.

The simplest conservation principle, often referred to as the *Lambertian assumption*, states that the intensity of a point remains constant along its trajectory. Thus, in the case of a time discrete sequence, it provides the classical *Displaced Frame Difference* equation (DFD)

$$f(\mathbf{x}, t) - f(\mathbf{x} - \mathbf{u}, t - 1) = 0 \tag{1}$$

or its differential approximation, the well known *Optical Flow Constraint* equation (OFC)

$$\left(u^1\partial_{x_1} + u^2\partial_{x_2} + \partial_t\right) f(\mathbf{x}, t) = \nabla_{\mathbf{3}} f(\mathbf{x}, t) \cdot \begin{pmatrix} u^1 \\ u^2 \\ 1 \end{pmatrix} = 0, \tag{2}$$

where $\nabla_{\mathbf{3}} f$ is the spatiotemporal gradient $(f_{x_1}, f_{x_2}, f_t)^T$ of f (we will use the notation ∇f for the *spatial* gradient $(f_{x_1}, f_{x_2})^T$ of f). The gradient provides an affine constraint on the velocity space, and is sometimes referred to as "motion constraint vector".

Although widely used, this principle is not satisfied in several real situations which include: changing luminance conditions, specularities, multiple motions and in the case of interest for this paper: *transparency*. Many of the above situations can be handled by solving (1) or (2) in a least-square or more robust way, some will use higher order invariants, but transparency can generally not be handled by such a simple modeling, since large areas of the image plane have more than one motion attached to them.

Transparency can be modeled as "superposition", linear or not, of moving layers, corresponding to pixelwise operations on layer intensities. Given n layers $l_i(\mathbf{x},t)$, a transparent image can be modeled as a combination of them,

$$f(\mathbf{x},t)=\sigma(\mathbf{x},t,l_1(\mathbf{x},t),\dots,l_n(\mathbf{x},t)) \tag{3}$$

where σ is a combination operation and each layer motion is described by a motion field $\mathbf{u}_i=(u_i^1,u_i^2)$. Given a sequence $f(\mathbf{x},t)$ obtained by combination of dynamic layers, the following questions are somewhat natural:

- How many layers are observed in the image, i.e. what is the value of n?
- Can we recover/estimate these layers?
- Can we recover/estimate their motions?

The purpose of this work is to address slightly different points, avoiding formulation directly targeted to layer recovery:

- How many motions are present in the image at each pixel?
- Can we estimate these motions?

The first set of questions refers directly to the computation of transparent moving layers while the second one refers to the classical layered motion recovery. Layers recovery requires more than just layered motion recovery: in order to associate the multi-velocity data to layers, the associated model must take into account the layers' spatiotemporal concurrence and provide a mechanism for multi-layer enhancement/attenuation given the observed data and the spatiotemporal coherence, see for instance [1].

The presence of multiple motions prevents the use of simple conservation assumptions: even if the Lambertian assumption holds for each layer, and if we assume that the combination operation σ is independent of image location, i.e. $\sigma(\mathbf{x},t,l_1,\dots,l_n)=\sigma(l_1,\dots,l_n)$, it will generally not hold for their superposition: from $\nabla l_i\cdot\mathbf{u}_i+l_{it}=0$, one gets

$$\nabla f\cdot\mathbf{u}_i+f_t=\sum_{j\neq i}\frac{\partial\sigma}{\partial l_j}\left(\nabla l_j\cdot\mathbf{u}_i+l_{jt}\right) \tag{4}$$

$$+\frac{\partial\sigma}{\partial l_i}\left(\nabla l_i\cdot\mathbf{u}_i+l_{it}\right). \tag{5}$$

The term (5) vanishes by the Lambertian assumption, but the right hand-side of (4) has no reason to. In the general situation, few is known about each layer, σ depends on imaging modalities, and motion recovery becomes a difficult task. It is nevertheless generally assumed that the number of layers/motions is relatively small, $n=2,3$.

A generally used assumption is the one of linear superposition, i.e. (3) becomes

$$f(\mathbf{x},t)=\sum_{i=1}^{n}l_i(\mathbf{x},t) \tag{6}$$

and this is the framework we place ourselves in this paper. Although more complex formation can be considered [2], this is in general a reasonable assumption: the light energy measured by a camera/eye is roughly proportional to the source energy times the reflectance and transmittance factors of the different layers that constitute the medium/scene under observation [3]. Going from this multiplicative model to the additive one (6) corresponds to measure a logarithm of the received energy. Weber's law asserts that for vision, the sensed brightness is indeed approximately logarithmic in the received energy. Many sensors such as CMOS ones behave similarly. When however a layer is fully opaque, the rationale will break down. A multiple motion recovery based on the linear assumption (6) should be able to cope with this situation, by allowing to vary spatially and temporally the number of recovered motion vectors.

In this paper we propose such an approach. It is based on local detectors sensitive to one or more velocities within a *finite velocity space*. We observe that the responses of these detectors usually provide a too local, noisy and somewhat too complex description of the velocities (detecting more velocities than the actually present at a given location). Thus, there is a need for *integration* and *regularization* of this local information, as well as *simplification* when necessary. This has led us to a novel variational approach that integrates and regularizes simultaneous vector motion fields. We show that a simple modification allows the algorithm to resolve Random Dots Kinematogram (RDK) sequences [4], in both single and transparent OFs.

This article, which extends our previous published conference paper [5], is organized as follows. Section 2 reviews the related work on motion recovery in the transparent situation. Section 3 describes the proposed framework based on a *finite sampling of the space of velocities* and states a discrete variational model to handle multiple motions. We introduce our approach which encodes prior knowledge about the OF smoothness and the expected, relatively small, number of motions per pixel (one or two). We show that a modification to our formulation allows to solve RDK sequences which are used in transparent and non-transparent motion perception experiments.The method performance is illustrated in Sect. 4, on synthetic, synthesized realistic and real sequences. Finally, we present our discussion and conclusions in Sect. 5.

2 Related Work

Variational motion recovery was introduced with the work of Horn and Schunck [6]. Motion in the sequence f is recovered by a penalized least squares of the OFC equation (2), by minimizing the quadratic "OFC versus flow field smoothness" functional

$$\int_D \left(\nabla_3 f(\mathbf{x},t)\cdot(u^1,u^2,1)^T\right)^2 + \alpha\left(|\nabla u^1|^2 + |\nabla u^2|^2\right) d\mathbf{x}\, dt.$$

Since this paper was published, a considerable amount of work has been devoted to improving this, via the use of non linear least squares, different smoothness models, incorporation of parametric motion models etc. leading to extremely accurate algorithms.

Variational methods are now considered as some of the best ones for the recovery of single motion, as illustrated by the work of Papenberg et al. [7] as well as Nir et al. [8]. For some review and benchmark, we mention the classical paper of Barron et al. [9] as well as the work of Baker et al. [10] around a database of image sequences with ground truth for a systematic evaluation and benchmark of optical flow algorithms.[1]

Less attention has been devoted to the case of transparency, although it is an active area of research. Starting from the linear superposition model of (6), Bergen et al. in [11] derive an iterative *three frames algorithm for estimating two motions*, by deriving first a *2-fold displaced frame difference equation* using the three frames: assume that f is the sum of two layers $f = l_1 + l_2$, moving respectively with motion $\mathbf{u}_1$ and $\mathbf{u}_2$, and that the Lambertian assumption holds for each layer: $l_i(\mathbf{x},t) - l_i(\mathbf{x}-\mathbf{u}_i, t-1) = 0$. Applying first the DFD (1) for $\mathbf{u}_i$ gives

$$\begin{aligned} f(\mathbf{x},t) - f(\mathbf{x}-\mathbf{u}_i, t-1) &= l_j(\mathbf{x},t) - l_j(\mathbf{x}-\mathbf{u}_i, t-1) \\ &:= d_j(\mathbf{x},t) \end{aligned} \tag{7}$$

with $(i,j) = (1,2)$ or $(i,j) = (2,1)$. The DFD is in general non zero, but one of the layers, l_i, has been eliminated. In case the motion of each layer l_i is constant on at least the three frames t, $t-1$ and $t-2$, the "difference" layer d_j satisfies the DFD $d_j(\mathbf{x},t) - d_j(\mathbf{x}-\mathbf{u}_j, t-1) = 0$. Assuming $\mathbf{u}_i$ is known, $\mathbf{u}_j$ can then be computed by a single estimation technique on d_j.

A theoretical study and extension of this idea is proposed in [12], providing a frequency domain interpretation and explaining a mechanism of "dominant velocity extraction", see also the work of Irani and Peleg in [13, 14]. Shizawa and Mase [15–17] have explored a frequency domain total least-squares formulation for the multiple motion problem. They replace the OFC (2) by the spatiotemporal *linear* homogeneous one

$$\nabla_3 f(\mathbf{x},t)\cdot\mathbf{u} = 0, \qquad \mathbf{u} = (u^1,u^2,u^3)^T \neq 0 \tag{8}$$

or its frequency counterpart $\mathbf{u}^T\omega\hat{f}(\omega) = 0$, where $\omega = (\omega_1,\omega_2,\omega_3)$ are the spatial and temporal frequencies and $\hat{f}$ is the Fourier transform of f. In the case of a unique constant motion, the best $\mathbf{u}$ can be retrieved as the minimizer of the energy

$$E_{single}(\mathbf{u}) = \frac{\mathbf{u}^T\left(\int \omega\omega^T|\hat{f}(\omega)|^2\, d\omega\right)\mathbf{u}}{\mathbf{u}^T\mathbf{u}\int|\hat{f}(\omega)^2|\, d\omega}.$$

This is a total least-square problem whose solution is given by an eigenvector corresponding to the smallest eigenvalue of the 3 × 3 positive semi definite matrix $\mathbf{A} = \int\omega\omega^T|\hat{f}(\omega)|^2\, d\omega$, called (spatio-temporal) structure tensor, from which a velocity is easily deduced (see [18, 19] for instance). For the recovery of n motions, the linear first order constraint (8) is replaced by an n-order one, n-multilinear, obtained by cascading the linear first order ones, the same way the DFD where cascaded for the 3-frames algorithm. For instance, in the two-motions case, the pair $(\vec{\mathbf{u}}_1, \vec{\mathbf{u}}_2)$ would be a zero of the bilinear symmetric map

$$(\mathbf{u_1},\mathbf{u_2}) \mapsto \mathbf{u_2}^T\mathcal{H}\mathbf{u_1} = 0, \tag{9}$$

where $\mathcal{H}$ is the spatiotemporal Hessian operator. Using total least-square formulation, a closed-form solution can be established when $n = 2$, but becomes rapidly more complicated for higher orders.

Mota et al. have extended these ideas in [20] and Mühlich and Aach have proposed an algebraic framework based on homogeneous components of symmetric algebras on the velocity space and its dual in [21]. The very algebraic structure of the motion constraints in frequency domain led Vernon to propose in [22] an algorithm for the decoupling of moving pattern, for both transparency and occlusion models. Using a similar modeling, Zhou and Kambhamettu have proposed a specific algorithm for the specific problem of reflections in [23].

The non homogeneous form of the constraint (9) (with $u_i^3 = 1$) provides the *2-Fold Optical Flow Constraint* equation, as introduced by Shizawa and Mase in [16]:

$$\left(u_1^1\partial_{x_1} + u_1^2\partial_{x_2} + \partial_t\right)\left(u_2^1\partial_{x_1} + u_2^2\partial_{x_2} + \partial_t\right) f(\mathbf{x},t) = 0 \tag{10}$$

(where a product of differential operators means composition of them, $\partial_x\partial_y = \partial_{xy}$). This form is used by Liu et al. in [24] with Hermite polynomial based differentiation filters, they also propose a method for detecting the presence of either single or multiple motions. Darrell and Simoncelli [25]

[1] At http://vision/middelbury.edu/flow/eval.

dualize this constraint in order to construct some Fourier "donuts" that respond to one or more velocities.

The non linear form of the constraint, as used by Burt et al. in their 3-frames algorithm mentioned above, provides what one may call the *2-Fold Displaced Frame Difference* equation:

$$f(\mathbf{x},t) - f(\mathbf{x}-\mathbf{u}_1, t-1) - f(\mathbf{x}-\mathbf{u}_2, t-1) + f(\mathbf{x}-\mathbf{u}_1-\mathbf{u}_2, t-2) = 0. \tag{11}$$

It can and has been extended to more than 2 motion, and has been used as a starting point by several authors. For instance, Stuke et al. use it in order to derive a block-matching approach in [26]. In order to promote spatially smooth solutions, they add a Markov Random Field spatial regularization framework to it in [27]. However, the use of a field of binary indicator variables results in a *computationally intensive minimization* for finding the global solution of the energy minimization method. Similarly, Auvray et al. [28] use a parametric variant of the above block matching and apply it for transparent X-Ray sequences (images where the integration of material density produces transparent sequences [29]). The method segments and estimates the OF by alternately applying IRLS and ICM methods in order to compute both velocities and discrete layer-indicator variables, respectively. Starting with (11), Pingault et al. perform in [30] a N-order Taylor expansion around velocity values. A multiresolution non linear least-squares estimation is performed, using a Levenberg-Marquardt algorithm.

