Fast Piecewise-Affine Motion Estimation Without Segmentation

Denis Fortun, Martin Storath, Dennis Rickert, Andreas Weinmann, and Michael Unser, *Fellow, IEEE*

***Abstract*— Current algorithmic approaches for piecewise affine motion estimation are based on alternating motion segmentation and estimation. We propose a new method to estimate piecewise affine motion fields directly without intermediate segmentation. To this end, we reformulate the problem by imposing piecewise constancy of the parameter field, and derive a specific proximal splitting optimization scheme. A key component of our framework is an efficient 1D piecewise-affine estimator for vector-valued signals. The first advantage of our approach over segmentation-based methods is its absence of initialization. The second advantage is its lower computational cost, which is independent of the complexity of the motion field. In addition to these features, we demonstrate competitive accuracy with other piecewise-parametric methods on standard evaluation benchmarks. Our new regularization scheme also outperforms the more standard use of total variation and total generalized variation.**

***Index Terms*— Motion estimation, optical flow, optimization, piecewise affine.**

I. Introduction

TWO important prior models have been explored for motion estimation. The first one works with a dense representation of motion and imposes at each pixel a smoothness constraint [19] such as total variation (TV) [5], [56] or total generalized variation (TGV) [4], [44]. These regularization terms are most often convex and well suited to a large collection of optimization techniques. However, they create undesirable artefacts like staircasing or smoothing of motion discontinuities. The second type of prior model works with parametric representations of motion, which may be chosen to provide a satisfying match of 3D translations on the camera plane. In particular, the piecewise-affine model overcomes the limitations of TV and TGV and has yielded very accurate results in several previous works [37], [42], [55]. Yet, in spite of these achievements, local smoothness priors are still preferred to piecewise-parametric ones in most motion estimation methods. The main difficulty that restrain the spreading of the piecewise-parametric model is the challenging optimization problem it involves. In this work, we address this optimization issue.

The problem is usually formulated as the joint segmentation of the motion field and estimation of the parameters inside each region, following the seminal work of Mumford and Shah [25] for the image segmentation part. The interdependency of these two tasks translates into highly non-convex optimization. The existing solutions proceed iteratively by alternating an optimization step with respect to the image partition and an optimization step with respect to the motion parameters. This alternating scheme causes two main issues. Firstly, the resulting scheme is very sensitive to initialization and can only be used for refinement. Secondly, the computational cost is often prohibitive for practical applications. In particular, it depends on the number of regions, which should typically be very large to achieve high accuracy.

In this paper, we propose a new method to estimate piecewise-affine motion fields. It eschews the explicit segmentation of motion and the aforementionned limitations of existing methods. Our main contributions can be summarized as follows:

- We revisit the standard formulation and impose a piecewise-constant regularization of the field of affine parameters. It leads to the direct estimation of a piecewise-affine motion field without intermediate segmentation.
- We propose an optimization method based on a specific proximal-splitting approach that yields a series of 1D piecewise-affine vectorial estimation problems. We design an efficient solver that is based on dynamic programming and inspired by the works on segmentation described in [33], [35], and [48].
- We present experimental evidence on the reference benchmarks *MPI Sintel* [7] and *Kitti* [16] showing that our approach outperforms the standard TV and TGV regularizations, and that it is competitive with the best performing piecewise-parametric methods [40], [55].

Manuscript received October 25, 2017; revised April 12, 2018; accepted June 30, 2018. Date of publication July 23, 2018; date of current version August 14, 2018. This work was supported in part by the German Research Foundation DFG under Grant STO1126/2-1 and Grant WE5886/4-1 and in part by the European Research Council under Grant 692726 (H2020-ERC Project GlobalBioIm). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Shuicheng Yan. *(Corresponding author: Denis Fortun.)*

D. Fortun is with the Signal Processing Core of Center for Biomedical Imaging, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland (e-mail: denis.fortun@epfl.ch).

M. Storath is with the Interdisciplinary Center for Scientific Computing, Image Analysis and Learning Group, Universität Heidelberg, 69117 Heidelberg, Germany.

D. Rickert is with the Institute of Computational Biology, Helmholtz Zentrum München, 85764 Neuherberg, Germany.

A. Weinmann is with the Institute of Computational Biology, Helmholtz Zentrum München, 85764 Neuherberg, Germany, and also with the Department of Mathematics and Natural Sciences, Hochschule Darmstadt, 64295 Darmstadt, Germany.

M. Unser is with the Biomedical Imaging Group, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2018.2856399

1057-7149 © 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

- Our optimization scheme does not require any initialization of the motion field, and it is faster than other piecewise-parametric approaches. In particular, the computational time does not depend on the complexity of the motion field.

Thus, our method combines the advantages of the piecewise-affine model with robustness and a low computational cost. It can be integrated as a regularizer in various motion estimation frameworks.

The outline of the paper is as follows: In Section II, we review related works and identify their limitations. In Section III, we describe the model and optimization method that we propose. In Section IV, we evaluate the performance of our method on standard benchmarks. In Section V, we give concluding remarks.

II. Related Works

Motion estimation aims at recovering the deformation field between to consecutive frames of an image sequence. Piecewise-parametric approaches based on joint motion segmentation and parameter estimation have been formulated as an optimization problem of the form

$$J = \min_{\mathcal{R},\mathbf{w}} \left(\rho(\mathbf{w}) + \frac{\lambda}{2} \sum_{n=1}^{L} \mathcal{L}(R_n) \right), \tag{1}$$

where L is the number of regions, $\mathcal{R} = \{R_1, \ldots, R_L\}$ is a partition of the image, $\mathbf{w}$ is a piecewise-parametric motion field on $\mathcal{R}$ (i.e. $\mathbf{w}$ is parametric on each R_n for $n = 1, \ldots, L$.), ρ is a function that imposes data fidelity, $\mathcal{L}$ is a segmentation prior usually defined as the total boundary length delineating the segmented regions R_n, and λ is a balance parameter between ρ and $\mathcal{L}$. Problems of this type are typically minimized alternately with respect to $\mathcal{R}$, which amounts to an image-partitioning problem, and with respect to $\mathbf{w}$, which amounts to a parametric motion fitting. A similar formulation has also been used recently for image segmentation [57].

The differences between existing methods concern mainly the solver for the image-partitioning problem. In a continuous setting, following the approach of Mumford and Shah [25], the problem has been addressed with an implicit level-set representation of the partitioning curve in [13], [28], and [43]. A primal-dual optimization strategy was used in [42]. In a discrete setting, iterated conditional modes and high confidence first approaches were exploited in [3], [24], and [27]. Graph-cuts methods have also been used in [31], and more recently in [55]. Layered models, introduced in [46], involve a similar optimization problem but add a depth information between the different regions, from which occlusions can be derived. This model has been revitalized in [32] and [37]–[39].

The first limitation of these approaches is their sensitivity to initialization when optimizing (1) with an alternating scheme. This is illustrated in [39] and [42], where the optimization is initialized through advanced motion estimation methods [36], [50]. In [8], an alternating direction method of multipliers (ADMM) approach is used to solve (1) without intermediate segmentation steps. However, the underlying model is piecewise-constant and not rich enough in most practical scenarios; it is initialized by a block matching algorithm.

The second limitation is the computational time, which is mostly spent on the image-partitioning problem. The earliest works retain at most five regions to make the problem tractable [3], [13], [24], [28]. More recently, the layered approach [39] handles a larger number of regions but requires several hours of computation, and the primal dual approach [42] can take up to one hour despite a GPU implementation. The method proposed in [55] achieves around fifteen minutes for (1240×370) image, with a graph cut minimization approach.

In contrast, our method is initializaton-free and has a significantly lower computational cost.

Beyond solving (1), other techniques can be involved to improve the results. They include the handling of occlusions [2], [27], [42], label cost terms to limit the number of regions [42], [55], edge-driven models to fit image boundaries [28], deviations from the parametric models to estimate more complex deformations [37], smoothness of the parameters of neighboring regions [55], or post-processing refinements with a variational optimization of TV-based models [55]. Yet other methods rely on similar principles but incorporate additional information obtained from their applicative context, like epipolar constraints [21], [45], temporal consistency [21], or semantic information about the type of moving objects in the scene [32].

Extensions of TV to second order derivatives result in approximately piecewise-affine solutions [29], [41]. However, the ℓ_1 norm does not delineate moving objects as sharply as the Mumford-Shah model (1). In this line, the over-parametrized approach [20], [26], which models a spatially varying parameter field with TV regularization, also shows this undesirable effect.

III. Proposed Piecewise-Affine Estimation

In this section, we detail our method to estimate piecewise-affine motion fields. After the model and minimization problem, we present our optimization strategy based on directional splitting. The splitting leads to vector-valued denoising problems in 1D with piecewise-affine regularization. We provide an efficient solver for these problems (Section III-D).

A. Piecewise-Affine Model

Let two successive frames of an image sequence be $I_1, I_2 : \Omega \to \mathbb{R}$, where $\Omega \subset \mathbb{N}^2$ is the image grid. Our goal is to estimate the piecewise-affine motion field $\mathbf{w} : \Omega \to \mathbb{R}^2$ that transports I_1 to I_2 according to (1). It is common to discretize the length $\mathcal{L}(R_n)$ of the boundary of a segment R_n by

$$\mathcal{L}(R_n) = \sum_{k=1}^{K} \alpha_k |\{\mathbf{x} \in R_n : \mathbf{x} + \mathbf{d}_k \notin R_n\}|, \tag{2}$$

where $\mathbf{d}_k$ is an element of the set $\mathcal{D} \subset (Z^2)^K$ of directions, $\alpha_k > 0$ is its corresponding weight [1], [9], and $|A|$ denotes the number of elements of a set $A \subset \Omega$.. The choice of $\mathcal{D}$ and α determines how well the regularizer approximates

rotational invariance. Considering only horizontal and vertical directions, with $\mathcal{D} = \{(0,1),(1,0)\}$, creates block artifacts similar to those of anisotropic TV regularization. To attenuate them, we use the four-directional neighborhood system $\mathcal{D} = \{(0,1),(1,0),(1,1),(-1,1)\}$ that includes diagonal directions. The weights are chosen such that the norm built from the basis vectors of $\mathcal{D}$ best approximates the isotropic Euclidean norm [9], [35].