All the above mentioned approaches are based on a single higher-order constraint designed to "react" to multiple local motions. Another important class of approaches uses the dual paradigm of several order-one motion constraints originally designed for detecting single motion. They often use the idea of local dominance of one layer in some spatiotemporal neighborhood of the image sequence. These dominances are scattered in the image plane, and associated with different layers at different positions. In the robust statistics approach of Black and Anandan [31], the image plane is assumed to be partitioned into regions, each one corresponding to a parametric motion model $\mathbf{u} = \mathbf{u}(a)$, a being a low dimensional parameter for this motion model. The motion parameters are then assumed to represent the motion of two layers that cover the entire image plane. The layers are recovered by an iterative parameter/region estimation and by a nulling process. Given an "initial" region $\mathcal{O}_i$, a dominant motion $\mathbf{u} = \mathbf{u}(a_i)$ is estimated together with its inlier pixel region $\mathcal{R}_i \subset \mathcal{O}_i$, then the corresponding outlier region $\mathcal{O}_{i+1} = \mathcal{O}_i \backslash \mathcal{R}_i$, and after n iterations, the decomposition

$$\mathcal{R}_1 \cup \mathcal{R}_2 \cup \cdots \cup \mathcal{R}_n \cup \mathcal{O}_{n+1},$$

where $\mathcal{R}_i$ moves with velocity $\mathbf{u}(a_i)$ and $\mathcal{O}_{n+1}$ is the final outlier region. This strategy is then used in their article with $n = 2$.

Mixture models for multiple motions have been introduced by Jepson and Black [32]. The authors assume that the motion in layers can be explained by up to N constant parametric motion fields and a mixture model. The layer selection probability and layer motion parameters are estimated over image patches and computed by an EM-like algorithm, unfortunately, in the general case it is not easy to determine the size and position of the patches. Ju et al. [33] propose a model in which, multi-layered affine models are defined on small rectangular image patches (bones), and an inter-patch term (skin) introduces a regularization effect in the model parameters estimation. Then layer ownerships and affine model parameters are computed within a robust estimation framework by using an EM algorithm augmented with a spatial regularization. Black et al. [34] compute a set of membership weights in order to link layers with regions. Although the method captures the changes in illumination, it does not allow one to compute the OF of moving transparencies. In an interesting approach [35], the translation of obstructed pixels can be estimated by a learning process of flexible sprites, however the transparency case is not addressed. Weiss and Adelson [36] as well as Rivera et al. [37] have proposed EM-based approaches for computing different layered motion models. They use as prior knowledge the smoothness property of the velocities. The solution is given by a field of layer probabilities. Both methods produce pixelwise unimodal solutions (single motions) since they use a distance measure for single motions as well as entropy controls. For more details about the different types of constraints and proposed approaches, we refer the reader to [38].

Finally, there is a family of approaches which use the "*single-then-transparent velocity extraction*" strategy in order to estimate transparent OF [39, 40]. The first step of this method is to estimate an affine approximation of a single motion by processing several frames of the sequence. Secondly, by nulling the first motion from the sequence it is possible to estimate the second velocity (in [39] an iterative framework is used for refining the solution at the cost of assuming *invertible* global parametric motions). This strategy has two drawbacks: (a) the analyzed sequence must be composed of a single dominant motion and a non-dominant one (so that a good-enough approximation of the dominant one can be estimated by standard parametric OF techniques), and (b) the practical restriction to recover two layers (for instance, the *min/max alternation* algorithm in [39]) in combination with the assumption of global parametric motions, restrict those methods to the recovery of only two different velocities for the whole sequence. The former limitation impedes to solve sequences where there is not a dominant velocity, as for instance our sequence in Fig. 7. The latter limitation is significant in cases involving standard transparencies between two objects (each pixel contains at most two velocities), but the sequence contains more than two velocities (due to the fact that there are more than two objects or due to complex non-rigid motions), as for instance our sequences in Figs. 9 and 11. In Sect. 4.4 (Fig. 10) we test the above limitations by means of experiments.

To conclude this section, we discuss briefly how multiple motion recovery can be used to image layer recovery. Once the transparent OF is computed, it is still a challenging task to recover the original image layers. In the two layers situation, the use of (7) allows, theoretically, to recover partial information about the individual layers. Because only temporal layer difference is provided, the problem is ill posed and error in motion estimation and noise make it complex even in the two layers situation, while the three or more layers case becomes extremely arduous. Toro et al. have proposed an inverse problem regularization approach in [1]. Other authors use specific motion behavior information, as Sarel and Irani in [41], or assume constant ego-motion as Oo et al. in [42].

3 From Local to Global by a Variational Model

We introduce in this section our 2-steps approach for transparent motion estimation in this section: first the gathering of local information, and then its integration in order to provide some global one, via a discrete variational model.

For the finite sampling of the velocity space, we consider N vectors

$$\{\mathbf{u}_1, \dots, \mathbf{u}_N\}, \quad \text{with } \mathbf{u}_i = (u_i^1, u_i^2)^T, \tag{12}$$

describing the set of possible velocities. Our goal will be to determine the likelihood of having the velocity $\mathbf{u}_i$ at a given position. To do so, we use an initial *local* estimate of this likelihood. This likelihood is encoded via the distance function $d(\mathbf{u}_i, \mathbf{r}) \in \mathbb{R}^+|_{i=1,\dots,N}$ which describes at each spatiotemporal position $\mathbf{r} = (\mathbf{x}, t)$ whether the velocity $\mathbf{u}_i$ can locally explain the apparent motion (characterized by $d(\mathbf{u}_i, \mathbf{r}) \approx 0$) or not (characterized $d(\mathbf{u}_i, \mathbf{r}) \gg 0$). Such a motion probe d is defined in the next subsection.

3.1 How to Estimate Local Velocity Information?

The two following similarity operators are, certainly, the most used in motion recovery algorithms: the displaced frame difference, correlation based and non linear,

$$M_{\mathcal{C}}^{(1)}(\mathbf{u}_i) f(\mathbf{x}, t) \overset{\text{def}}{=} f(\mathbf{x}, t) - f(\mathbf{x} - \mathbf{u}_i, t-1); \tag{13}$$

and its linearized version, differential-based,

$$M_{\mathcal{D}}^{(1)}(\mathbf{u}_i) f(x_1, x_2, t) \overset{\text{def}}{=} \left(u_i^1 \frac{\partial}{\partial x_1} + u_i^2 \frac{\partial}{\partial x_2} + \frac{\partial}{\partial t} \right) f(x_1, x_2, t),$$

which gives rise to the well known OF constraint (2). The superscript $^{(1)}$ indicates that these operators are designed for probing one motion vector. Following Shizawa and Mase [17], one can define an operator for two velocities as the composition of single velocity operators (here the superscript $^{(2)}$ indicates 2 motions):

$$M_{\mathcal{D}}^{(2)}(\mathbf{u}_i, \mathbf{u}_j) f(\mathbf{x}, t) \overset{\text{def}}{=} M_{\mathcal{D}}^{(1)}(\mathbf{u}_i) M_{\mathcal{D}}^{(1)}(\mathbf{u}_j) f(\mathbf{x}, t) \tag{14}$$

where products $\frac{\partial}{\partial r}\frac{\partial}{\partial s}$ are expanded as $\frac{\partial^2}{\partial r \partial s}$, see [25]. Composing instead the non linear correlation operators $M_{\mathcal{C}}^{(1)}$ provides the non linear operator for two velocities—the 2-fold displaced frame difference which corresponds to the distance (11) reported in [26] as

$$\begin{aligned} M_{\mathcal{C}}^{(2)}&(\mathbf{u}_i, \mathbf{u}_j) f(\mathbf{x}, t) \\ &\overset{\text{def}}{=} f(\mathbf{x}, t) - f(\mathbf{x} - \mathbf{u}_i, t-1) \\ &\quad - f(\mathbf{x} - \mathbf{u}_j, t-1) + f(\mathbf{x} - \mathbf{u}_i - \mathbf{u}_j, t-2). \end{aligned} \tag{15}$$

We introduce here the general mechanism we have used in order to select the local velocity descriptors $d(\mathbf{u}_i, \mathbf{r})$ from our motion operators. We have a set of "velocities probe" operators

$$\mathbf{M} = \Big\{ M^{(1)}(\mathbf{u}_1), \dots, M^{(1)}(\mathbf{u}_N), \\ M^{(2)}(\mathbf{u}_1, \mathbf{u}_2), M^{(2)}(\mathbf{u}_1, \mathbf{u}_3), \dots, M^{(2)}(\mathbf{u}_{N-1}, \mathbf{u}_N) \Big\}.$$

Let $\mathbf{M}_{\mathbf{u}_i} \subset \mathbf{M}$ be the subset of all the operators $M^{(1)}$ and $M^{(2)}$ involving $\mathbf{u}_i$, then we define

$$d_{\mathcal{C}}(\mathbf{u}_i, \mathbf{r}) = \min_{M_{\mathcal{C}}^{(k)} \in \mathbf{M}_{\mathbf{u}_i}} \frac{1}{k} \sum_{\mathbf{s} \in W_r} \left(M_{\mathcal{C}}^{(k)} f(\mathbf{s}) \right)^2 \tag{16}$$

and

$$d_{\mathcal{D}}(\mathbf{u}_i, \mathbf{r}) = \min_{M_{\mathcal{D}}^{(k)} \in \mathbf{M}_{\mathbf{u}_i}} \frac{1}{k} \sum_{\mathbf{s} \in W_r} \left(M_{\mathcal{D}}^{(k)} f(\mathbf{s}) \right)^2 \tag{17}$$

when $\mathbf{M}$ is composed of operators $\{M_{\mathcal{C}}^{(1)}, M_{\mathcal{C}}^{(2)}\}$ or $\{M_{\mathcal{D}}^{(1)}, M_{\mathcal{D}}^{(2)}\}$ respectively. W_r is a $n \times n$ spatial window center at r, we use $n = 3$ in all our experiments. As we will see in the next section, we can indistinctly plug-in one or the other distance in our generic-application regularization framework. Note that because $M_{\mathcal{D}}^{(1)}$ is the series Taylor's approximation of $M_{\mathcal{C}}^{(1)}$, $d_{\mathcal{C}}$ is more suitable for long displacements.

Last but not least, there is a problematic situation in which the two-displacement operator $M^{(2)}$ may incorrectly detect two motions instead of one within a region

$\mathcal{R}$ which is moving with a single-displacement $\vec{u}_i$: for any $\boldsymbol{\delta} = \mathbf{u}_1, \dots, \mathbf{u}_N$, such that the point $\mathbf{x}' = \mathbf{x} - \boldsymbol{\delta} \in \mathcal{R}$, we detect both $\mathbf{u}_i$ and $\boldsymbol{\delta}$ due to

$$\begin{aligned} &M_{\mathcal{C}}^{(2)}(\mathbf{u}_i, \boldsymbol{\delta}) f(\mathbf{x}, t) \\ &\quad = \underbrace{f(\mathbf{x}, t) - f(\mathbf{x} - \mathbf{u}_i, t-1)}_{M_{\mathcal{C}}^{(1)}(\mathbf{u}_i) f(\mathbf{x},t) = 0} \\ &\qquad - \underbrace{[f(\mathbf{x} - \boldsymbol{\delta}, t-1) - f(\mathbf{x} - \boldsymbol{\delta} - \mathbf{u}_i, t-2)]}_{M_{\mathcal{C}}^{(1)}(\mathbf{u}_i) f(\mathbf{x} - \delta, t-1) = M_{\mathcal{C}}^{(1)}(\mathbf{u}_i) f(\mathbf{x}', t-1) = 0} = 0, \end{aligned}$$

because the velocity $\mathbf{u}_i$ is also present at coordinate $(\mathbf{x}', t-1)$. To prevent this ambiguity, we use the value of operators $M^{(2)}$ only when

$$M^{(2)}(\mathbf{u}_i, \cdot) f(\mathbf{x}, t) < \min \left[M^{(1)}(\mathbf{u}_j) f(\mathbf{x}, t) \right] \quad \forall \mathbf{u}_j,$$

i.e., when the velocity probe $M^{(2)}$ is explicitly *better than* (and not *equal to*) all the $M^{(1)}$ probes.

3.2 Global Motion Integration via a Variational Approach

Once we have gathered local motion estimates, we need to integrate them in order to obtain a global and robust velocity estimation. Because more motions than the actually present may be detected, and this is due to flat intensity regions or the *aperture problem*, we need to simplify the information, while regularizing it spatially in order to get rid of noise.

Let us define the function $\alpha_i(\mathbf{r})$, which corresponds to the probability that velocity $\mathbf{u}_i$ explains the apparent motion at the spatiotemporal position $\mathbf{r}$. We define the unknowns of the problem as the vector valued field α: $\alpha(\mathbf{r}) = [\alpha_1(\mathbf{r}), \dots, \alpha_N(\mathbf{r})]^T$, with $\alpha_i(\mathbf{r}) \in [0, 1]$ $\forall r \in \Omega \times \{0, \dots, T\}$.

Note that, although each component $\alpha_i(\mathbf{r})$ can be interpreted as a probability, $\alpha(\mathbf{r})$ is not a probability measure (as in [36, 37]) in the sense that the sum of its components is not constrained to be equal to one. If two motions $\mathbf{u}_i$ and $\mathbf{u}_j$ are present at a particular pixel position, then we expect that $\alpha_i(\mathbf{r}) \approx \alpha_j(\mathbf{r}) \approx 1$. Conversely, the velocity(ies) at a position $\mathbf{r}$ can be extracted from $\alpha(\mathbf{r})$ by selecting the velocity(ies) $\mathbf{u}_i$ with highest $\alpha_i(\mathbf{r})$ value(s).