Henceforth, we assume that the motion field $\mathbf{w}$ can be written in terms of the parameter field $\mathbf{P} : \Omega \to \mathbb{R}^{2\times 3}$ as

$$\mathbf{w}(\mathbf{x}) = \mathbf{P}(\mathbf{x})\bar{\mathbf{x}} \tag{3}$$

for all $\mathbf{x} \in \Omega$, where $\bar{\mathbf{x}} = (x_1, x_2, 1)$ denotes the homogeneous coordinates of $\mathbf{x}$. When $\mathbf{P}$ is piecewise constant, it defines a partition $\mathcal{R}$ of the domain Ω. This allows us to conveniently express the piecewise-affine model (1) as

$$\begin{aligned} J = \min_{\mathbf{w},\mathbf{P}} \left(\rho(\mathbf{w}) + \lambda \sum_{k=1}^{K} \alpha_k \|\nabla_{\mathbf{d}_k} \mathbf{P}\|_0 \right) \\ \text{s.t. } \mathbf{w}(\mathbf{x}) = \mathbf{P}(\mathbf{x})\bar{\mathbf{x}} \end{aligned} \tag{4}$$

for all $\mathbf{x} \in \Omega$, where $\|\nabla_{\mathbf{d}_k}\mathbf{P}\|_0$ counts the number of parameter changes with respect to the direction $\mathbf{d}_k$, as given by

$$\|\nabla_{\mathbf{d}_k}\mathbf{P}\|_0 = |\{\mathbf{x} \in \Omega : \mathbf{P}(\mathbf{x}) \neq \mathbf{P}(\mathbf{x}+\mathbf{d}_k)\}|\,. \tag{5}$$

Note that the factor $\frac{1}{2}$ that was compensating for the double counting of the boundary lengths in (1) is not needed in (4). Although our final goal is to estimate the flow field $\mathbf{w}$, the introduction of $\mathbf{P}$ in (4) is important for the derivation of our proposed algorithm. Differently from the over-parametrized approach [17], [26], we do not estimate the parameters but directly the motion field.

The data term in (1) reflects the assumption of the conservation of an image feature along the motion trajectory. Here, we rely on the usual assumption of constant brightness and penalize deviations with an ℓ_1 norm to gain robustness to local violations such as occlusions or illumination changes. The linearized form of this criterion is

$$\rho_{\mathrm{d}}(\mathbf{w}) = \sum_{\mathbf{x}\in\Omega} |\nabla^\top I_2(\mathbf{x})\, \mathbf{w}(\mathbf{x}) + I_t(\mathbf{x})|, \tag{6}$$

where $\nabla^\top = \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}\right)^\top$ and I_t is the discrete temporal image gradient given by $I_t = (I_2 - I_1)$. Note that this data term (6) does not depend on the parameter field $\mathbf{P}$ but only on the associated flow field $\mathbf{w}$.

B. Splitting Approach and Augmented Lagrangian Resolution

The problem (4) is non-convex and NP-hard. Thus, the convergence to a global minimum cannot be guaranteed. To find a practical solution, we devise a splitting strategy. We divide (4) into easier subproblems in an ADMM-like augmented-Lagrangian framework, which has turned out to often work well for non-convex problems [11], [18], [34], [47], [54].

The starting point for our method is the formulation in terms of the parameter field (4). We introduce splitting variables $\mathbf{P}_k$ to decouple the data term and the terms associated to the directions $\mathbf{d}_k$ of the regularization. This leads to the optimization problem

$$\begin{aligned} J = \min_{\mathbf{w},\mathbf{P}_1,\ldots,\mathbf{P}_K} \rho(\mathbf{w}) + \lambda \sum_{k=1}^{K} \alpha_k \|\nabla_{d_k}\mathbf{P}_k\|_0 \\ \text{s.t. } \mathbf{w}(\mathbf{x}) = \mathbf{z}_k(\mathbf{x}), \\ \mathbf{z}_k(\mathbf{x}) = \mathbf{P}_k(\mathbf{x})\bar{\mathbf{x}}, \quad \forall \mathbf{x} \in \Omega,\ \forall k = 1,\ldots,K. \end{aligned} \tag{7}$$

Then, the augmented Lagrangian (in scaled form) of (7) writes

$$\begin{aligned} &\mathcal{A}_\eta(\mathbf{w}, \{\mathbf{P}_k\}_k, \{\mathbf{z}_k\}_k, \{\boldsymbol{\mu}_k\}_k) \\ &= \rho(\mathbf{w}) + \lambda \sum_{k=1}^{K} \alpha_k \|\nabla_{\mathbf{d}_k}\mathbf{P}_k\|_0 \\ &\quad + \frac{\eta}{2} \sum_{k=1}^{K} \sum_{\mathbf{x}\in\Omega} \left\| \mathbf{w}(\mathbf{x}) - \mathbf{z}_k(\mathbf{x}) + \frac{\boldsymbol{\mu}_k(\mathbf{x})}{\eta} \right\|_2^2 - \frac{1}{2\eta} \left\|\boldsymbol{\mu}_k(\mathbf{x})\right\|_2^2, \\ &\quad \text{s.t. } \mathbf{z}_k(\mathbf{x}) = \mathbf{P}_k(\mathbf{x})\bar{\mathbf{x}}, \quad \forall \mathbf{x} \in \Omega,\ \forall k \in \{1,\ldots,K\}, \end{aligned}$$

where $\{\boldsymbol{\mu}_k\}_{k=1,\ldots,K}$ are Lagrange multipliers and $\eta > 0$ is a parameter that controls the fulfillment of the constraints, and influences the speed of convergence. Further, $\|\cdot\|_2^2$ denotes the squared Euclidean norm in $\mathbb{R}^2$. Note that we include the equality constraints into the target functional only with respect to the motion field variables. The couplings of the parameter fields $\mathbf{P}_k$ and the flow fields $\mathbf{z}_k$ remain as explicit constraints. This will become important when solving the subproblems.

Next, we follow the ADMM strategy and iteratively minimize the augmented Lagrangian with respect to $\mathbf{w}$ and $\mathbf{z}_k$, and perform gradient ascents on the Lagrange multipliers as

$$\begin{aligned} \mathbf{w}^{(n+1)} &= \arg\min_{\mathbf{w}} \mathcal{A}_\eta(\mathbf{w}, \mathbf{z}_1^{(n)}, \ldots, \mathbf{z}_K^{(n)}) \\ (\mathbf{z}_1^{(n+1)}, \cdot) &= \arg\min_{\mathbf{z}_1,\mathbf{P}_1} \mathcal{A}_\eta(\mathbf{w}^{(n+1)}, \mathbf{z}_1, \ldots, \mathbf{z}_K^{(n)}, P_1) \\ &\qquad \text{s.t. } \mathbf{z}_1(\mathbf{x}) = \mathbf{P}_1(\mathbf{x})\bar{\mathbf{x}},\ \forall \mathbf{x} \in \Omega \\ &\ \ \vdots \\ (\mathbf{z}_K^{(n+1)}, \cdot) &= \arg\min_{\mathbf{z}_K,\mathbf{P}_K} \mathcal{A}_\eta(\mathbf{w}^{(n+1)}, \mathbf{z}_1^{(n+1)}, \ldots, \mathbf{z}_K, \mathbf{P}_K) \\ &\qquad \text{s.t. } \mathbf{z}_K(\mathbf{x}) = \mathbf{P}_K(\mathbf{x}), \quad \forall \mathbf{x} \in \Omega \\ \boldsymbol{\mu}_1^{(n+1)}(\mathbf{x}) &= \boldsymbol{\mu}_1^{(i)}(\mathbf{x}) + \eta(\mathbf{w}^{(n+1)}(\mathbf{x}) - \mathbf{z}_1^{(n+1)}(\mathbf{x})), \quad \forall \mathbf{x} \in \Omega \\ &\ \ \vdots \\ \boldsymbol{\mu}_K^{(n+1)}(\mathbf{x}) &= \boldsymbol{\mu}_K^{(n)}(\mathbf{x}) + \eta(\mathbf{w}^{(n+1)}(\mathbf{x}) - \mathbf{z}_K^{(n+1)}(\mathbf{x})), \quad \forall \mathbf{x} \in \Omega. \end{aligned} \tag{8}$$

Observe that we only need the minimizing arguments with respect to the $\mathbf{z}_k$ variables, but not with respect to the parameter field. This is the reason why we omit the minimizer with respect to $\mathbf{P}_k$ on the left hand side of (8).

The accuracy and efficiency of our approach is based on our ability to solve exactly and at low computational cost each subproblem in (8). In Sections III-C and III-D, we detail our solvers for the update of $\mathbf{w}$ and $\mathbf{z}_k$, respectively.

C. Update of $\mathbf{w}$

The minimization with respect to $\mathbf{w}$ in (8) writes

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left(\rho(\mathbf{w}) + \frac{\eta}{2} \sum_{k=1}^{K} \sum_{\mathbf{x}\in\Omega} \left\| \mathbf{w}(\mathbf{x}) - \mathbf{z}_k(\mathbf{x}) + \frac{\boldsymbol{\mu}_k(\mathbf{x})}{\eta} \right\|_2^2 \right). \tag{9}$$

With simple manipulations, we rewrite (9) as

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \left(\rho(\mathbf{w}) + \frac{\eta K}{2} \sum_{\mathbf{x}\in\Omega} \|\mathbf{w}(\mathbf{x}) - \mathbf{r}(\mathbf{x})\|^2 \right), \tag{10}$$

where

$$\mathbf{r}(\mathbf{x}) = \frac{1}{K} \sum_{k=1}^{K} \left(\mathbf{z}_k(\mathbf{x}) - \frac{\boldsymbol{\mu}_k(\mathbf{x})}{\eta} \right). \tag{11}$$

Problem (10) is pointwise and admits a closed-form solution with the thresholding scheme

$$\hat{\mathbf{w}}(\mathbf{x}) = \mathbf{r}(\mathbf{x}) + \begin{cases} \dfrac{\nabla I}{\eta K}, & \rho_0(\mathbf{r}(\mathbf{x})) < -\dfrac{\|\nabla I\|_2^2}{\eta K} \\ -\dfrac{\nabla I}{\eta K}, & \rho_0(\mathbf{r}(\mathbf{x})) > \dfrac{\|\nabla I\|_2^2}{\eta K} \\ -\rho_0(\mathbf{r}(\mathbf{x})) \dfrac{\nabla I}{\|\nabla I\|_2^2}, & |\rho_0(\mathbf{r}(\mathbf{x}))| \le \dfrac{\|\nabla I\|_2^2}{\eta K}, \end{cases} \tag{12}$$

where $\nabla I = \nabla^\top I_2(\mathbf{x})$ and $\rho_0(\mathbf{r}(\mathbf{x})) = |\nabla I\, \mathbf{r}(\mathbf{x}) + I_t|$. A similar step appears in the context of a primal-dual optimization framework [10], [56].

Note that while we give here the solution for a data term derived from brightness constancy, the pointwise nature of the problem makes it tractable for other assumptions. For example, solutions for more sophisticated data fidelity terms based on normalized cross correlation or census transform are studied in [44].

D. Fast Update of $\mathbf{z}_k$

We address the minimization of the augmented Lagrangian with respect to $\mathbf{z}_k$ in the ADMM steps (8). It is instructive to first consider the case $\mathbf{z}_1$ which corresponds to the minimization in the vertical direction $\mathbf{d}_1 = (0, 1)$.

Our first step is to reduce the problem to a one-dimensional parameter estimation. To this end, we write the corresponding line in (8) as

$$\begin{aligned} (\hat{\mathbf{z}}_1, \cdot) = \; & \arg\min_{\mathbf{z},\mathbf{P}} \left(\kappa \|\nabla_{\mathbf{d}_1} \mathbf{P}\|_0 + \sum_{\mathbf{x}\in\Omega} \|\mathbf{v}(\mathbf{x}) - \mathbf{z}(\mathbf{x})\|_2^2 \right), \\ & \text{s.t. } \mathbf{P}(\mathbf{x})\bar{\mathbf{x}} = \mathbf{z}(\mathbf{x}), \end{aligned} \tag{13}$$

with $\mathbf{v}(\mathbf{x}) = \mathbf{w}(\mathbf{x}) + \frac{\boldsymbol{\mu}_k(\mathbf{x})}{\eta}$ and $\kappa = \frac{2\alpha_k \lambda}{\eta}$. Recall that $\mathbf{x} \in \Omega = \{1, \ldots, m\} \times \{1, \ldots, n\}$ and that $\mathbf{z} \in \mathbb{R}^2$ and $\mathbf{P} \in \mathbb{R}^{2\times 3}$. A crucial observation is that $\|\nabla_{\mathbf{d}_1} \mathbf{P}\|_0$ only takes into account neighborhood differences within the vertical scan lines. Therefore, the two-dimensional optimization problem (13) boils down to independent one-dimensional subproblems. Let us fix a vertical scan line by choosing a fixed index x_1; the corresponding 1D minimizer $\hat{\mathbf{z}}_1$ is then given by

$$\begin{aligned} & (\hat{\mathbf{z}}_1(x_1, \cdot), \cdot) \\ & = \arg\min_{\mathbf{z}',\mathbf{P}'} \left(\kappa \|\nabla_{\mathbf{d}_1} \mathbf{P}'\|_0 + \sum_{i=1}^{n} \left\| \mathbf{v}'(i) - \mathbf{z}'(i) \right\|_2^2 \right), \\ & \quad \text{s.t. } P'_{11}(i)x_1 + P'_{12}(i)i + P'_{13}(i) = z'_1(i) \\ & \qquad P'_{21}(i)x_1 + P'_{22}(i)i + P'_{23}(i) = z'_2(i), \end{aligned} \tag{14}$$

for all m, where $\mathbf{v}'(m) = \mathbf{v}(x_1, m)$, and where $\mathbf{z}'$ and $\mathbf{P}'$ are the flow field and parameter field on a one-dimensional line, respectively. As x_1 is fixed, the search space in P' can be reduced to parameter fields which are constant in the P'_{11} and P'_{21} component, say $P'_{11} = P'_{21} = 0$, without increasing the functional value. Let us denote such a reduced parameter field by $\mathbf{P}''$; i.e. $\mathbf{P}''(i) = (0, P'_{12}(i), P'_{13}(i); 0, P'_{22}(i), P'_{23}(i))$. The remaining four entries of the reduced parameter field are estimated via

$$\begin{aligned} \hat{\mathbf{P}}'' = \arg\min_{\mathbf{P}''} \Big(& \kappa \|\nabla_{\mathbf{d}_1} \mathbf{P}''\|_0 + \sum_{i=1}^{n} (v'_1(i) - P'_{12}(i)i - P'_{13}(i))^2 \\ & + (v'_2(i) - P'_{22}(i)i - P'_{23}(i))^2 \Big). \end{aligned} \tag{15}$$

The crucial point is that the problem (15) can be solved exactly and efficiently and that $\hat{\mathbf{z}}(x_1, \cdot)$ in (14) is recovered directly from the reduced parameter field $\hat{\mathbf{P}}''$ by

$$\begin{aligned} \hat{z}'_1(x_1, i) &= \hat{P}'_{12}(i)i + \hat{P}'_{13}(i) \\ \hat{z}'_2(x_1, i) &= \hat{P}'_{22}(i)i + \hat{P}'_{23}(i) \end{aligned} \tag{16}$$

without computing an optimal full parameter field $\hat{\mathbf{P}}'$ in (14).

We propose to solve problem (15) by dynamic programming. To that end, we cast it to a partitioning problem. We denote by $\mathcal{I}$ a partition of $\mathcal{N} = \{1, 2, \ldots, n\}$, so that $\mathcal{I}$ consists of subsets of $\mathcal{N}$ such that $\cup_{I\in\mathcal{I}} = \mathcal{N}$ and $I \cap J = \emptyset$ whenever $I \neq J$. Here, we additionally require that each $I \in \mathcal{I}$ is a "discrete interval"; that is, I is of the form $\{l, l+1, \ldots, r\}$. The minimum functional value in (14) is equal to the minimum value of the functional

$$B(\mathcal{I}) = \kappa(|\mathcal{I}| - 1) + \sum_{I\in\mathcal{I}} \sum_{t=1}^{2} \min_{a,b\in\mathbb{R}} \sum_{p\in I} (ap + b - v'_t(p))^2 \tag{17}$$

taken over all partitions $\mathcal{I}$ of $\mathcal{N}$. (Note that the sum over $t = 1, 2$ comes from expanding the Euclidean norm in $\mathbb{R}^2$.) From an optimal partition $\hat{\mathcal{I}}$ which minimizes B, the minimizer $\hat{\mathbf{P}}''$ of (15) can be obtained by letting $\hat{\mathbf{P}}''$ on $I \in \hat{\mathcal{I}}$ the (vectorial) affine linear parameters determined by

$$(\hat{P}'_{t2}, \hat{P}'_{t3}) = \arg\min_{a,b} \sum_{p\in I} (ap + b - v_t(x_1, p))^2, \quad \text{for } t = 1, 2. \tag{18}$$

It now remains to compute an optimal partition $\hat{\mathcal{I}}$ for problem (17). Our solver is based on the scheme presented in [15], [23], and [51] which we explain next. We denote the optimal functional value for data given on the domain $\{1, \ldots, r\}$ by

$$B^*_r = \min_{\mathcal{I} \text{ partition on } \{1,\ldots,r\}} B(\mathcal{I}). \tag{19}$$

It satisfies the Bellman equation

$$B_r^* = \min_{l=1,\dots,r} \left(B_{l-1}^* + \kappa + \sum_{t=1}^{2} \epsilon_{lrt} \right), \tag{20}$$

where we let $B_0^* = -\kappa$ and

$$\epsilon_{lrt} = \min_{a,b\in\mathbb{R}} \sum_{p=l}^{r} (ap + b - v_t(x_1, p))^2. \tag{21}$$

This reveals that B_r^* can be computed from B_{l-1}^* and ϵ_{lrt} for $l = 1, \dots, r$ and for $t = 1, 2$. By the dynamic programming principle, we successively compute B_1^*, B_2^*, until we reach B_n^*. As our primary interest is the partition $\mathcal{I}$ rather than the functional value, we keep track of a corresponding optimal partition. An economic way is to store at step r the minimizing argument l^* of (20); see [15] for a detailed description of that data structure. We further note that the ϵ_{lrt} in (21) can be computed in $\mathcal{O}(1)$ by precomputation of the moments of the data in (14); see Appendix A for a detailed description. The worst case complexity of this algorithm is $\mathcal{O}(n^2)$, where n is the number of elements in one line of the motion field. Thus, we get the complexity $\mathcal{O}(N^{3/2})$, where N denotes the number of pixels in the image. Since n lines can be processed simultaneously, the complexity is $\mathcal{O}(N)$ if n processors are available. To further accelerate the computations, we adopt the pruning strategy of [33].

So far, we have discussed the direction $\mathbf{d}_1$. For the directions $\mathbf{d}_2, \dots, \mathbf{d}_K$, we get intrinsically one-dimensional problems along the paths determined by the finite-difference vectors in a similar way. More precisely, we solve the one-dimensional problems of the form (14) linewise along vertical paths for $k = 2$ and along diagonal and antidiagonal paths for $k = 3, 4$, respectively. The 1D subproblems in vertical direction have length m. Meanwhile, those in the vertical direction have varying lengths, because the number of pixels in a diagonal direction depends on its offset.

IV. Experimental Results

A. Large Displacements Model

Modern evaluation benchmarks often include large displacements. To cope with them, we extend the model described in Section III by adopting the approach described in [6], [30], and [49]. It has become standard for variational motion estimation. This amounts to adding a term $\phi(\mathbf{w}, \mathbf{m})$ to the model (4), to promote similarity of the motion field $\mathbf{w}$ to the motion of a precomputed set of matched pixels $\mathbf{m} : \Lambda \subset \Omega \to \mathbb{R}^2$, defined on a sparse subset Λ of the image grid. This leads to

$$\begin{aligned} J = \min_{\mathbf{w},\mathbf{P}} \; & \rho(\mathbf{w}) + \gamma\, \phi(\mathbf{w}, \mathbf{m}) + \lambda \sum_{k=1}^{K} \alpha_k \|\nabla_{\mathbf{d}_k} \mathbf{P}\|_0 \\ & \text{s.t. } \mathbf{w}(\mathbf{x}) = \mathbf{P}(\mathbf{x})\bar{\mathbf{x}} \end{aligned} \tag{22}$$

for all $\mathbf{x} \in \Omega$, where $\gamma > 0$ is a balance parameter and $\phi(\mathbf{w}, \mathbf{m})$ is defined by

$$\phi(\mathbf{w}, \mathbf{m}) = \sum_{\mathbf{x}\in\Omega} c(\mathbf{x})\, \|\mathbf{w}(\mathbf{x}) - \mathbf{m}(\mathbf{x})\|_1, \tag{23}$$

TABLE I
Comparison of AEP on Benchmarks

MPI Sintel		
Method	AEP Training set	AEP Test set
Our method	**2.27**	**1.35**
Ours-TV	2.53	-
DataFlow [44]	5.16	-
DeepFlow [49]	3.04	1.77
Classic+NL [38]	5.22	3.77
PH-Flow [55]	-	1.71
FC-2Layers-FF [40]	-	3.05
Kitti		
Method	AEP Training set	AEP Test set
Our method	**1.29**	1.5
Ours-TV	1.56	-
DataFlow [44]	1.36	1.9
DeepFlow [49]	1.45	1.5
Classic+NL [38]	2.61	2.8
PH-Flow [55]	-	**1.3**
NLTGV-Census [29]	-	1.6

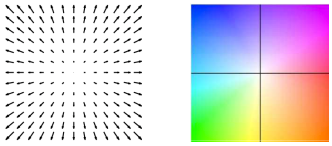

Fig. 1. Equivalence between vector (left) and color (right) representations of displacement vectors.

where c is the indicator function of Λ defined by

$$c(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in \Lambda \\ 0, & \text{else.} \end{cases} \tag{24}$$

We compute the matches with the method described in [49], using the public code of the authors.[1] This new term has almost no impact on the computational cost in the optimization framework described in Section III. We introduce an additional splitting variable associated to ϕ, which generates a new subproblem that is solved directly, like in Section III-C. We give the detailed minimization steps in Appendix B.