Then in order to estimate the *global* multi-valued velocity field from the *local* data $d(\mathbf{u}_i, \mathbf{r})$, we compute the minimizer of the objective function $\mathcal{E}(\alpha)$ defined by

$$\begin{aligned} \mathcal{E}(\alpha) = \sum_r \Bigg\{ &\sum_i d(\mathbf{u}_i, \mathbf{r}) \alpha_i^2(\mathbf{r}) + \lambda_a (1 - \alpha_i(\mathbf{r}))^2 && (18) \\ &+ \frac{\lambda_s}{2} \sum_{\mathbf{s}: s \in \mathcal{N}_r} \sum_i w_i(\mathbf{r}, \mathbf{s}) [\alpha_i(\mathbf{r}) - \alpha_i(\mathbf{s})]^2 && (19) \\ &+ \lambda_c \left[\kappa N \bar{\alpha}^2(\mathbf{r}) - \sum_i \alpha_i^2(\mathbf{r}) \right] \Bigg\}, && (20) \end{aligned}$$

subject to the constraints $\alpha_i(\mathbf{r}) \in [0, 1]$ for all i; with $\bar{\alpha}(\mathbf{r}) \overset{\text{def}}{=} \frac{1}{N} \sum_i \alpha_i(\mathbf{r})$, $w_i(\mathbf{r}, \mathbf{s})$ are anisotropic diffusion weights defined in the sequel, $\mathcal{N}_\mathbf{r} \overset{\text{def}}{=} \{\mathbf{s} : \mathbf{r}, \mathbf{s} \in \Omega \times [0, T], \|\mathbf{r} - \mathbf{s}\| < 2\}$ is the spatiotemporal neighborhood of the $\mathbf{r}$ position, and $\kappa, \lambda_a, \lambda_s, \lambda_c$ are user-defined positive constants. We will in the sequel, denote by $U_c(\alpha, \mathbf{r})$ the term between the square brackets in formula (20). Let us now comment the different terms of this energy.

3.2.1 *Attach Potential in Term* (18)

This term links the input (the functions d's) to the unknown α. For computing the presence of the i-th model, we use an approach related with the outlier rejection method [43] and with the EM formulation [33, 34, 36, 37]. Minimizing term in (18) w.r.t. $\alpha_i(\mathbf{r})$ produces $\alpha_i(\mathbf{r})$ close to 0 for high $d(\mathbf{u}_i, \mathbf{r})$ values, indicating in this way that such a motion model is not likely at position $\mathbf{r}$. Furthermore, the second quadratic term in (18) avoids the null trivial solution $\alpha(\mathbf{r}) = 0$ by pushing the $\alpha_i(\mathbf{r})$ towards the value 1.

3.2.2 *Spatial Regularization Potential in Term* (19)

Local information is integrated through this regularization term. At a given location $\mathbf{r}$, we minimize the difference between vector $\alpha(\mathbf{r})$ and all the vectors $\alpha(\mathbf{s})$ in its neighborhood, $\mathcal{N}_\mathbf{r}$. Because our indicator variables are real valued, we can use differentiable potentials with their well-known algorithmic advantages. The smoothing process is controlled by directional fixed weights [44], in this case we use:

$$w_i(\mathbf{r}, \mathbf{s}) = \left((\mathbf{s} - r)^T \, \bar{\mathbf{I}}_i \, (\mathbf{s} - r) \right) / \|\mathbf{s} - r\|^4,$$

generated from the i-th tensor associated to the i-th velocity model: $\bar{\mathbf{I}}_i = \gamma \mathbf{I}_d + \mathbf{U}_i \mathbf{U}_i{}^T$, where $\mathbf{I}_d$ is the identity matrix, $\gamma = 0.1$ and $\mathbf{U}_i = [u_i^1, u_i^2, 1]^T / \|[u_i^2, u_i^2, 1]\|$ is a homogeneous-coordinate unit vector. For small γ values (as the one proposed here) these weights promote a strong smoothness along the i-th velocity direction. The empirically-fixed 4-th power of the distance restricts the spatial influence of the smoothness term, see also [38]. As a consequence piece-wise smooth OFs are recovered with defined boundaries along the velocity model (see Fig. 6).

3.2.3 *Inter-Model Competition Potential in Term* (20)

Our aim is to detect multiple simultaneous motions (transparent motions), thus we may have problems at sites where multiple spurious matches are locally detected, for example in homogeneous regions, where $d(\mathbf{u}_i, \mathbf{r}) \approx 0$ for many (maybe all) velocities. For this reason we need a mechanism to eliminate spurious models (i.e. to cancel some of the components of $\alpha(\mathbf{r})$) and promote the valid ones: we would like to recover almost-binary solutions.

Regularization terms for this aim have been proposed in [36, 37, 45]. We improve those proposals by performing an *inter-model* regularization over the $\alpha(\mathbf{r})$ vector by means of the quadratic term defined in (20). For avoiding a disruption in the method presentation, the complete mathematical deduction of this regularization term is presented in Appendix. Here we continue with our discursive presentation.

We can tune the κ parameter in (20) depending on the number of models that we want to detect at each pixel. Figure 1 shows the regularization behavior of this term by itself along the iterations for $\kappa = 1$, 2 and 4 (we used $N = 16$ velocity models). Note that only the prominent coefficients (those which have significant values since the first iterations) are enhanced and the others are completely attenuated. As can be seen, potential (20) is suitable for recovering multimodal solutions and effectively controls the number of models with the parameter κ. In fact, potential (20) can be tuned so that for a given κ value, a multimodal solution (with two or more detected motions) has lowest cost for a given number of modes: for instance, the final solution (in the convergence) shown in the bottom of Fig. 1 has the cost shown in first row of Table 1, then, by arbitrarily turning-off some α_i coefficient or, conversely by arbitrarily turning-on a third coefficient (second and third rows in Table 1, respectively) the resultant energy is increased. Such a behavior is plotted in Fig. 2 for several κ values and for a different number of modes in the vector solution. This behavior makes an important distinction with respect to Shannon's entropy, which always present lower energy for unimodal solutions [36, 37]. Additionally, our proposed potential, based on quadratic terms, is easily differentiable and therefore we can use simple minimization algorithms.

Because of the negative terms in (20), the cost function $\mathcal{E}(\alpha)$ may become non-convex when $\lambda_c \gg \lambda_s$. Figure 3 illustrates this for a simple 1D example (i.e. for a single row image): we place ourselves at a given location r, with $N = 2$ velocity models and corresponding measured distances $d(\mathbf{u}_1, r) \gg 0$ and $d(\mathbf{u}_2, r) \approx 0$, and since we consider a single row $\mathcal{N}_r = \{r-1, r+1\}$. For illustrative aims, we fixed the neighbor α-vectors as $\alpha(r-1) = [0.2, 0.75]^T$, $\alpha(r+1) = [0.15, 0.85]^T$. Clearly, given the local distances $d(\mathbf{u}_i, r)$ and the two α-vectors in $\mathcal{N}_r$, model $\mathbf{u}_2$ is the most plausible one. Figure 3(a) shows the contour lines of functional with the inter-model regularization term turned-off, i.e. $\lambda_c = 0$. Then, by increasing the λ_c value in Fig. 3(b), we lead the solution to the desired one (almost binary). In the same way, an excess in the value of λ_c parameter generates a non-convex cost function with several local minima, as is illustrated in Fig. 3(c). However, in the region close to the global solution the gradient of the functional correctly indicates the desired minimization direction. For this reason, we propose to gradually introduce the λ_c value, due to it is

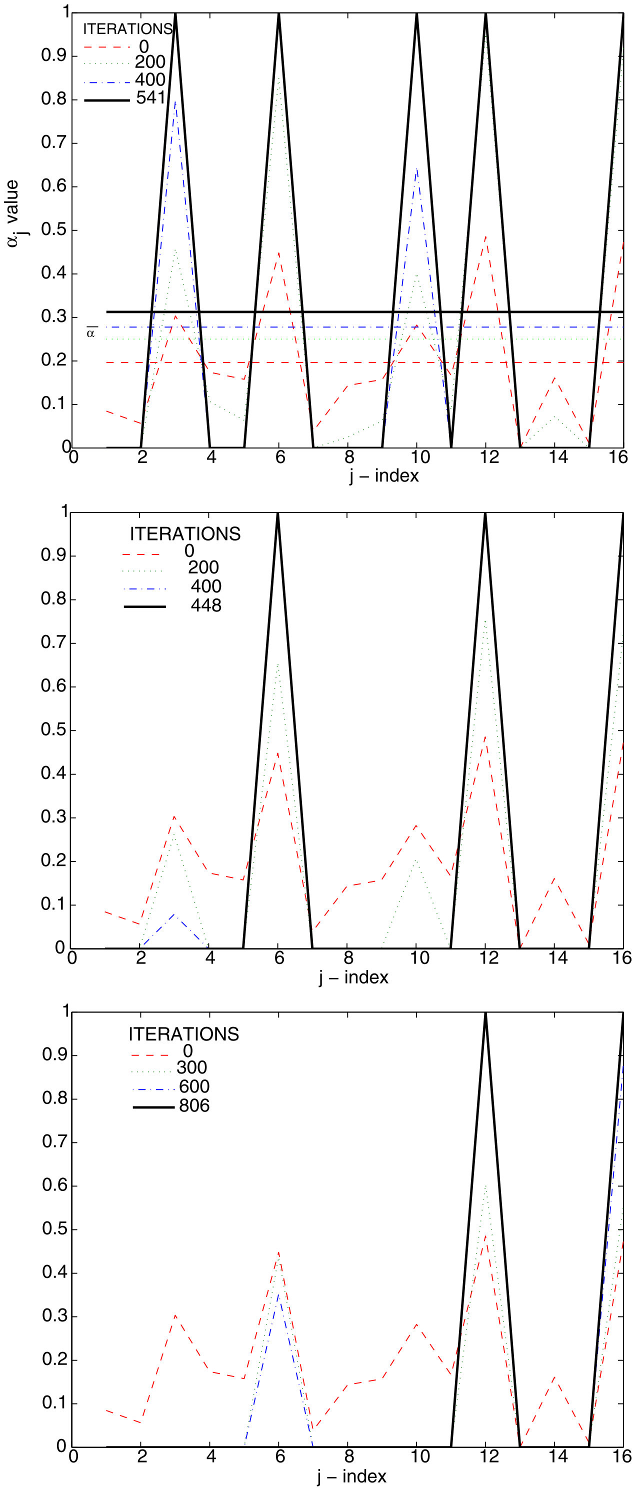


Fig. 1 Regularization effect along the iterations of the inter-model competition term (20). At iteration zero several models have significant values; at minimization convergence the non-prominent coefficients are completely attenuated. Result for $\kappa = 1$ in the top, i.e. term (23), where the *horizontal line* marks the mean vector value $\bar{\alpha}$, for $\kappa = 2$ in the *middle* and for $\kappa = 4$ in the *bottom*

Table 1 Resultant cost of $U_c(\alpha, \mathbf{r})$ in (20) by arbitrarily turning off/on some α_i coefficients from the optimal solution shown in the bottom of Fig. 1

Cost $U_c(\alpha, \mathbf{r})$ (20)	Turned-on coefficients (all others are zero)	Comments
-1.00	$\alpha_{12} = \alpha_{16} = 1$	Optimal
-0.75	$\alpha_{16} = 1$	Non-optimal
-0.75	$\alpha_6 = \alpha_{12} = \alpha_{16} = 1$	Non-optimal

important to perform the coefficient-contrast regularization until we have an intermediate regularized solution [as for instance the one presented in Fig. 3(a)]. Thus, as is detailed in Sect. 3.3, we perform a deterministic annealing over the parameter λ_c along the minimization.

Finally, by using the contrast term (20) it is possible to avoid the second term in (18) that promotes the α "switching-on", see Appendix. However, as such a scheme is difficult to tune we prefer to control the different tasks in independent terms.

3.3 Implementation

Without the constraint $\alpha_i(\mathbf{r}) \in [0, 1]$, in the convex case, the minimization would be performed by solving $\nabla_\alpha \mathcal{E} = 0$. Using a Gauss-Seidel solver, a relaxation step for a given $\alpha_i(r)$ would be

$$\alpha_i(\mathbf{r}) \leftarrow \frac{\lambda_s \sum_{s \in \mathcal{N}_r} w_i(\mathbf{r}, \mathbf{s})\alpha_i(\mathbf{s}) - \lambda_c \frac{\kappa}{N} \sum_{k \neq i} \alpha_k(\mathbf{r}) + \lambda_a}{d(\mathbf{u}_i, \mathbf{r}) + \lambda_s \sum_{s \in \mathcal{N}_r} w_i(\mathbf{r}, \mathbf{s}) + \lambda_c(\kappa - 1) + \lambda_a}.$$

Although the data term in our cost function tends to encourage the $\alpha_i(\mathbf{r})$'s to be close to $[0, 1]$, these constraints are not strictly enforced, and we cannot expect in general that the global minimum would fulfill them, i.e. we cannot expect that our optimal solution $\tilde{\alpha}$ satisfies $\nabla_\alpha \mathcal{E}(\tilde{\alpha}) = 0$. Nevertheless, the above relaxation step is energy decreasing, when the cost function is convex in the neighborhood of the current α estimate. We may thus implement our minimization by interleaving these relaxations steps and projecting the $\alpha_i(\mathbf{r})$'s back in the constraint intervals. We found in fact, that in order to obtain a smooth algorithm convergence, it is important to keep fixed the mean $\bar{\alpha}(\mathbf{r})$ while updating the entries of the current $\alpha(\mathbf{r})$ vector. This leads to the following procedure: our sweeps are in pixel lexicographic order, with a given fixed order on the $\alpha_i(\mathbf{r})$, and at a given position $\mathbf{r}$, we update each $\alpha_i(\mathbf{r})$ via

$$\alpha_i(\mathbf{r}) \leftarrow \frac{\lambda_s \sum_{s \in \mathcal{N}_r} w_i(\mathbf{r}, \mathbf{s})\alpha_i(\mathbf{s}) - \kappa\lambda_c\bar{\alpha}(\mathbf{r}) + \lambda_a}{d(\mathbf{u}_i, \mathbf{r}) + \lambda_s \sum_{s \in \mathcal{N}_r} w_i(\mathbf{r}, \mathbf{s}) - \lambda_c + \lambda_a}$$

while projecting back out-of-bounds $\alpha_i(\mathbf{r})$-values to the constraint range $[0, 1]$ after each computation. When we are done at a given location $\mathbf{r}$, then we update the mean value $\bar{\alpha}(\mathbf{r})$. We initially set all the $\alpha_i(\mathbf{r})$'s to the value 0.5, and as mentioned in the Sect. 3.2.3, we perform a deterministic annealing on λ_c. The annealing scheme is the following. For each iteration $k = 1, 2, \ldots, n$, we set

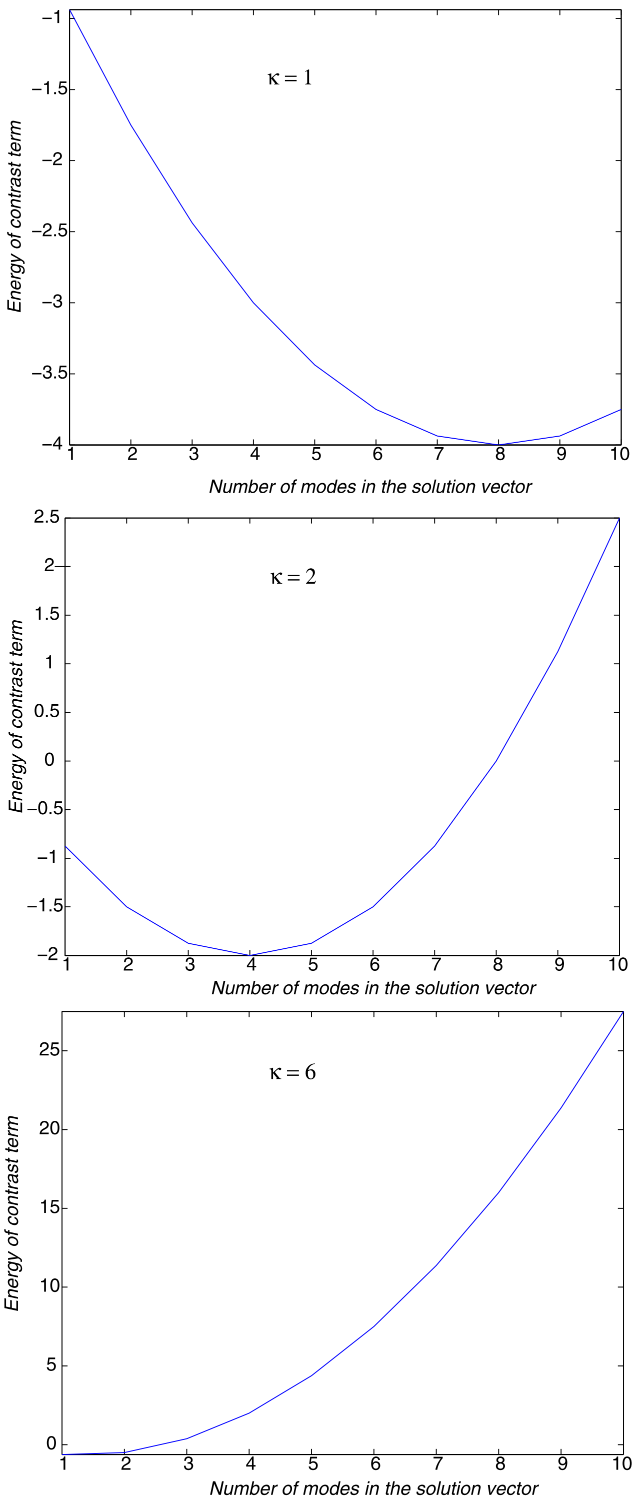


Fig. 2 Number of modes (i.e. $\alpha_i = 1$) vs. the cost of term (20) with $N = 16$ and different κ values. The number of modes that minimize the potential (20) is controlled by parameter κ

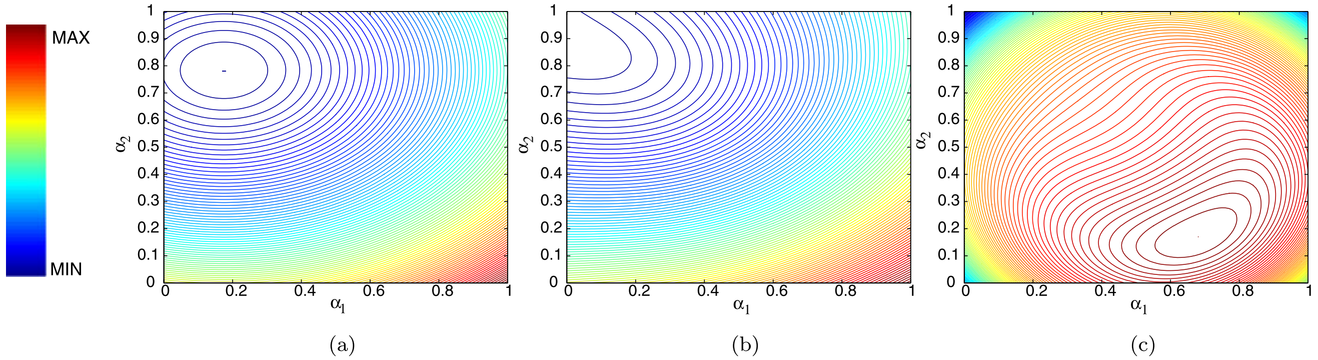


Fig. 3 Convex or non-convex regularized cost function (18)–(20) depending on the parameter selection. Contour lines of the functional $\mathcal{E}(\alpha)$ for parameters $[\lambda_s, \lambda_c]$ equal to (**a**) [1, 0], (**b**) [1, 1] and (**c**) [1, 20], respectively

$\lambda_c^{(k)} = \lambda_c a_k$, where λ_c is the chosen contrast level and $a_k = 1 - 0.95^{(100k/n)}$ is a factor that increases to 1 in approximately 90% of the total iterations. Such a deterministic annealing is inspired in the work of Blake and Zisserman [46], where *graduated non-convexity* was used for approximating the global solution for non convex minimization. The results are sensitive to the annealing speed of λ_c: premature increment results in an early convergence, thus lead us to an incorrect solution. Nevertheless, the proposed schedule seems work well in all the evaluated cases and thus we used the same annealing scheduling in all our experiments.

For the sake of reproducibility and to avoid dependence on dynamic range of input sequences, we normalize the input distances $d(\mathbf{u}_i, \mathbf{r}) \in [0, 1]$, thus we can report a range for the user-defined parameters. A large value for λ_s eliminates noise but a too large ones blurs the motion boundaries. We used $\lambda_s \in [0.1, 15]$ for an adequate noise reduction. In our experiments, parameter $\kappa = 1$ performs well for most noise-free synthetic sequences. For noise-contaminated, real sequences (with homogeneous regions) or when the number of base velocities are increased (then several spurious models may be present) the prominent models are obtained by increasing this parameter, $\kappa \in [1, 4]$. Note that the κ value depends on the size N of the velocity dictionary in (12). Thus, in order to obtain an approximate value, the user should use a procedure similar to the one we depict in Fig. 2 with the actual number of velocities. We set parameter $\lambda_a \in [0.001, 0.05]$ and $\lambda_c \in [0.001, 0.05]$ for our experiments. Note also that in all cases, we compute our dense OF using at most 200 iterations.

3.4 Application to Perception Experiments: Random Dot Kinematogram Sequences

Random Dots Kinematogram (RDK) are sequences commonly used in motion perception experiments [4, 47, 48]. RDK sequences are composed of a set of moving randomly distributed dots which are moving with different directions and/or speeds, see Figs. 13(a) and 14(a). The movement of these points in a direction causes the perception of the movement of the entire display. Moreover, if a part of these dots are moving in a direction while all the others are moving in another direction, the human observer has the perception of transparency. A full representation of these visual situations include both local and global motions. These sequences arise from the experiments in [4], that show that representations of multiple velocities do not coexist at the finest spatial scale of motion analysis. According to the experiment in [48], the transparency detection occurs in two early vision stages: (a) local motion detectors which show a winner-take-all interaction and (b) integration of motion signals over a more extended region. Neurophysiologically, there is evidence that such processes (local and integration) are carried out in different networks of the cortical area: neurons V1 encode more local motion information and spatial integrative process seems to be a property of neurons V5, so called MT. Note that from a simulation point of view and computationally speaking, these sequences are very challenging, given that there are not explicit boundaries between regions.

Inspired by the motion perception model reported in [48], we propose to recover in two stages the predominant motions in RDK sequences. We first smooth noisy or sparse motion signals to disambiguate the aperture problem (mimicking V1 processing), then we propagate the information to the entire display in a integration process (mimicking MT processing), as we explain in the following.

In first stage we slightly modify the cost function (18)–(20) in order to deal with the non-textured regions (areas without dots) that must be integrated to the local motions. The adapted cost function is:

$$U_a(\alpha) = \sum_r \left\{ T_r \left[\sum_i d(\mathbf{u}_i, \mathbf{r}) \alpha_i^2(\mathbf{r}) + \lambda_a (1 - \alpha_i(\mathbf{r}))^2 \right] \right.$$
$$\left. + \frac{\lambda_s}{2} \sum_{\mathbf{s}: s \in \mathcal{N}_r} \sum_i w_i(\mathbf{r}, \mathbf{s}) [\alpha_i(\mathbf{r}) - \alpha_i(\mathbf{s})]^2 \right.$$

$$+ T_r \lambda_c \left[\kappa N \bar{\alpha}^2(\mathbf{r}) - \sum_i \alpha_i^2(\mathbf{r}) \right] + \lambda_p \sum_i \|\mathbf{u}_i\| \alpha_i^2(\mathbf{r}) \Bigg\} \quad (21)$$

where the term

$$T_r = |\nabla f_r| / \left(\lambda_g + |\nabla f_r|\right)$$

indicates a confidence coefficient that depends on the local texture (image gradient) and λ_g is a user-defined parameter that controls the contrast on T_r. Regions where the aperture problem can be solved have a large T_r coefficient (this kind of regions include corners, textured ones and borders), while homogeneous regions without texture, where any movement is equal likely to occur have a T_r close to zero. We use this coefficient for detecting regions where the movement information is poor so that we need to acquire the information from its neighborhood. Additionally, we use the last term in (21) that assumes a prior favoring slow velocities, used in [49]. This term, controlled by the parameter λ_p, promotes that the movement [0, 0] is set as the likely one, when there is no other prior knowledge, and allow us to detect motionless non-textured regions.

For solving (21) we use the single-motion distance defined in (13), since for these sequences, representations of multiple velocities do no coexist at this spatial scale (see [4]).

The integration of global motion needs a large interaction area (different scales). In this second stage we use as initial solution the α coefficients obtained in the previous stage (where the local movements have been detected), and then we perform the diffusion-based spatial integration by the minimization of the local energies:

$$U_b(\alpha(\mathbf{r})) = \sum_i^N \Bigg[\sum_{s:s\in\mathcal{N}_r} \alpha_i(\mathbf{s})\,(\alpha_i(\mathbf{r}) - \alpha_i(\mathbf{s}))^2 + \lambda_c \left(\kappa N \bar{\alpha}^2(\mathbf{r}) - \sum_i \alpha_i^2(\mathbf{r}) \right) \Bigg],$$

where the diffusion weights are the $\alpha_i(s)$ coefficients. By using these diffusion weights, we promote that each position r to be similar to the neighbor position s, when position s has a large $\alpha_i(s)$ coefficient, i.e., where the neighbor was marked by the previous stage as a position with a predominant α coefficient (close to one).

For our experiments we use the following user parameters: for the texture detector we set $\lambda_g = 1$ and use a small value $\lambda_p = 0.1$ in order to allow the detection of velocities with norm different to zero.

4 Experiments on Synthetic and Real Sequences

In this section we present results for synthetic and real transparent sequences; all of them can be downloaded at the web site: www-sop.inria.fr/odyssee/data/sequences/.

4.1 Local Measurements Are Noise Sensitive

Figure 4(a) shows a synthetic sequence (size 54 × 54 × 16) with transparent motion, very similar to the one tested in [27]: there is a moving background (with velocity $\hat{\mathbf{u}} = [0, -1]$) and an overlapped moving transparent square (with velocity $\hat{\mathbf{v}} = [1, 0]$), with additive Gaussian noise. Figure 4(b) (resp. 4(c)) show the OFs associated to the minima of local detector distance in (16) (resp. (17)). This represents indeed what will be the input of our approach and illustrate the need for velocity integration even for low levels of noise.