Finally, to overcome the restriction of small displacements of the linearized data term (6), the estimation is integrated in a standard coarse-to-fine scheme [5].

B. Evaluation Datasets

We validate our method on two reference benchmarks for motion estimation.

The *MPI Sintel* benchmark is composed of sequences extracted from a realistic animated movie. It contains 1 064 training sequences with available ground truth and 564 test sequences

[1] http://lear.inrialpes.fr/src/deepmatching/

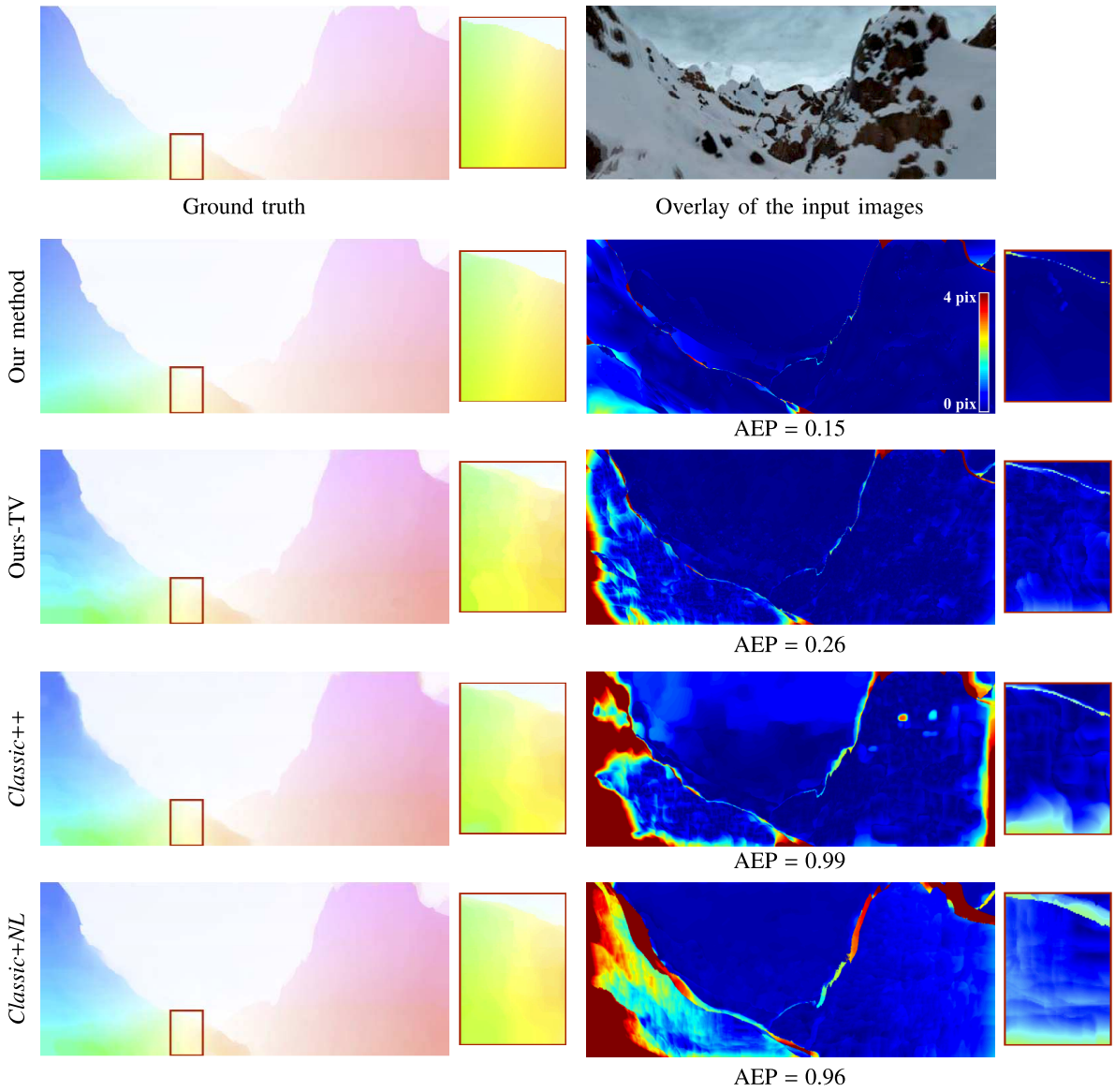

Fig. 2. Comparison with the TV motion fields (left column) and their associated error maps (right column). We provide a zoomed cutout of the motion field in each case. The error map shows the euclidean distance between the estimated motion vector and the ground truth at each pixel, represented with the colorbar displayed on the second row.

used for blind evaluation [7]. Each sequence has a *final* and a *clean* version. The *final* version introduces perturbations such as motion blur, defocus, or atmospheric fog, which are not present in the *clean* version. These effects are handled by the data term or by specific estimation strategies. To focus on the evaluation of the regularization, we used the *clean* dataset in our experiments.

The *Kitti* benchmark [16] is composed of 193 training sequences and 193 test sequences, acquired in real outdoor conditions on a platform installed on a moving car. A ground truth is provided only for half of the pixels. This benchmark is characterized by large illumination changes.

We compute the estimation accuracy with the endpoint error, defined at each pixel as the Euclidean distance between the estimated motion vector and the ground truth. We report the averaged endpoint error (AEP) on the whole image. Estimating motion in occluded regions is an important problem that is beyond the scope of this paper [14], [22], [52], [53]. Therefore, we compute the errors only in non-occluded regions.

The results on the two benchmarks are presented in Table I, on the *training* and *test* sets. We consider methods with public codes for the training set, and the ones with published results for the test set. Therefore, some methods are not present in both categories. We did not report the result of *DataFlow* for the *test* set of *MPI Sintel* since the published results have been obtained without the large displacement extension of Section IV-A, which is decisive to obtain comparable results.

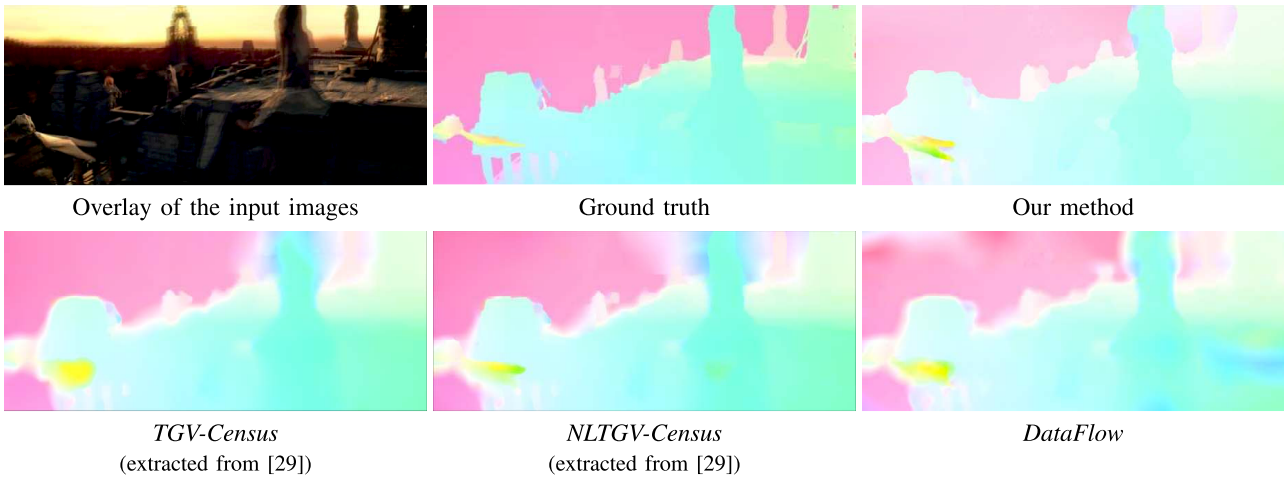

Fig. 3. Comparison with the TGV motion fields. The visualizations of *TGV-Census* and *NLTGV-Census* are taken from [29]. The input images are from the *final* version of the *MPI Sintel* dataset.

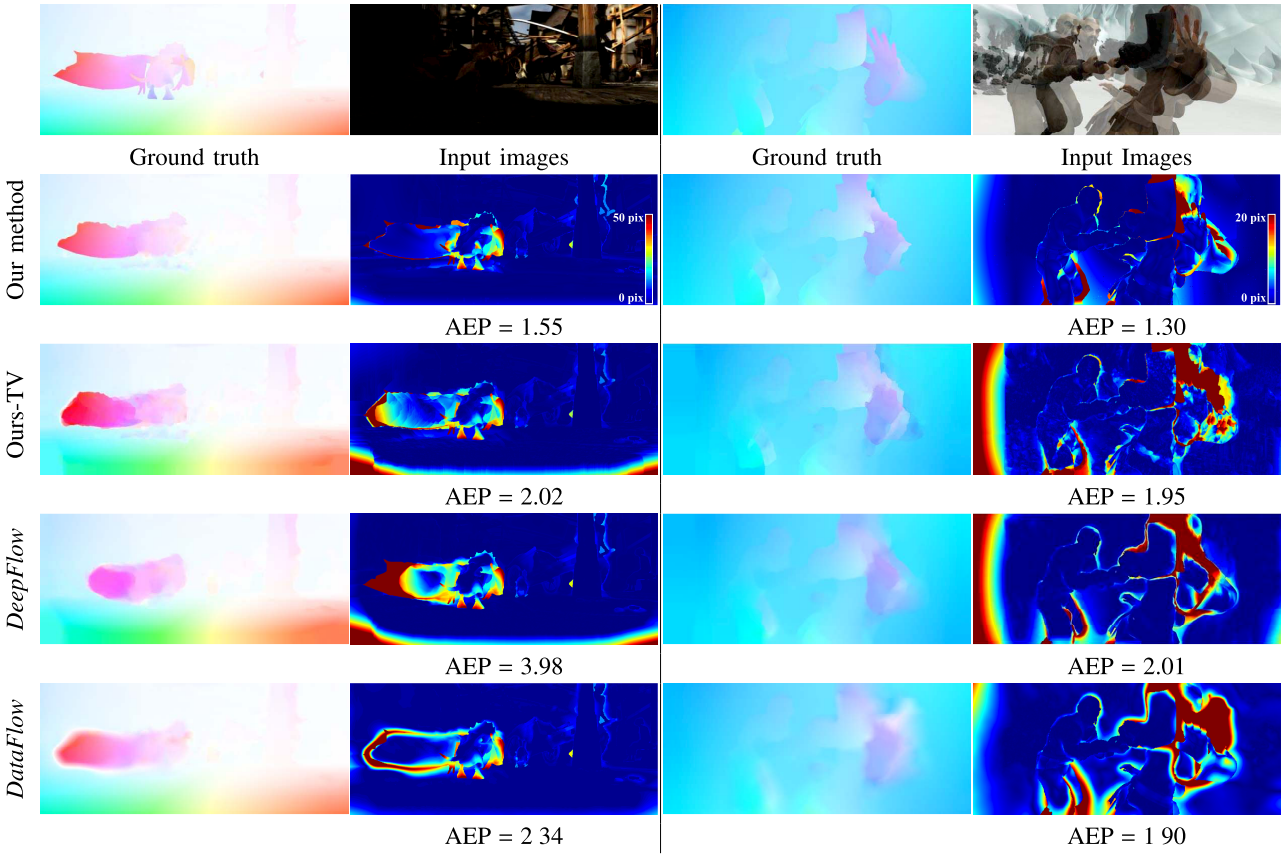

Fig. 4. Comparison of the motion fields (columns 1 and 3) and their associate error maps (columns 2 and 4) on the sequences *market_4* (columns 1 and 2) and *ambush_2* (columns 3 and 4) of the *MPI Sintel* benchmark. The error map shows the euclidean distance between the estimated motion vector and the ground truth at each pixel, represented with the colorbar displayed on the second row.

C. Visualization of Motion Fields

We display motion fields with the standard color code defined in Fig. 1. The hue indicates the direction of the displacement, and the saturation indicates the amplitude of the displacement.

D. Implementation Details

The parameters are optimized on a subset of 30% of the *training* data set, both on *MPI Sintel* and *Kitti*. The results of Table I are obtained on the rest of the sequences. To accelerate convergence, we increase the value of η at each iteration in (8).

We start with the initial value $\eta^{(0)} = 0.01$ and we define the sequence $(\eta^{(i)})_{i\in\mathbb{N}}$ by a geometric evolution $\eta^{(i+1)} = \tau\,\eta^{(i)}$ with $\tau = 1.1$. As pre-processing, we apply Gaussian filtering with a variance of 0.9 to the input images to reduce the influence of noise. We apply a weighted median filter as a post-processing at each scale of the coarse-to-fine scheme to remove outliers. The scale factor of the coarse-to-fine pyramid is set to 0.75.

The algorithm has been implemented in MATLAB, with a C++ implementation for the dynamic programming solver. The 1D piecewise-affine denoising subproblem (Section III-D), which consumes most of the computational time, is naturally parallelizable. The reported runtime results have been obtained with a parallelization on 4 cores, on an Intel Xeon CPU 2.6GHz.

E. Comparison Methods

We want to focus on regularization while validating our piecewise-affine model. We consider competing methods that are as close as possible to ours and compare our method with 1) the usual TV and TGV regularizations, and 2) other piecewise-parametric approaches.

1) TV-Based Methods: The method named *Classic++* is described in [38]. It uses the same data term as ours, with an anisotropic TV regularization but without the features described in Section IV-A.

The *DeepFlow* method [49] differs from our formulation of Section IV-A by an isotropic TV regularization instead of our piecewise-affine constraint and by a gradient conservation in addition to the intensity conservation (6).

Finally, we also consider the nonlocal extension of TV described in [38] and named *Classic+NL*.

To demonstrate the importance of the piecewise-affine model compared to TV regularization, we create a method that we term Ours-TV by replacing the one-dimensional piecewise-affine constraint in (22) by a one-dimensional TV regularization. This leads to the optimization problem

$$J = \min_{\mathbf{w}} \left(\rho(\mathbf{w}) + \gamma\,\phi(\mathbf{w},\mathbf{m}) + \lambda \sum_{k=1}^{K} \alpha_k\, TV_k(\mathbf{w}) \right), \quad (25)$$

where TV_k applies TV regularization in the kth direction. The minimization framework remains unchanged, except for the subproblems with respect to $\mathbf{z}_k$ in (8). They become TV-ℓ_2 denoising problems, efficiently solvable with the taut-string algorithm [12].

2) TGV-Based Methods: The second-order TGV regularization generalizes the TV approach and imposes a piecewise-affine form by introducing penalization of second derivatives with an ℓ_1 norm [4]. We consider the method *DataFlow* described in [44]. It uses TGV regularization with several advanced data terms. In our experiments, we used the sum of absolute differences, which is a patch-based version of brightness constancy. We integrated ϕ from (22) in *DataFlow*, taking advantage of the public code provided by the authors.

We also performed comparisons with the nonlocal version of TGV proposed in [29], termed *NL-TGV*. However, the data term in *NL-TGV* is different from ours. It is based on the census transform, which provides invariance to illumination changes.

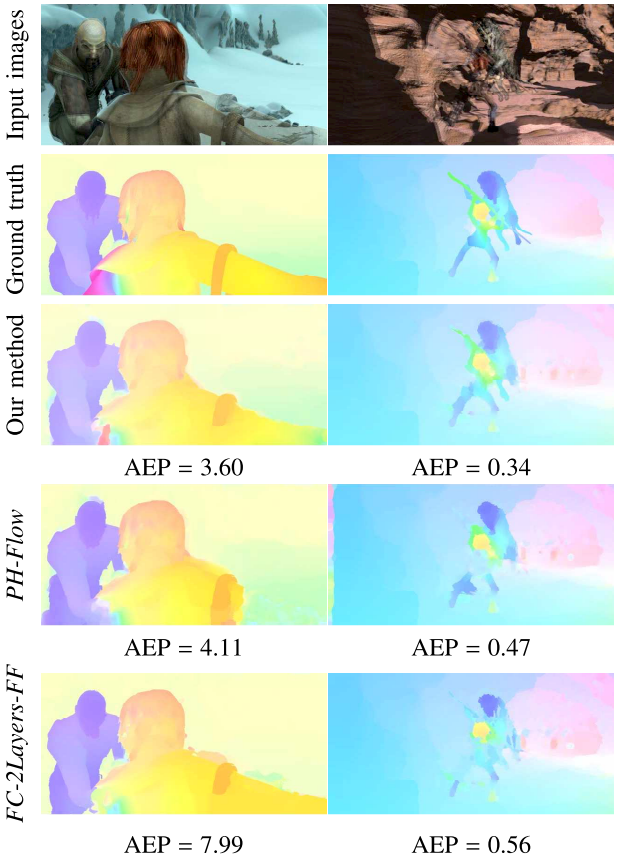

Fig. 5. Comparisons between the results of our method, *PH-Flow* [55], and *FC-2Layers-FF* [40] on images from the *test* sequence of the *MPI Sintel* benchmark.

3) Piecewise-Parametric Methods: The method termed *PH-Flow* estimates a piecewise homography model and is based on the formulation (1) with inter-piece regularization and graph cut optimization [55].

We also perform comparisons with the method *FC-2Layers-FF* [40], which is based on a layered representation and is not purely piecewise-parametric but allows deviations from an affine model in each of the segmented region.

We used the publicly available codes for *Classic++* and *Classic+NL*,[2] *DeepFlow*,[3] and *DataFlow*.[4]

F. TV and TGV Regularization

In Figure 2, we compare our result with methods based on TV regularization, namely, Ours-TV, *Classic++*, and its nonlocal variant *Classic+NL*, in the case of smooth variations of the motion field and small displacements. We display the estimated motion field and the endpoint error maps. The TV regularization produces typical staircasing artifacts due to the

[2]http://people.seas.harvard.edu/ dqsun/
[3]http://lear.inrialpes.fr/src/deepflow/
[4]http://github.com/vogechri/DataFlow/