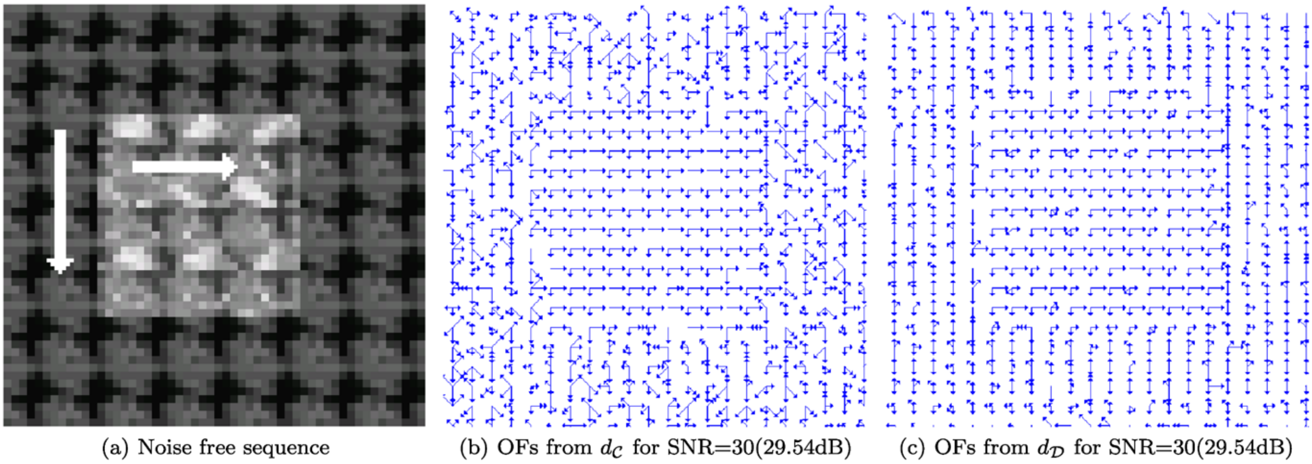

(a) Noise free sequence (b) OFs from $d_{\mathcal{C}}$ for SNR=30(29.54dB) (c) OFs from $d_{\mathcal{D}}$ for SNR=30(29.54dB)

Fig. 4 (**a**) A synthetic test sequence, and results obtained when noise is added and by using only the distances (**b**) $d_{\mathcal{C}}$ in (16) and (**c**) $d_{\mathcal{D}}$ in (17)

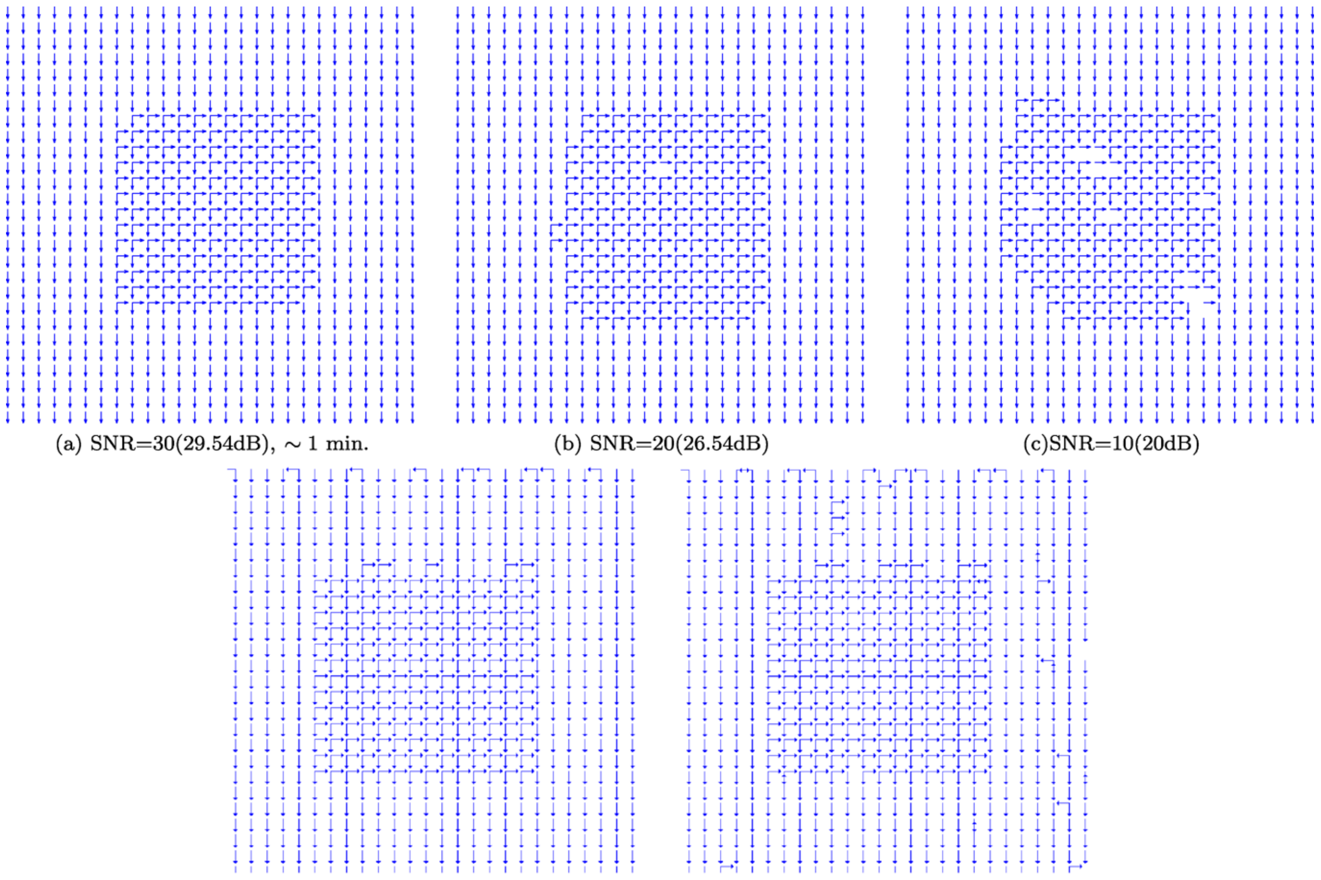
4.2 Regularization of Local Measurements

In order to measure the robustness of our approach, we corrupted the sequence in Fig. 4(a) with different Gaussian white noises, a low noise (SNR = 30), a medium one (SNR = 20) and a large one (SNR = 10). Figures 5(a)–(c) show the obtained results. The percentages of pixels with a wrong estimation are 0.64%, 2.39% and 4.48% respectively: Our approach can deal with a strong noise corruption, providing better results than [27] (compare Figs. 5(d) and 5(e) with results from [27]). The velocity basis was composed of 33 vectors, specified through their magnitudes and orientations, respectively $\{0, 1, 2, 3, 4\}$ pixels and angles of $\{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3}{4}\pi, \pi, \frac{5}{4}\pi, \frac{3}{2}\pi, \frac{7}{4}\pi\}$ radians. For comparison purposes, Figs. 5(d) and 5(e) show the computed OF with the computationally expensive Gibbs sampler which minimizes the discrete cost function proposed by Stuke et al. in [27]. They used a deterministic relaxation ICM algorithm, prone to converge to a local minimum as no annealing strategy is used, and this is the reason they recommend to use instead the Gibbs sampler approach. The noise-free case is shown in Fig. 5(d), and the SNR = 30 (29.54 dB) case in Fig. 5(e). The shown results correspond to the computed solution after 150,000 iterations (about 2.5 hours, in a PC Pentium IV, 3.0 GHz) that is 150 times slower than our approach. For the Gibbs sampler method, the results quality rapidly decreases as the noise level increases: see Fig. 5(e) and compare it with the one computed with the proposed method in about 1 minute in Fig. 5(a).

The behavior of the spatial regularization and inter-model competition is illustrated in Fig. 6. It shows the evolution of the layer associated with velocity $[1, 0]$: as expected, large values appear in the square region whereas small values appear in the background region.

4.3 Realistic Textured Sequences

High textured sequences are relatively easy to solve using local motion measures. In order to evaluate the real performances of the methods, we use more realistic textured scenes with homogeneous regions where many velocities will locally explain the data. It is then necessary to be able


(a) SNR=30(29.54dB), ∼ 1 min.

(b) SNR=20(26.54dB)

(c)SNR=10(20dB)

(d) Gibbs S., noise-free

(e) Gibbs S., SNR=30(29.54dB), ∼ 2.5 hrs.

Fig. 5 *First row.* Results obtained with our approach applied to the synthetic sequence presented in Fig. 4(a) with different noise levels (input was $d_{\mathcal{D}}$). *Second row.* Comparison with Gibbs Sampler scheme (as proposed in [27]): (**d**) noise-free case and (**e**) noisy case SNR = 30 (29.54 dB), confront with our result in (**a**)

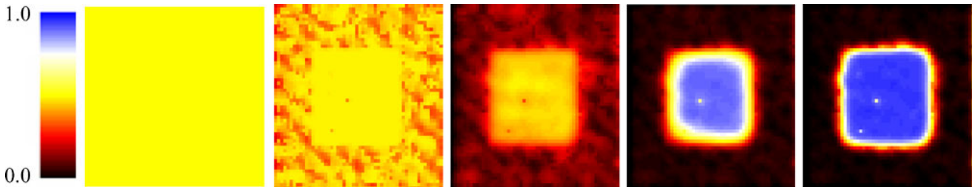
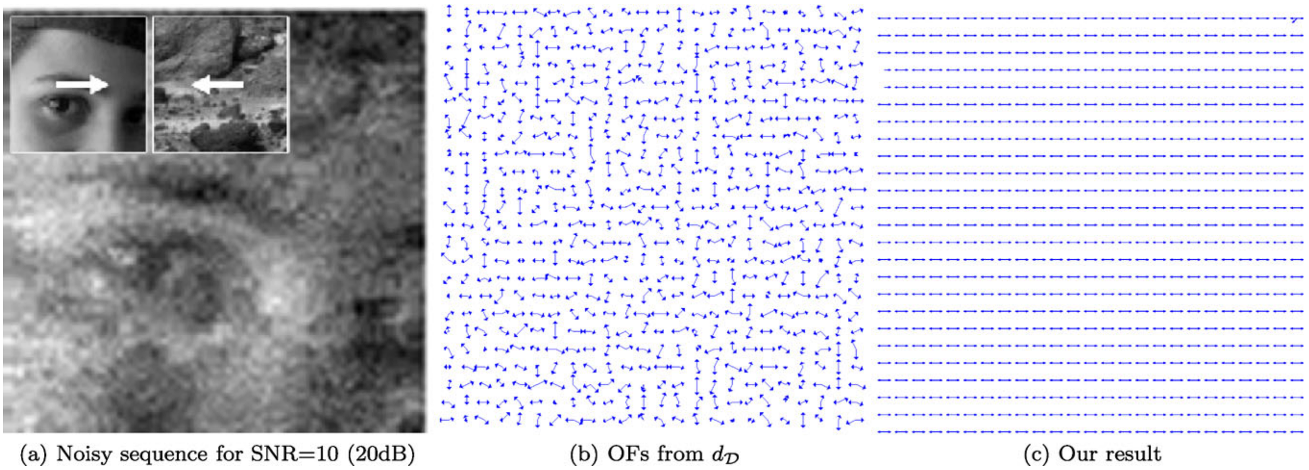


Fig. 6 Evolution during minimization (in pseudo-color scale) of $\alpha_i(\mathbf{r})$ for the layered motion $\mathbf{u}_i = [1, 0]$ for SNR = 20 (26.54 dB). Iterates 0, 1, 11, 31 and 200 are shown



(a) Noisy sequence for SNR=10 (20dB) (b) OFs from $d_{\mathcal{D}}$ (c) Our result

Fig. 7 Two noisy realistic textured patterns in translation. We show the velocities associated to minimum distance in (17) and our regularized result

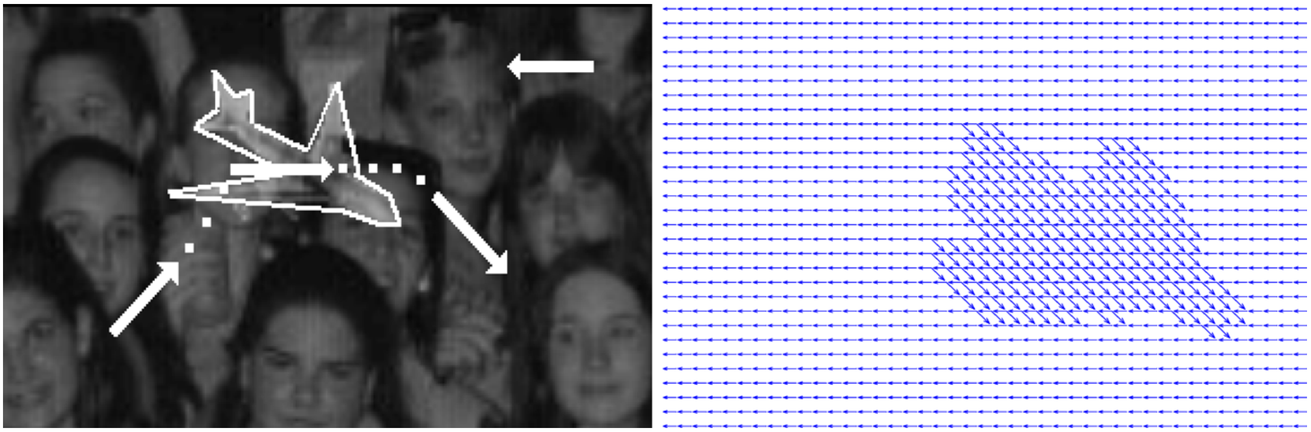


Fig. 8 One frame of a sequence with a transparent object moving with changing translational speed over a translating background and the recovered multi-motion field. Results for more frames of this sequence can be found in www.cimat.mx/~mrivera/vision/transparent_sequences/index.html

to carry the information from less ambiguous regions. The next experiment is designed with that purpose in mind. It is composed of two moving photographs: a face I_1 with motion $\mathbf{u} = [1, 0]$ (limited textured scene) and a rocky Mars landscape I_2, with motion $\mathbf{v} = [-1, 0]$. The sequence was generated with $f = 0.6 I_1 + 0.4 I_2$, see Fig. 7(a). Figure 7(b) shows the OF associated to the minimum distance in (17) used in the attach term (18). The computed velocity field is shown in Fig. 7(c). Note that the right OF is recovered in all the pixels regardless of the high amount of noise.

Figure 8 shows a sequence with a time-varying transparent region and motions. The changing velocities are sketched in Fig. 8(a). An example of the obtained multi-velocity vector field is shown in Fig. 8(b). For this experiment we used the distance measure in (16).

The next sequence, shown in Fig. 9, is artificial too, and made of a rotating earth globe (1 degree per frame) added to a texture image, the latter translating with velocity $[-1, 0]$, and with low-level added noise. Original layers are show in Fig. 9(a). Because of the rotation, a large set of velocities (with different orientations and magnitudes) are present

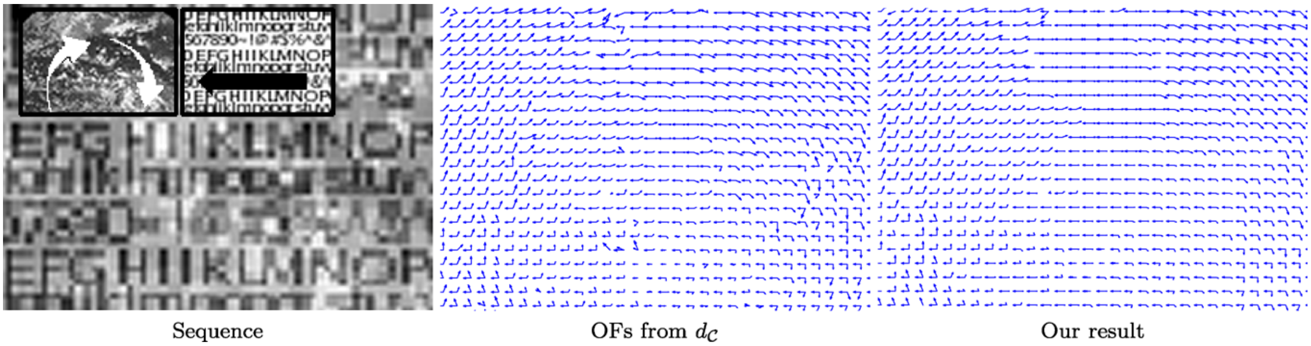


Fig. 9 Transparent motion sequence with complex rotating motion, SNR = 60 (35.56 dB). We show: the sequence, the associated velocities to minimum distance in (16) and our method's result

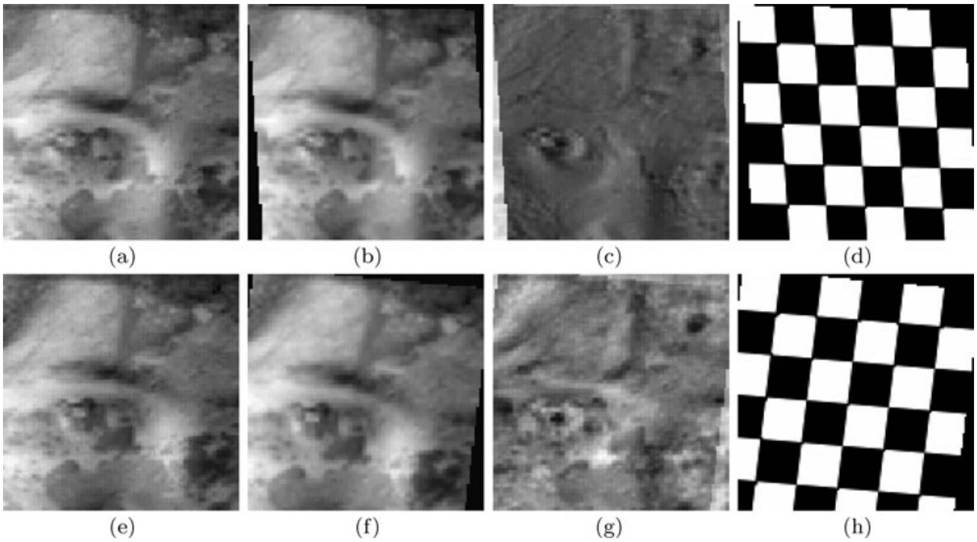


Fig. 10 Performance of the dominant-velocity estimation strategy for the sequence in Fig. 7. In (**a**) frame #3, (**b**) result of the affine registration (i.e., dominant motion estimation) of frame #3 to frame #1, (**c**) registration error, and (**d**) recovered affine registration applied to a regular grid. In (**e**), (**f**), (**g**) and (**h**) same results for frame #9, respectively. The computed displacement is neither correct nor consistent because of the absence of a dominant motion (confront with our correct result in Fig. 7(c))

and are in effect computed as the final solution in Fig. 9(c). We use a dictionary with $N = 97$ velocities: 16 uniformly distributed directions (incremental angle $2\pi/16$) and magnitudes equal to $[0, 0.33, 0.66, 1.0, 1.33, 1.66, 2.0]$. In this experiment we estimate dense smooth flow which does not rely on any motion assumption or model. We note in this complex case that:

1. the recovered (rotational) optical flow looks smooth because of the large dictionary used,
2. the method recovers effectively the multiple OF and,
3. the inter-model competition prevents false motion layer activation.

4.4 Limitations of the *Single-Then-Transparent Velocity Extraction* Algorithm

Here we test the limitations of the *single-then-transparent velocity extraction* strategy (commented in Sect. 2). For this aim we estimate, by using that strategy, the dominant-motion in the noise-free case for our transparent sequence used in Fig. 7. The dominant parametric motion was computed by affine registration with Gauss Newton optimization [50] with a four-level multi-grid approach. The actual velocities are $[-1, 0]$ and $[1, 0]$. The results are shown in Fig. 10. Note that, even for the noise-free case, the dominant motion estimation for different frames is neither correct nor consistent: the estimated velocities are incorrect and pointing in opposite directions because of the absence of a clear dominant motion (see the correct transparent OF in Fig. 7(c)). Note that a correct motion recovery should show a single object (face or rocks) in Figs. 10(c) and 10(g). Given the incorrect initial estimation for the first velocity, it is not possible to estimate the second one. We found similar problems for the rest of the frames (not shown). In order to characterize the sensibility of that method to the amount of transparency, we variate the mixture parameter β of the layer composition $f = \beta I_1 + (1 - \beta) I_2$ in the range $[0.2, 0.8]$ with increments of size 0.1. For the noise-free case we note that the *single-then-transparent velocity extraction* detects the correct motion for $\beta \leq 0.3$ and $\beta \geq 0.7$ (i.e. only when a

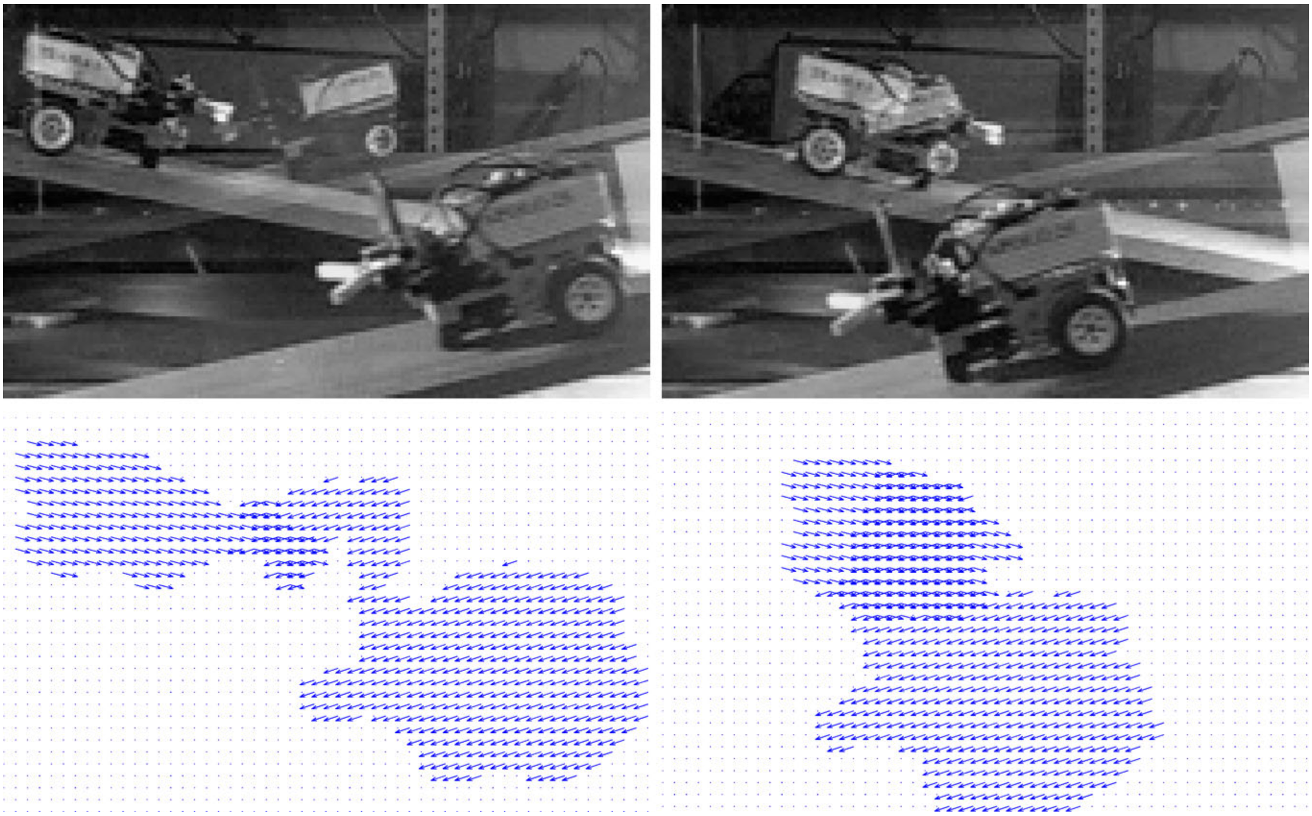


Fig. 11 Real transparent sequence and the computed transparent OF in the *second row*. The used input was d_C

clear dominant motion is present) and is thus more sensitive to the amount of transparency than our approach.

4.5 Transparency and Occlusion in a Real Sequence

We show in Fig. 11 the results obtained for a real sequence composed of two robots moving down a slope. The upper-left robot is located behind a glass while the lower-right robot is located in front of the camera. The reflection of the second robot is located in the upper-central part. For this experiment we used as input the distance measure in (16). The recovered velocities were $[1.5, -0.4]$ pixels for the upper-left robot and $[-1.5, -0.5]$ for both the lower-right robot and its reflection, as it is shown in the second row of Fig. 11. Despite the fact that the lower right-robot is moving a little faster than its reflection (easy to deduct from the projection geometry), both are detected with the same velocity model. This is explained by the low resolution of the discrete velocity basis. We use a dictionary with $N = 231$ velocities taken from an equidistant grid in the interval $[-2.0, 2.0] \times [0.0, -1.0]$ with increments of size 0.1. For this experiment, we perform a spatiotemporal Gaussian smoothing to the sequence (with std $= 0.5$), and we process only the regions that contain displacements: the static background was removed automatically by thresholding the difference between consecutive frames, and then applying opening-closing morphological operators on the activity mask. The processed sequence has a resolution of 28 frames with 101×166 pixels.

4.6 Recovering Rigid Objects from the Transparent OF

In this section we propose a procedure for recovering the rigid transparent objects involved in the sequence of M frames. In the general case, for each frame F_M, F_{M-1}, $F_{M-2}, \ldots, F_2$ we can group neighboring voxels associated to the same α layer, and then displace them according to the computed velocity to the previous frame until reach the first frame (as in an object tracking procedure). Once each set of pixels reached the first frame we have M different images of the same object with a different transparent overlap. Thus, we can average the image intensities for this object, this process eliminates the changing pattern (due to the transparent overlap) and keeps the object's structure.

We illustrate this idea for the noisy sequence (SNR $= 10$) in Fig. 7(a) with $M = 17$ frames. In this case, the process is simple because the whole frame is associated to the two velocity layers shown in Fig. 7(c), thus we recovered the whole images. Figure 12 shows the original sequence and frames in Fig. 12(a) and the recovered image layers in Figs. 12(b)

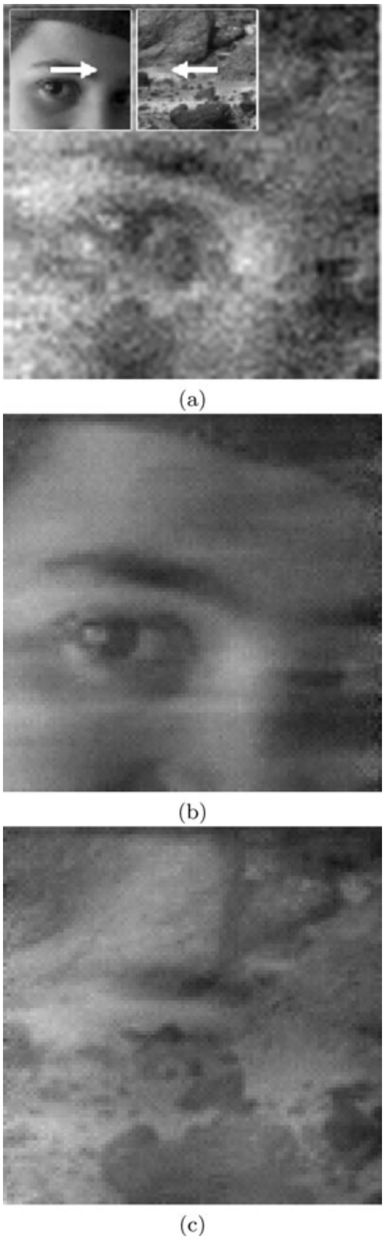


Fig. 12 (**a**) Noisy sequence from experiment in Fig. 7(a) and original frames, (**b**) and (**c**) recovered image frames according to procedure in Sect. 4.6

and 12(c), the root mean squared errors between the actual and recovered frames are 26.72 and 38.65 respectively (the dynamic range of the images is $[0, 255]$). Note that the recovered images present a contrast reduction due to the average process, and also, structures from the other image that lie along the current velocity are kept, because they are constant along the frames and the average can not eliminate them (as for instance in our example, the eyebrow is still present in the rocky landscape).