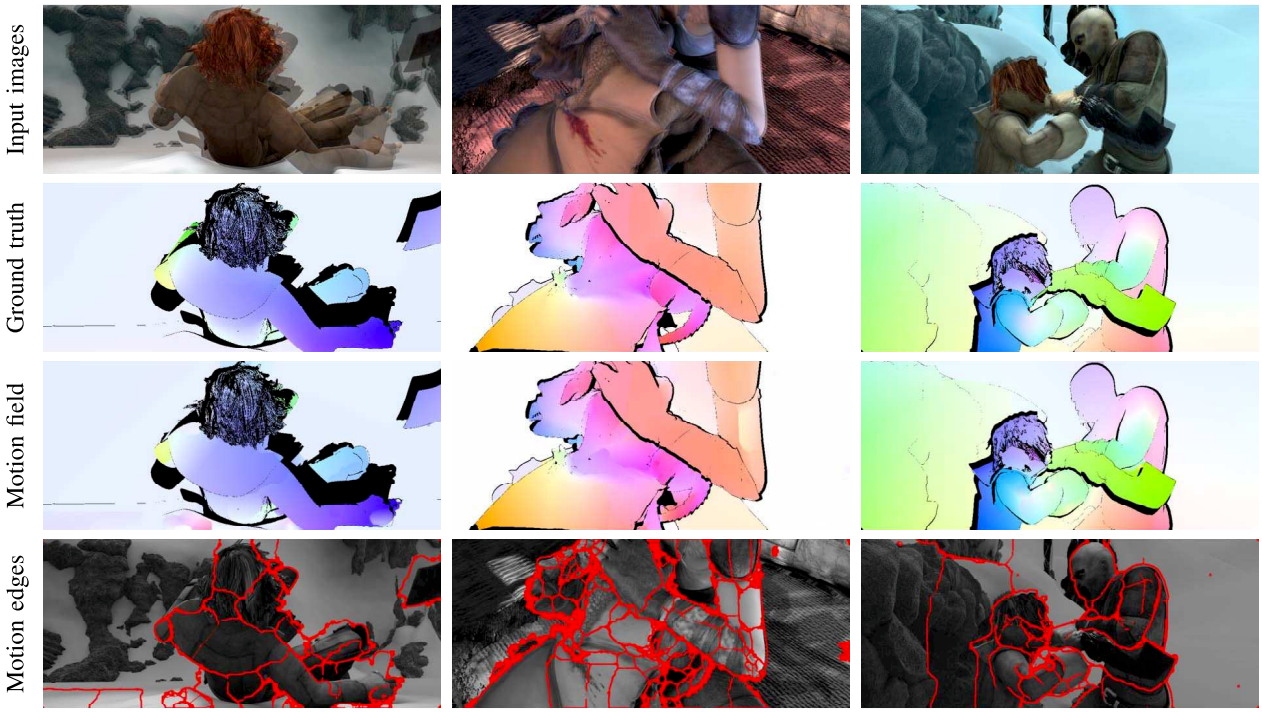

Fig. 6. Reconstruction of motion edges (last row) from the motion field estimated with our method (third row) on examples of the *MPI Sintel* dataset. Black pixels in the motion fields represent occluded region.

piecewise constancy of the solution. This effect is emphasized in the cutouts of Figure 2. Our piecewise-affine approach does not produce staircasing and is much closer to the ground truth, both visually and in terms of AEP. We also observe that the motion discontinuities are more accurately recovered with our approach.

In Figure 3, we compare our method with the methods *TGV-Census*, *NLTGV-Census* and *DataFlow*, which are based on TGV regularization. The absence of staircasing of TGV comes at the price of some blurring artifacts in the result. Even the nonlocal approach *NLTGV-Census*, which is specifically designed to reduce blurring, cannot solve completely the problem. In contrast, our method combines a good restitution of affine displacements with a satisfactory recovery of sharp motion discontinuities.

In Figure 4, we compare the results of our method, Ours-TV, $DeepFlow$ and $DataFlow$. We recall that the essential difference between these methods is only the regularization strategy. We observe that the sharpness of discontinuities is always better preserved in our results compared to $DeepFlow$ and $DataFlow$. Generally, the global shapes of moving objects are more accurately delineated by our method. We also observe staircasing artifacts in the results of Ours-TV. Altogether, the best AEP is achieved by our method. Note that large errors at image borders are due to occlusions and are not taken into account in the computation of the AEP.

These qualitative observations are confirmed by the better results of our approach in Table I compared to the methods with a similar framework but different regularization: Ours-TV, *DeepFlow*, *DataFlow*, *Classic+NL*, and *NLTGV-Census*.

G. Piecewise-Parametric Methods

1) Accuracy: In Figure 5, we show visual comparisons between our method, *PH-Flow*, and *FC-2Layers-FF*. On these examples, our method is able to retrieve more details and to delineate motion discontinuities more accurately. The numerical results of Table I show a clear advantage on *MPI Sintel*. On the *Kitti* benchmark, our method is close to the best performing method *PH-Flow*.

2) Initialization and Post-Processing: Our method is initialization-free, contrarily to all other piecewise-parametric motion estimation methods, as described in Section II. It is also refinement-free, while the results of *PH-Flow* are obtained after refinement with the method *Classic+NL* dedicated to small displacements.

3) Runtime: Our method needs 135s to compute a motion field on a pair of 1241×376 images of the Kitti benchmark, while the runtime of *PH-FLow* reported on the Kitti website is 800s for the same image pair. *FC-2Layers-FF* is not present in the Kitti benchmark but reports a rutime of 40 minutes for a pair of 600×400 images of the Middlebury benchmark. Another major advantage of our method is that the runtime is completely independent of the complexity of the motion field. On the contrary, the computational cost of the segmentation performed by *PH-Flow* and *FC-2Layers-FF* depends highly on the number of regions, which can be very high for complex motion fields. Also note that we estimate motion on non-downsampled images, while *PH-Flow* computes piecewise parametric motion on images downsampled by a factor of 2 before refinement.

[4]We reproduce the images publicly available on the website of the benchmark. We also give the AEP associated to each motion field.

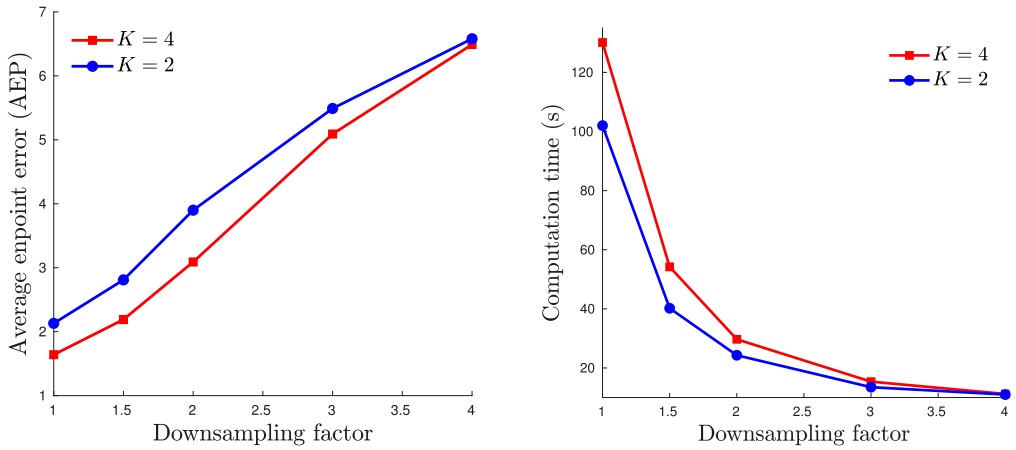

Fig. 7. Plot of the AEP (left) and computation time (right) with respect to the downsampling factor applied to the input images, and for two different values of K.

H. Reconstitution of Piecewise-Affine Edges

To illustrate the piecewise-affine form of the estimated motion fields, we show in Figure 6 a reconstruction of motion edges obtained by thresholding the magnitude of spatial derivatives of the motion field. When the scene is composed of a few moving parts undergoing simple deformations, the image domain is divided in a few meaningful regions. When the motion is more complex, our method decomposes the motion field in smaller pieces. A crucial aspect of our method is that this increasing complexity has no effect on the computational cost.

I. Tradeoff Between Computation Time and Accuracy

For specific applications, a compromise may be required between accuracy and computation time. To analyze the impact of image downsampling on this issue, we report in Fig. 7 the AEP and the computation time obtained on the first frame of the KITTI dataset, for different downsampling factors applied to the input images. The computed motion fields are upsampled with bicubic interpolation to the original image size and are post-processed with a weighted median filter. We also show the effect of the number of directions K in the discretization of the gradient (4). We compare the results with $K = 4$ (which is our choice in all previous experiments) and $K = 2$.

We observe that the error increases almost linearly, while the decrease of the computation time is very fast for small downsampling factors. Similarly, the error is slightly higher when $K = 2$ compared to $K = 4$, but the computation time is significantly lower for small downsampling factors. These elements could be taken into account for the choice of a tradeoff between accuracy and computation time.

V. Conclusion

We have proposed a new method to estimate piecewise-affine motion fields. In contrast to related methods, our approach does not rely on explicit segmentation but directly estimates a piecewise-affine motion field. Key steps in the derivation are the specific formulation of the energy functional as a constrained optimization problem and the decomposition into tractable subproblems by an alternatig direction method of multipliers strategy. Then, these subproblems are cast to (non-convex) univariate piecewise-affine problems. A crucial ingredient of our method is that we are able to solve them exactly and efficiently. Our method overcomes the two main limitations of previous piecewise-parametric approaches, namely, sensitivity to initialization and computational cost. Yet, our experiments show that it is competitive in terms of quality. Further, they suggest that the piecewise affine model improves upon total variation and total generalized variation regularizations when using similar data terms. The versatility of our proximal splitting strategy lets extensions of the method to new data terms be easily implemented. Thus, the proposed approach can serve as a general regularization framework for motion estimation.

Appendix A
Calculation of the Approximation Errors

We describe how to efficiently compute the approximation errors ϵ_{lrt} required in (20). Let $g \in \mathbb{R}^{n\times 2}$. (This corresponds to $g_{pt} = v_t(x_1, p)$. for all p, t in (20).) Taking the derivative of the right-hand side of (21) with respect to a, b yields the optimality conditions

$$\begin{aligned} \sum_{p=l}^{r} w_p(a_{lrt}\, p + b_{lrt} - g_{pt})p &= 0 \\ \sum_{p=l}^{r} w_p(a_{lrt}\, p + b_{lrt} - g_{pt}) &= 0. \end{aligned} \tag{26}$$

This linear system can be rewritten as

$$\begin{aligned} a_k E_{lr} + b_k G_{lr} &= I_{lrt}, \\ a_k G_{lr} + b_k H_{lr} &= J_{lrt}, \end{aligned} \tag{27}$$

with the auxiliary quantities

$$E_{lr} = \sum_{p=l}^{r} w_p p^2, \quad G_{lr} = \sum_{p=l}^{r} w_p p, \quad H_{lr} = \sum_{p=l}^{r} w_p, \tag{28}$$

$$I_{lrk} = \sum_{p=l}^{r} w_p g_{pt}\, p, \quad J_{lrk} = \sum_{p=l}^{r} w_p g_{pt}. \tag{29}$$

The solutions a^*_{lrk} and b^*_{lrk} are given by

$$a^*_{lrt} = \frac{I_{lr} H_{lr} - G_{lr} J_{lrk}}{E_{lr} H_{lr} - G^2_{lr}} \tag{30}$$

and

$$b^*_{lrt} = \frac{E_{lr} J_{lr} - I_{lrk} G_{lr}}{E_{lr} H_{lr} - G^2_{lr}}. \tag{31}$$

Plugging this into (21) gives us

$$\epsilon_{lrk} = \frac{J^2_{lr} E_{lr} - 2 G_{lr} J_{lrt} I_{lrt} + H_{lr} I^2_{lrt} + G^2_{lr} K_{lrt} - H_{lr} E_{lr} K_{lrt}}{G^2_{lr} - H_{lr} E_{lr}} \tag{32}$$

where $K_{lrt} = \sum_{p=l}^{r} w_p g^2_{pt}$. Note that the involved sums can be computed efficiently by utilizing precomputations of moments. For example, E_{lr} can be computed via $E_{lr} = E'_r - E'_{l-1}$ where $E'_t = \sum_{p=1}^{t} w_p p^2$. So E_{lr} can be computed in $O(1)$ if the vector E' is precomputed. E' in turn can be computed in $O(n)$. For the other summations, analogous schemes are applied.