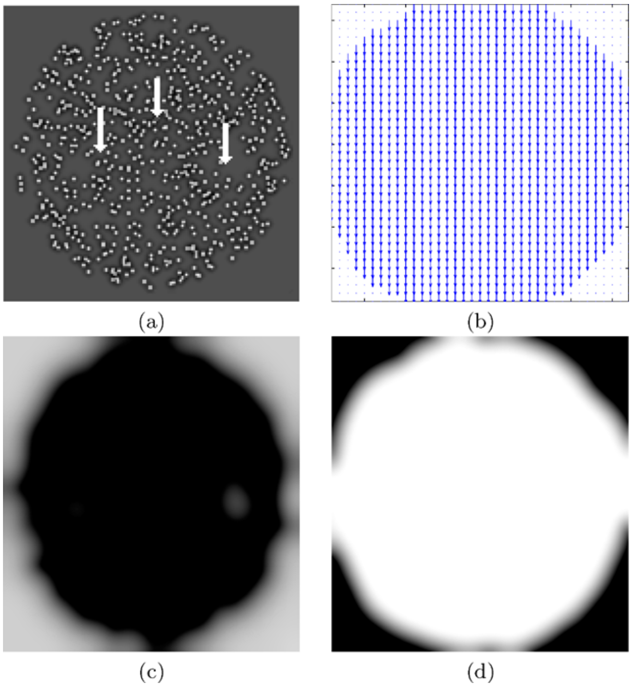


Fig. 13 Results obtained for a RDK sequence with 1 movement. (**a**) Original sequence, (**b**) recovered velocity field, (**c**) and (**d**) show α layers for velocities $[0, 0]$ and $[0, -1]$ respectively

4.7 Results on RDK Sequences

Figure 13 shows the obtained results for a RDK sequence with one movement $\hat{u} = [0, -1]$. The original $301 \times 301 \times 9$ sequence is shown in Fig. 13(a). The recovered subsampled velocity field is shown in Fig. 13(b). We show the recovered velocity layers with values different from zero in order to indicate the regions where each velocity model $\mathbf{u}_i$ is present; Fig. 13(c) shows the α layer associated to the model $\mathbf{u}_i = [0, 0]$ and Fig. 13(d) shows the α layer associated with the velocity model $\mathbf{u}_j = [0, -1]$. The white regions indicate areas with coefficients close to one and the dark zones indicates the presence of coefficients close to zero.

Figure 14 shows a more complex experiment involving a RDK transparent sequence composed of two opposed movements $\hat{u}_1 = [1, 0]$ and $\hat{u}_2 = [-1, 0]$. The sequence dimensions where $301 \times 301 \times 9$, Fig. 14(a). We show the subsampled velocity field in Fig. 14(b). The alpha layers for the three actual movements, $\mathbf{u}_i = [0, 0]$, $\mathbf{u}_j = [1, 0]$ and $\mathbf{u}_k = [-1, 0]$, are shown in Figs. 14(c) 14(d) and 14(e) respectively, the other α layers are composed of zeros. The velocity dictionary is composed of $N = 25$ uniformly distributed vectors. We tested different RDK sequences composed of different velocities (not shown) and we obtained similar results.

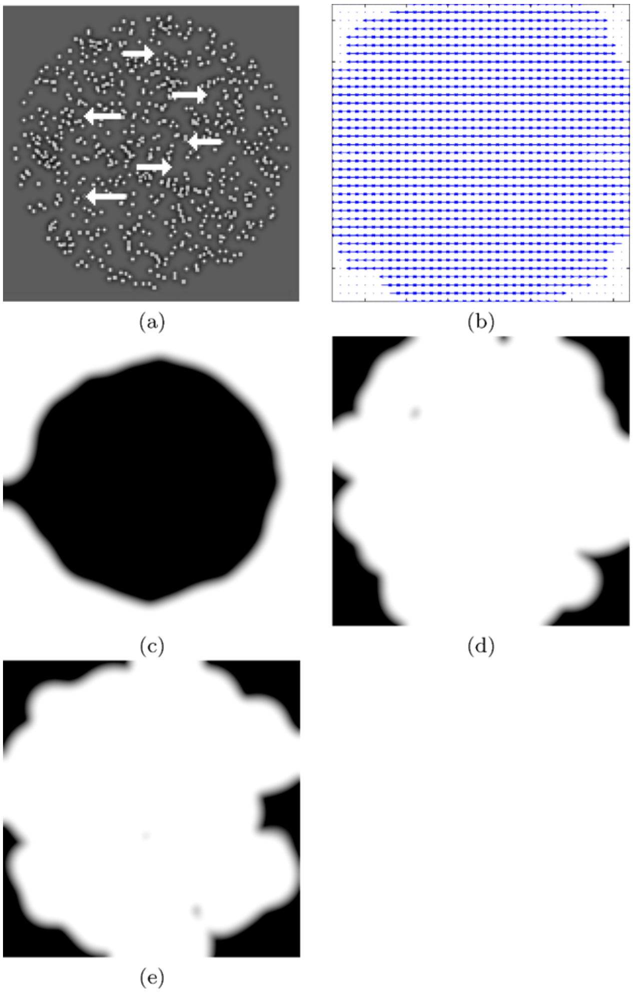


Fig. 14 A transparent RDK sequence and results. (**a**) Original transparent sequence, (**b**) recovered multi-velocity field, (**c**), (**d**) and (**e**) show the α layers for velocities $[0, 0]$, $[1, 0]$ and $[-1, 0]$ respectively

5 Discussion and Conclusion

We have presented a novel variational formulation for the estimation of multiple motions, with an emphasis on transparency. Our regularization proposal is an extension of previous variational approaches based on layered OF computation [36, 37]. The novelty and extensions are the following: What we recover is a vector valued field that indicates the presence or absence of some given motions at each spatiotemporal location. Our formulation (a) extends previously reported methods by using a distance measure suitable for transparent motions, (b) introduces oriented spatial regularization weights which promote a layer's smoothness along the associated velocity model, and (c) proposes an inter-model competition mechanism well-suited for multi-valued solutions. In our case (multiple motions), the inter-model competition behaves similarly to the mechanisms used for entropy-control for single motion fields [36, 37]. This term is by itself a novel contribution of this work, since we do not need special preprocessing in order to tackle sequences with one or more layers, as was shown in synthetic experiments in textured and non-textured sequences as well as in real sequences.

Our non-parametric velocity scheme allows the recovery of an arbitrary number of displacements at the cost of using a fixed dictionary of velocities. The size of such a dictionary determines the required computational burden and the quality of the results. However, a recent work, reported in [51], presents a convenient strategy to deal with a large set of velocity hypothesis; our schema can be extended to use that strategy. In our experiments we tested both a medium-size dictionary for the experiments in Figs. 4, 5, 6, 7, 8, 13, and 14, and a large-size dictionary for the experiments in Figs. 9 and 11.

The robust-to-noise feature of our proposal depicts an additional difference with respect to several methods such as in [39, 40, 52]: to solve highly noisy sequences (as for instance our experiment in Fig. 7) is not the aim of previous methods. For instance, an insight into the pernicious effect of noise in the estimation is given in [39]. With this in mind, we note that our proposal belongs to a different family of approaches, specifically regularization of local transparent velocities, as in [27].

Although our regularization framework can deal with more than two models at the same spatiotemporal position (as demonstrated in [53]), we note that, for practical purposes, the used multi-velocity detectors are unstable (i.e., they are noise sensitive) for more than two velocities. In this sense, the lack of motion detectors for more than two displacements limits the practical application of our proposal to the two-transparent layer cases.

We compared our method with two competitive approaches [27, 39]; our experiments show how our proposal improves the motion detection. However, the automatic transparent OF estimation is still a difficult computer vision task and belongs to an active research area. Our work, as the ones in the state of the art [11, 15–17, 20, 25–28, 39, 42], is based on the assumption that displacements are constant for at least three consecutive frames. Such an assumption is hard to achieve for sequences with non-rigid objects. Consequently, we believe that the development of robust methods for rapid changes in displacements remains an open problem.

We presented an easy-to-implement procedure to recover the rigid transparent layers given the multiple motion field. The quality of the results depends on the number of frames in the sequence given that our framework is based on an average process of the observed layers. Taking into account the available methods for layer separation, it is important to note that they impose different constraints in the transparent sequence in order to be able to separate the layers.

For instance, our method can deal with noisy sequences but then it requires several frames of rigid objects, the approach in [1] works on noise-free sequences and recovers just the smoothest layers (it could have problems with textured objects), the method in [41] separates two layers by assuming that one presents a periodical movement, the work in [39] requires sequences containing a dominant motion, and the proposal in [42] works based on a specific camera motion constraint. In this regard, it is important to investigate new approaches for constraint-free transparent sequences.

Finally, we have tested the application of our regularization proposal on RDK sequences: we proposed a regional based single motion estimation on a diffusion based integration. Our experiments show that is possible to use our model to integrate such single-motion local estimations into transparent layers.

In future work, we consider that it is important to study in more depth the diffusion terms of the spatial integration and also investigate how different velocity maps may interact.

Acknowledgements This research was supported in part by CONACYT, Mexico: SNI Scholarship and grant 61367 for A. Ramirez-Manzanares and M. Rivera, respectively. F. Lauze has been supported by the Danish Research Council, Program Committee on Strategic Growth, under Grant number 09-065145.

Appendix: The Inter-Model Regularization

Sparsity or "compact coding" scheme has been biologically motivated: in the context of biological vision models it is desirable to minimize the number of models that respond to any particular event. The prior information is that the probability of any model to be present is equal, but such probability is low for any given model. Such coding scheme is like the one suggested by experiments in retina and primary visual cortex analysis of natural images [54–56], where a suitable response may be described in terms of structural primitives (as lines for instance), so that only a small subset from all detectors should indicate contribution.

In order to promote sparsity in the case of probability measure vectors, $\alpha(\mathbf{r})$, Weiss and Adelson proposed in [36] to penalize the Shannon's entropy:

$$\hat{U}_{c_1}(\alpha,\mathbf{r}) = -\sum_i \alpha_i(\mathbf{r})\log(\alpha_i(\mathbf{r})).$$

The high non linearity of this term and its derivatives makes the minimization scheme rather complex. Instead, Rivera et al. propose in [37] to penalize the Gini's entropy by means of a quadratic potential as

$$\hat{U}_{c_2}(\alpha,\mathbf{r}) = 1 - \sum_i \alpha_i^2(\mathbf{r}).$$

Note however that in both cases the minimum entropy is reached for unimodal solutions [57] (i.e. with only one coefficient $\alpha_i(\mathbf{r})$ close to one and all the others close to zero), an undesirable behavior for this work. Multimodal non linear potentials that promote sparsity have been proposed by Olshausen et al. in [45] as

$$\hat{U}_{c_3}(\alpha,\mathbf{r}) = -\sum_i \exp(-\alpha_i^2(\mathbf{r})),$$

$$\hat{U}_{c_4}(\alpha,\mathbf{r}) = \sum_i \log(1+\alpha_i^2(\mathbf{r}))$$

and

$$\hat{U}_{c_5}(\alpha,\mathbf{r}) = \sum_i |\alpha_i(\mathbf{r})| = \|\alpha(\mathbf{r})\|_1.$$

For these potentials, the authors reported no meaningful differences in implementation, although their non linear feature results in well-known minimization drawbacks.

In order to overcome the shortcomings of the above mentioned terms, we propose to perform an *inter-model* regularization over the $\alpha(\mathbf{r})$ vector by means of a quad-ratic term defined in a general form as

$$\hat{U}_c(\alpha,\mathbf{r}) = -\sum_i \left(\alpha_i(\mathbf{r}) - C_{\alpha(\mathbf{r})}\right)^2, \tag{22}$$

where $C_{\alpha(\mathbf{r})}$ is an intermediate value lying between zero and the maximum value expected by any α_i coefficient (1 in our case). This term promotes large contrast in the $\alpha_i(\mathbf{r})$ coefficients, since it is minimized as each $\alpha_i(\mathbf{r})$ is as far as possible from the value $C_{\alpha(\mathbf{r})}$. We can understand $C_{\alpha(\mathbf{r})}$ as a parameter which acts as a threshold between the significant and non significant models.

Take $C_{\alpha(\mathbf{r})}$ to be the mean value $\bar{\alpha}(\mathbf{r})$ of the $\alpha(\mathbf{r})$ vector: $\bar{\alpha}(\mathbf{r}) = \frac{1}{N}\sum_k \alpha_k(\mathbf{r})$ (as in [44, 53]): (22) becomes proportional to the opposite of the variance of the $\alpha_i(\mathbf{r})$'s, therefore encouraging deviation with respect to their mean. In fact, by expanding the contrast term (22), one gets

$$\hat{U}_c(\alpha,\mathbf{r}) = -\sum_i \alpha_i(\mathbf{r})^2 + N\bar{\alpha}(\mathbf{r})^2. \tag{23}$$

Thus, the first term of (23) promotes the "switching on" of models and avoids the trivial solution $\alpha(\mathbf{r}) = 0$, while the second term penalizes the number of switched-on models. Hence for a fixed mean value (controlled by the second term) the first term (that acts in a similar way to Gini's entropy term) prefers highly contrasted solutions.

According to our experiments and inspired from the previous analysis, we propose a more versatile inter-model regularization potential in (20) as

$$U_c(\alpha,\mathbf{r}) = -\sum_i \alpha_i(\mathbf{r})^2 + \kappa N\bar{\alpha}(\mathbf{r})^2$$

where $\kappa > 0$. In this way, we can tune the κ parameter depending on the number of models that we want to detect at each pixel, see the Sect. 3.2.3 for illustrative experiments.

References

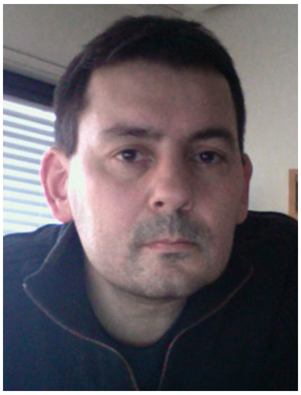
1. Toro, J., Owens, F., Medina, R.: Using known motion fields for image separation in transparency. Pattern Recognit. Lett. **24**, 597–605 (2003)
2. Oppenheim, A.V.: Superposition in a class of nonlinear systems. In: Proceedings of IEEE International Convention, New York, USA, pp. 171–177 (1964)
3. Guenther, R.D.: Modern Optics. Wiley, New York (1990)
4. Qian, N., Andersen, R., Adelson, E.: Transparent motion perception as detection of unbalanced motion signals. III. Modeling. J. Neurosci. **14**(12), 7381–7392 (1994)
5. Ramirez-Manzanares, A., Rivera, M., Kornprobst, P., Lauze, F.: A variational approach for multi-valued velocity field estimation in transparent sequences. In: Proceedings of International Conference on Scale Space and Variational Methods in Computer Vision, Ischia, Italy. LNCS, vol. 4485, pp. 227–238 (2007)
6. Horn, B., Schunck, B.: Determining optical flow. Artif. Intell. **17**, 185–203 (1981)
7. Papenberg, N., Bruhn, A., Brox, T., Didas, S., Weickert, J.: Highly accurate optical flow computations with theoretically justified warping. Int. J. Comput. Vis. **67**(2), 141–158 (2006)
8. Nir, T., Bruckstein, A., Kimmel, R.: Over-parameterized variational optical flow. Int. J. Comput. Vis. **76**(2), 205–216 (2008)
9. Barron, J., Fleet, D., Beauchemin, S.: Performance of optical flow techniques. Int. J. Comput. Vis. **12**(1), 43–77 (1994)
10. Baker, S., Scharstein, D., Lewis, J.P., Roth, K., Black, M.J., Szeliski, R.: A database and evaluation methodology for optical flow. In: Proceeding of the 11th International Conference on Computer Vision, IEEE, Rio de Janeiro, Brazil (2007)
11. Bergen, J.R., Burt, P.J., Hingorani, R., Peleg, S.: Computing two motions from three frames. In: Third International Conference on Computer Vision, Osaka, Japan, pp. 27–32 (1990)
12. Burt, P.J., Hingorani, R., Kolczynski, R.J.: Mechanisms for isolating component patterns in the sequential analysis of multiple motion. In: Proceedings of IEEE Workshop on Visual Motion, Princeton, NJ, pp. 187–193 (1991)
13. Irani, M., Peleg, S.: Motion analysis for image enhancement: resolution, occlusion, and transparency. J. Vis. Commun. Image Represent. **4**(4), 324–335 (1993)
14. Irani, M., Rousso, B., Peleg, S.: Computing occluding and transparent motions. Int. J. Comput. Vis. **12**(1), 5–16 (1994)
15. Shizawa, M., Mase, K.: Simultaneous multiple optical flow estimation. In: Proceedings of 10th International Conference on Pattern Recognition, pp. 274–278 (1990)
16. Shizawa, M., Mase, K.: Principle of superposition: a common computational framework for analysis of multiple motion. In: Proceedings of IEEE Workshop on Visual Motion, pp. 164–172 (1991)
17. Shizawa, M., Mase, K.: A unified computational theory for motion transparency and motion boundaries based on eigenergy analysis. In: Proceedings of the International Conference on Computer Vision and Pattern Recognition, IEEE, Lahaina, Hawai, pp. 289–295 (1991)
18. Förstner, W.: A feature based corresponding algorithm for image matching. Int. Arch. Photogramm. Remote Sens. **26**, 150–166 (1986)
19. Bigun, J., Granlund, G.H., Wiklund, J.: Multidimensional orientation estimation with applications to texture analysis and optical flow. IEEE Trans. Pattern Anal. Mach. Intell. **13**(8), 775–790 (1991). Report LiTH-ISY-I-0828 1986 and Report LiTH-ISY-I-1148 1990, both at Computer Vision Laboratory, Linköping University, Sweden
20. Mota, C., Stuke, I., Aach, T., Barth, E.: Divide-and-Conquer strategies for estimating multiple transparent motions. In: Proceedings of 1st International Workshop on Complex Motion, Schloss Reisensburg, Germany. LNCS, vol. 3417, pp. 66–78 (2005)
21. Mühlich, M., Aach, T.: A theory of multiple orientation estimation. In: Proceedings of the European Conference on Computer Vision. LNCS, vol. 3952, pp. 69–82 (2006)
22. Vernon, D.: Decoupling Fourier components of dynamic image sequences: a theory of signal separation, image segmentation and optical flow estimation. In: European Conference on Computer Vision, vol. 2, pp. 69–85 (1998)
23. Zhou, W., Kambhamettu, C.: Separation of reflection by Fourier decoupling. In: Asian Conference on Computer Vision, Jeju Island, Korea (2004)
24. Liu, H.C., Hong, T.H., Herman, M., Chellappa, R.: Spatio-temporal filters for transparent motion segmentation. In: Proceedings of the International Conference on Image Processing, Washington, USA, pp. 464–468 (1995)
25. Darrell, T., Simoncelli, E.: Separation of transparent motion into layers using velocity-tuned mechanisms. Tech. Rep. 244, MIT Media Laboratory Vision and Modeling Group (1993)
26. Stuke, I., Aach, T., Barth, E., Mota, C.: Estimation of multiple motions by block matching. In: Proceedings in 4th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, pp. 358–362 (2003)
27. Stuke, I., Aach, T., Barth, E., Mota, C.: Multiple-motion-estimation by block matching using MRF. Int. J. Comput. Inform. Sci. **26**, 141–152 (2004)
28. Auvray, V., Bouthemy, P., Lienard, J.: Motion estimation in x-ray image sequence with bi-distributed transparency. In: Proceedings of IEEE International Conference on Image Processing, Atlanta, USA, pp. 1057–1060 (2006)
29. Fitzpatrick, J.M.: The existence of geometrical density-image transformations corresponding to object motion. Comput. Vis. Graph. Image Process. **44**(2), 155–174 (1988)
30. Pingault, M., Bruno, E., Pellerin, D.: A robust multiscale b-spline function decomposition for estimating motion transparency. IEEE Trans. Image Process. **12**(11), 1416–1426 (2003)
31. Black, M.J., Anandan, P.: The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. CVGIP, Image Underst. **63**(1), 75–104 (1996)
32. Jepson, A., Black, M.J.: Mixture models for optical flow computation. In: Proceedings of Intl. Conf. on Computer Vision and Pattern Recognition, pp. 760–761 (1993)
33. Ju, S.X., Black, M.J., Jepson, A.D.: Skin and bones: Multi-layer, locally affine, optical flow and regularization with transparency. In: Proceedings of Intl. Conf. on Computer Vision and Pattern Recognition, San Francisco, CA, pp. 307–314 (1996)
34. Black, M.J., Fleet, D.J., Yacoob, Y.: Robustly estimating changes in image appearance. Comput. Vis. Image Underst. **78**, 8–31 (2000)
35. Jojic, N., Frey, B.: Learning flexible sprites in video layers. In: Proceedings in IEEE Conf. on Computer Vision and Pattern Recognition, pp. 199–206 (2001)
36. Weiss, Y., Adelson, E.H.: A unified mixture framework for motion segmentation: incorporating spatial coherence and estimating the number of models. In: Proceedings of the International Conference on Computer Vision and Pattern Recognition, IEEE, San Francisco, CA, pp. 321–326 (1996)
37. Rivera, M., Ocegueda, O., Marroquin, J.L.: Entropy-controlled quadratic Markov measure field models for efficient image segmentation. IEEE Trans. Image Process. **16**(12), 3047–3057 (2007)

38. Ramirez-Manzanares, A., Rivera, M., Kornprobst, P., Lauze, F.: Multi-valued motion fields estimation for transparent sequences with a variational approach. Tech. Rep. RR-5920, INRIA (Also, Reporte Técnico CIMAT, (CC)I-06-12) (2006)
39. Szeliski, R., Avidan, S., Anandan, P.: Layer extraction from multiple images containing reflections and transparency. In: Proceedings in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 246–253 (2000)
40. Nicolescu, M., Medioni, G.: Layered 4d representation and voting for grouping from motion. IEEE Trans. Pattern Anal. Mach. Intell. **25**(4), 492–501 (2003)
41. Sarel, B., Irani, M.: Separating transparent layers of repetitive dynamic behaviors. In: Proceedings of the Tenth International Conference on Computer Vision, vol. 1, pp. 26–32. IEEE Computer Society, Beijing (2005)
42. Oo, T., Kawasaki, H., Ohsawa, Y., Ikeuchi, K.: The separation of reflected and transparent layers from real-world image sequences. Mach. Vis. Appl. **18**(1), 17–24 (2007)
43. Black, M.J., Rangarajan, P.: On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. Int. J. Comput. Vis. **19**(1), 57–91 (1996)
44. Ramirez-Manzanares, A., Rivera, M.: Brain nerve boundless estimation by restoring and filtering intra-voxel information in DT-MRI. In: Proceedings of Second Workshop on Variational and Level Sets Methods, pp. 71–80 (2003)
45. Olshausen, B.A., Field, D.J.: Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature **381**, 607–609 (1996)
46. Blake, A., Zisserman, A.: Visual Reconstruction, 1st edn. MIT Press, Cambridge (1987)
47. Grossberg, S., Mingolla, E., Viswanathan, L.: Neural dynamics of motion integration and segmentation within and across apertures. Vis. Res. **41**, 2521–2553 (2001)
48. Braddick, O.: Local and global representations of velocity; transparency opponency and global direction perception. Perception **26**, 995–1010 (1997)
49. Weiss, Y., Simoncelli, E., Adelson, E.: Motion illusions as optimal percepts. Nat. Neurosci. **5**(6), 598–604 (2002)
50. Fleet, D.J., Weiss, Y.: Optical flow estimation. In: Paragios, N., Chen, Y., Faugeras, O. (eds.) Mathematical Models for Computer Vision: The Handbook. Springer, Berlin (2005)
51. Bayerl, P., Neumann, H.: A fast biologically inspired algorithm for recurrent motion estimation. IEEE Trans. Pattern Anal. Mach. Intell. **29**(2), 246–260 (2007)
52. Sarel, B., Irani, M.: Separating transparent layers through layer information exchange. In: Pajdla, T., Matas, J. (eds.) Proceedings of the 8th European Conference on Computer Vision, pp. 328–341. Springer, Berlin (2004)
53. Ramirez-Manzanares, A., Rivera, M.: Basis tensor decomposition for restoring intra-voxel structure and stochastic walks for inferring brain connectivity in DT-MRI. Int. J. Comput. Vis. **69**(1), 77–92 (2006)
54. Field, D.J.: Scale-invariance and self-similar 'wavelet' transforms: an analysis of natural scenes and mammalian visual systems. In: O.U. Press (ed.) Wavelets, Fractals and Fourier Transforms: New Developments and New Applications. Oxford University Press, London (1993)
55. Field, D.J.: What is the goal of sensory coding? Neural Comput. **6**, 559–601 (1994)
56. Olshausen, B.A., Field, D.J.: Sparse coding with an overcomplete basis set: a strategy employed by v1? Vis. Res. **37**, 3311–3325 (1997)
57. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 1st edn. Springer, New York (2003)

Alonso Ramírez-Manzanares has studied at the Research Center of Mathematics (CIMAT, A.C.), Mexico, where he graduated with a Doctorate in Computer Science in 2007. His work is focused on low-level vision and image processing. During his Masters and D.Sc. studies, he worked on the development of 'multi-model' regularization methods. These models are capable of estimating and regularizing more than a single model per pixel/voxel. From 2007 to the present he is a faculty member and researcher at the Mathematics Department of the University of Guanajuato. As a researcher, his main interests is to develop applications for solving general regularization problems on robotic vision and medical image processing. Dr. Ramirez-Manzanares is Fellow of the National Researcher System (SNI) of the Mexican Government.

Mariano Rivera received in 1989 the B.E. degree in electronics from the Durango Institute of Technology, Mexico. In 1993 he received the M.Sc. degree in electronics from the Chihuahua Institute of Technology, Mexico. In 1997 he received the D.Sc. degree in optics from the Center for Research in Optics (CIO), Leon, Mexico. Since 1997, he is with the Computer Science Department at the Center for Research in Mathematics (CIMAT), Guanajuato, Mexico. His current interests include computer vision, image processing, machine learning and optical metrology. His research is summarized in more than 60 papers in scientific journals and conference proceeding. Dr. Rivera is Fellow of the National Researcher System (SNI) of the Mexican Government.

Pierre Kornprobst received his Ph.D. in Mathematics from Nice-Sophia Antipolis University, France, in 1998. He is a research scientist at INRIA. His research interests include calculus of variations, nonlinear PDEs and numerical analysis as applied to image processing (e.g., image restoration, super resolution, inpainting, motion estimation and motion recognition). He is the co-author of the 2002 monograph Mathematical Problems in Image Processing (Springer). Since 2002, he focuses on the exploration of the brain from the mathematical and computational perspectives. His goal is to bridge the gap between biological and computational vision by proposing novel bio-inspired models.



François Lauze has studied Mathematics at the University of Nice-Sophia-Antipolis where he graduated with a Ph.D. in Algebraic Geometry in 1994, before moving to Ouagadougou, Burkina Faso, where he taught mathematics and then Denmark where he studied Image Analysis at the IT University of Copenhagen and graduated with yet another Ph.D. in 2004. From 2006 to 2008 he was a Senior Researcher with Nordic Bioscience Imaging and since 2008 he has been with the Image Group at the University of Copenhagen where he is currently associate professor. His main topics of interest include Image reconstruction and segmentation, motion and video processing, shapes and shape statistics, using variational framework, partial differential equations, differential and Riemannian geometry.