Appendix B
Optimization for the Large Displacement Model

To solve the minimization problem (22) for the extended model, we follow the splitting scheme used for (7). We introduce an additional splitting variable associated to the term ϕ. Accordingly, (22) rewrites

$$\begin{aligned} J = \min_{\mathbf{w},\mathbf{u},\mathbf{P}_1,\ldots,\mathbf{P}_K} & \left(\rho(\mathbf{w}) + \gamma\ \phi(\mathbf{u},\mathbf{m}) + \lambda \sum_{k=1}^{K} \alpha_k \|\nabla_{d_k} \mathbf{P}_k\|_0 \right) \\ & \text{s.t. } \mathbf{w}(\mathbf{x}) = \mathbf{z}_k(\mathbf{x}), \\ & \quad \mathbf{u}(\mathbf{x}) = \mathbf{w}(\mathbf{x}), \\ & \quad \mathbf{z}_k(\mathbf{x}) = \mathbf{P}_k(\mathbf{x})\bar{\mathbf{x}}, \quad \forall \mathbf{x} \in \Omega,\ \forall k = 1,\ldots,K. \end{aligned} \tag{33}$$

The augmented Lagrangian associated to (33) is then

$$\begin{aligned} & \mathcal{A}_{\eta_1,\eta_2}(\mathbf{w},\mathbf{u},\{\mathbf{P}_k\}_k,\{\mathbf{z}_k\}_k,\{\boldsymbol{\mu}_k\}_k,\boldsymbol{\xi}) \\ & = \rho(\mathbf{w}) + \gamma\ \phi(\mathbf{u},\mathbf{m}) + \lambda \sum_{k=1}^{K} \alpha_k \|\nabla_{\mathbf{d}_k} \mathbf{P}_k\|_0 \\ & \quad + \frac{\eta_1}{2} \sum_{k=1}^{K} \sum_{\mathbf{x}\in\Omega} \left\| \mathbf{w}(\mathbf{x}) - \mathbf{z}_k(\mathbf{x}) + \frac{\boldsymbol{\mu}_k(\mathbf{x})}{\eta_1} \right\|^2_2 - \frac{1}{2\eta_1} \left\| \boldsymbol{\mu}_k(\mathbf{x}) \right\|^2_2 \\ & \quad + \frac{\eta_2}{2} \sum_{\mathbf{x}\in\Omega} \left\| \mathbf{u}(\mathbf{x}) - \mathbf{w}(\mathbf{x}) + \frac{\boldsymbol{\xi}(\mathbf{x})}{\eta_2} \right\|^2_2 - \frac{1}{2\eta_2} \|\boldsymbol{\xi}(\mathbf{x})\|^2_2 \\ & \quad \text{s.t. } \mathbf{z}_k(\mathbf{x}) = \mathbf{P}_k(\mathbf{x})\bar{\mathbf{x}}, \quad \forall \mathbf{x} \in \Omega,\ \forall k \in \{1,\ldots,K\}. \end{aligned} \tag{34}$$

Similarly to the update scheme (8), the ADMM steps involve minimizing $\mathcal{A}_{\eta_1,\eta_2}(\mathbf{w},\mathbf{u},\{\mathbf{P}_k\}_k,\{\mathbf{z}_k\}_k,\{\boldsymbol{\mu}_k\}_k,\boldsymbol{\xi})$ with respect to $\mathbf{w}$, $\mathbf{z}_k$, and $\mathbf{u}$. The minimization problem with respect to $\mathbf{z}_k$ is the same as in Section III-D. We detail now the updates of $\mathbf{w}$ and $\mathbf{u}$, which are very similar to the description of Section III-C.

A. Update of $\mathbf{w}$

The minimization w.r.t u in can be rewritten

$$\min_{\mathbf{w}} \rho(\mathbf{w}) + \frac{\eta_1 K + \eta_2}{2} \sum_{\mathbf{x}\in\Omega} (\mathbf{w}(\mathbf{x}) - \mathbf{t}(\mathbf{x}))^2, \tag{35}$$

where

$$\mathbf{t}(\mathbf{x}) = \frac{1}{\eta_1\ K + \eta_2} \left(\eta_1\ K\ \mathbf{r}(\mathbf{x}) + \eta_2 \left(\mathbf{u}(\mathbf{x}) + \frac{\boldsymbol{\xi}(\mathbf{x})}{\eta_2} \right) \right). \tag{36}$$

The problem is pointwise and admits a closed-form solution with the thresholding scheme

$$\mathbf{w}(\mathbf{x}) = \mathbf{t}(\mathbf{x}) + \begin{cases} \dfrac{\nabla I}{\eta_1\ K}, & \rho_0(\mathbf{t}(\mathbf{x})) < -\dfrac{\|\nabla I\|^2_2}{\eta_1\ K} \\ -\dfrac{\nabla I}{\eta_1\ K}, & \rho_0(\mathbf{t}(\mathbf{x})) > \dfrac{\|\nabla I\|^2_2}{\eta_1\ K} \\ -\rho_0(\mathbf{t}(\mathbf{x})) \dfrac{\nabla I}{\|\nabla I\|^2_2}, & |\rho_0(\mathbf{t}(\mathbf{x}))| \le \dfrac{\|\nabla I\|^2_2}{\eta_1 K}. \end{cases} \tag{37}$$

B. Update of $\mathbf{u}$

The minimization w.r.t u in writes

$$J = \min_{\mathbf{u}} \left(\gamma\, \phi(\mathbf{u},\mathbf{m}) + \frac{\eta_2}{2} \sum_{\mathbf{x}\in\Omega} \|\mathbf{u}(\mathbf{x}) - \mathbf{v}(\mathbf{x})\|^2 \right). \tag{38}$$

where $\mathbf{v}(\mathbf{x}) = \mathbf{w}(\mathbf{x}) - \frac{\boldsymbol{\xi}(\mathbf{x})}{\eta_2}$.

The problem is pointwise and admits a closed-form solution with the thresholding scheme

$$u_k(\mathbf{x}) = \begin{cases} v_k(\mathbf{x}), & c(\mathbf{x}) = 0, \\ v_k(\mathbf{x}) + \dfrac{\gamma}{\eta_2}, & v_k(\mathbf{x}) - m_k(\mathbf{x}) < -\dfrac{\gamma}{\eta_2} \quad \text{and} \quad c(\mathbf{x}) \neq 0 \\ v_k(\mathbf{x}) - \dfrac{\gamma}{\eta_2}, & v_k(\mathbf{x}) - m_k(\mathbf{x}) > \dfrac{\gamma}{\eta_2} \quad \text{and} \quad c(\mathbf{x}) \neq 0 \\ m_k(\mathbf{x}), & |v_k(\mathbf{x}) - m_k(\mathbf{x})| \le \dfrac{\gamma}{\eta_2} \quad \text{and} \quad c(\mathbf{x}) \neq 0. \end{cases} \tag{39}$$

where $k = \{1,2\}$ and we use the notations $\mathbf{u}(\mathbf{x}) = (u_1(\mathbf{x}), u_2(\mathbf{x}))$, $\mathbf{v}(\mathbf{x}) = (v_1(\mathbf{x}), v_2(\mathbf{x}))$, and $\mathbf{m}(\mathbf{x}) = (m_1(\mathbf{x}), m_2(\mathbf{x}))$.

References

[1] A. Blake and A. Zisserman, *Visual Reconstruction*. Cambridge, MA, USA: MIT Press, 1987.

[2] M. Bleyer, C. Rhemann, and M. Gelautz, "Segmentation-based motion with occlusions using graph-cut optimization," in *Proc. Joint Pattern Recognit. Symp.*, Berlin, Germany, Sep. 2006, pp. 465–474.

[3] P. Bouthemy and E. Francois, "Motion segmentation and qualitative dynamic scene analysis from an image sequence," *Int. J. Comput. Vis.*, vol. 10, no. 2, pp. 157–182, 1993.

[4] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation," *SIAM J. Imag. Sci.*, vol. 3, no. 3, pp. 492–526, 2010.

[5] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High accuracy optical flow estimation based on a theory for warping," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Prague, Czech Republic, 2004, pp. 25–36.

[6] T. Brox and J. Malik, "Large displacement optical flow: Descriptor matching in variational motion estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 3, pp. 500–513, Mar. 2011.

[7] D. J. Butler, J. Wulff, G. Stanley, and M. Black, "A naturalistic open source movie for optical flow evaluation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Florence, Italy: Springer-Verlag, 2012, pp. 611–625.

[8] X. Cai, J. H. Fitschen, M. Nikolova, G. Steidl, and M. Storath, "Disparity and optical flow partitioning using extended Potts priors," *Inf. Inference*, vol. 4, no. 1, pp. 43–62, 2015.

[9] A. Chambolle, "Finite-differences discretizations of the Mumford–Shah functional," *ESAIM, Math. Model. Numer. Anal.*, vol. 33, no. 2, pp. 261–288, 1999.

[10] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *J. Math. Imag. Vis.*, vol. 40, no. 1, pp. 120–145, 2011.

[11] R. Chartrand and B. Wohlberg, "A nonconvex ADMM algorithm for group sparsity with sparse groups," in *Proc. Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2013, pp. 6009–6013.

[12] L. Condat, "A direct algorithm for 1-D total variation denoising," *IEEE Signal Process. Lett.*, vol. 20, no. 11, pp. 1054–1057, Nov. 2013.

[13] D. Cremers and S. Soatto, "Motion competition: A variational approach to piecewise parametric motion segmentation," *Int. J. Comput. Vis.*, vol. 62, no. 3, pp. 249–265, 2005.

[14] D. Fortun, P. Bouthemy, and C. Kervrann, "Aggregation of local parametric candidates with exemplar-based occlusion handling for optical flow," *Comput. Vis. Image Understand.*, vol. 145, pp. 81–94, Apr. 2016.

[15] F. Friedrich, A. Kempe, V. Liebscher, and G. Winkler, "Complexity penalized M-estimation," *J. Comput. Graph. Statist.*, vol. 17, no. 1, pp. 201–224, 2008.

[16] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 3354–3361.

[17] R. Giryes, M. Elad, and A. M. Bruckstein, "Sparsity based methods for overparameterized variational problems," *SIAM J. Imag. Sci.*, vol. 8, no. 3, pp. 2133–2159, 2015.

[18] K. Hohm, M. Storath, and A. Weinmann, "An algorithmic framework for Mumford–Shah regularization of inverse problems in imaging," *Inverse Problems*, vol. 31, no. 11, p. 115011, 2015.

[19] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, nos. 1–3, pp. 185–203, Aug. 1981.

[20] M. Hornáv cek, F. Besse, J. Kautz, A. Fitzgibbon, and C. Rother, "Highly overparameterized optical flow using patchmatch belief propagation," in *Proc. Eur. Conf. Comput. Vis.*, Zürich, Switzerland, 2014, pp. 220–234.

[21] J. Hur and S. Roth, "Joint optical flow and temporally consistent semantic segmentation," in *Proc. ECCV*, 2016, pp. 163–177.

[22] S. Ince and J. Konrad, "Occlusion-aware optical flow estimation," *IEEE Trans. Image Process.*, vol. 17, no. 8, pp. 1443–1451, Aug. 2008.

[23] J. Kleinberg and E. Tardos, *Algorithm Design*. London, U.K.: Pearson, 2006.

[24] E. Mémin and P. Pérez, "Hierarchical estimation and segmentation of dense motion fields," *Int. J. Comput. Vis.*, vol. 46, no. 2, pp. 129–155, 2002.

[25] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Commun. Pure Appl. Math.*, vol. 42, no. 5, pp. 577–685, 1989.

[26] T. Nir, A. M. Bruckstein, and R. Kimmel, "Over-parameterized variational optical flow," *Int. J. Comput. Vis.*, vol. 76, no. 2, pp. 205–216, 2008.

[27] J.-M. Odobez and P. Bouthemy, "Direct incremental model-based image motion segmentation for video analysis," *Signal Process.*, vol. 66, no. 2, pp. 143–155, 1998.

[28] N. Paragios and R. Deriche, "Geodesic active regions and level set methods for motion estimation and tracking," *Comput. Vis. Image Understand.*, vol. 97, no. 3, pp. 259–282, 2005.

[29] R. Ranftl, K. Bredies, and T. Pock, "Non-local total generalized variation for optical flow estimation," in *Proc. Eur. Conf. Comput. Vis.*, Zürich, Switzerland, 2014, pp. 439–454.

[30] J. Revaud, P. Weinzaepfel, C. Harchaoui, and Z. Schmid, "EpicFlow: Edge-preserving interpolation of correspondences for optical flow," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, May 2015, pp. 1164–1172.

[31] T. Schoenemann and D. Cremers, "Near real-time motion segmentation using graph cuts," in *Proc. Joint Pattern Recognit. Symp.*, 2006, pp. 455–464.

[32] L. Sevilla-Lara, D. Sun, V. Jampani, and M. J. Black, "Optical flow with semantic segmentation and localized layers," in *Proc. CVPR*, 2016, pp. 3889–3898.

[33] M. Storath and A. Weinmann, "Fast partitioning of vector-valued images," *SIAM J. Imag. Sci.*, vol. 7, no. 3, pp. 1826–1852, 2014.

[34] M. Storath, A. Weinmann, and L. Demaret, "Jump-sparse and sparse recovery using Potts functionals," *IEEE Trans. Signal Process.*, vol. 62, no. 14, pp. 3654–3666, Jul. 2014.

[35] M. Storath, A. Weinmann, J. Frikel, and M. Unser, "Joint image reconstruction and segmentation using the Potts model," *Inverse Problems*, vol. 31, no. 2, p. 025003, 2015.

[36] D. Sun, S. Roth, and M. J. Black, "Secrets of optical flow estimation and their principles," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2432–2439.

[37] D. Sun, E. B. Sudderth, and M. J. Black, "Layered image motion with explicit occlusions, temporal consistency, and depth ordering," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Vancouver, BC, Canada, 2010, pp. 2226–2234.

[38] D. Sun, S. Roth, and M. J. Black, "A quantitative analysis of current practices in optical flow estimation and the principles behind them," *Int. J. Comput. Vis.*, vol. 106, no. 2, pp. 115–137, 2014.

[39] D. Sun, E. B. Sudderth, and M. J. Black, "Layered segmentation and optical flow estimation over time," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 1768–1775.

[40] D. Sun, J. Wulff, E. B. Sudderth, H. Pfister, and M. J. Black, "A fully-connected layered model of foreground and background flow," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 2451–2458.

[41] W. Trobin, T. Pock, D. Cremers, and H. Bischof, "An unbiased second-order prior for high-accuracy motion estimation," in *Proc. Joint Pattern Recognit. Symp.*, 2008, pp. 396–405.

[42] M. Unger, M. Werlberger, T. Pock, and H. Bischof, "Joint motion estimation and segmentation of complex scenes with label costs and occlusion modeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 1878–1885.

[43] C. Vazquez, A. Mitiche, and R. Laganiere, "Joint multiregion segmentation and parametric estimation of image motion by basis function representation and level set evolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 782–793, May 2006.

[44] C. Vogel, S. Roth, and K. Schindler, "An evaluation of data costs for optical flow," in *Proc. German Conf. Pattern Recognit.*, 2013, pp. 343–353.

[45] C. Vogel, K. Schindler, and S. Roth, "3D scene flow estimation with a piecewise rigid scene model," *Int. J. Comput. Vis.*, vol. 115, no. 1, pp. 1–28, 2015.

[46] J. Y. A. Wang and E. H. Adelson, "Representing moving images with layers," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 625–638, Sep. 1994.

[47] Y. Wang, W. Yin, and J. Zeng. (2015). "Global convergence of ADMM in nonconvex nonsmooth optimization." [Online]. Available: https://arxiv.org/abs/1511.06324

[48] A. Weinmann and M. Storath, "Iterative Potts and Blake–Zisserman minimization for the recovery of functions with discontinuities from indirect measurements," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 471, no. 2176, p. 20140638, 2015.

[49] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "DeepFlow: Large displacement optical flow with deep matching," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, Sydney, NSW, Australia, Dec. 2013, pp. 1385–1392.

[50] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, and H. Bischof, "Anisotropic Huber-L1 optical flow," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2009, p. 3.

[51] G. Winkler and V. Liebscher, "Smoothers for discontinuous signals," *J. Nonparam. Statist.*, vol. 14, nos. 1–2, pp. 203–222, 2002.

[52] J. Xiao, H. Cheng, H. Sawhney, C. Rao, and M. Isnardi, "Bilateral filtering-based optical flow estimation with occlusion detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2006, pp. 211–224.

[53] L. Xu, J. Jia, and Y. Matsushita, "Motion detail preserving optical flow estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1744–1757, Sep. 2012.

[54] Z. Xu, S. De, M. Figueiredo, C. Studer, and T. Goldstein. (2016). "An empirical study of ADMM for nonconvex problems." [Online]. Available: https://arxiv.org/abs/1612.03349

[55] J. Yang and H. Li, "Dense, accurate optical flow estimation with piecewise parametric model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1019–1027.

[56] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime TV-L^1 optical flow," in *Proc. Joint Pattern Recognit. Symp.*, 2007, pp. 214–223.

[57] Q. Zhu, B. Du, P. Yan, H. Lu, and L. Zhang, "Shape prior constrained PSO model for bladder wall MRI segmentation," *Neurocomputing*, vol. 294, pp. 19–28, Jun. 2017.

Denis Fortun received the M.Sc. degree from Télécom Physique Strasbourg and the University of Strasbourg in 2010 and the Ph.D. degree in signal analysis and telecommunication from the University of Rennes 1 in 2014. From 2014 to 2017, he held a post-doctoral position with the Biomedical Imaging Group, École Polytechnique Fédérale de Lausanne, and with the Center for Biomedical Imaging, University of Lausanne. Since 2018, he has been a permanent CNRS Researcher with the ICube Laboratory, Strasbourg. His main research interests include inverse problems for image reconstruction, motion estimation and image analysis, applied to for biological imaging.

Martin Storath received the Diploma degree in mathematics, the Honours degree in technology management, and the Ph.D. degree in mathematics from TU München, in 2008, 2009, and 2013, respectively. He was a Researcher with Helmholtz Zentrum München, Neuherberg, Germany, and with the Biomedical Imaging Group, École Polytechnique Fédérale de Lausanne. He currently holds a post-doctoral position with the Image Analysis and Learning Group, Universität Heidelberg. His research interests include image processing, variational methods, and inverse problems.

Dennis Rickert received the master's degree in bioinformatics from Technische Universität München in 2013. He is currently a Researcher with Helmholtz Zentrum München GmbH, Neuherberg, Germany. His research interests include image processing, parameter optimization and estimation for dynamical systems, and efficient implementations of computationally demanding algorithms using HPC approaches.

Andreas Weinmann received the Diploma degree (Hons) in mathematics and computer science from Technische Universität München in 2006 and the Ph.D. degree (Hons) from Technische Universitat Graz in 2010. He is currently affiliated with Mathematics Department, Hochschule Darmstadt, and with the Institute of Computational Biology, Helmholtz Zentrum München, Neuherberg, Germany. His research interests are applied analysis as well as signal and image processing.

Michael Unser (M'89–SM'94–F'99) is a Professor and the Director of the Biomedical Imaging Group, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland. His primary area of investigation is biomedical image processing. He is internationally recognized for his research contributions to sampling theory, wavelets, the use of splines for image processing, stochastic processes, and computational bioimaging. He has published over 300 journal papers on those topics. He is the author of the book *An Introduction to Sparse Stochastic Processes* (with P. Tafti, Cambridge University Press, 2014).

From 1985 to 1997, he was with the Biomedical Engineering and Instrumentation Program, National Institutes of Health, Bethesda, MD, USA, conducting research on bioimaging.

Prof. Unser is an EURASIP fellow (2009) and a member of the Swiss Academy of Engineering Sciences. He was a recipient of several international prizes, including four IEEE-SPS Best Paper Awards and two Technical Achievement Awards from the IEEE (2008 SPS and EMBS 2010).

Dr. Unser is the Founding Chair of the Technical Committee on Bio Imaging and Signal Processing of the IEEE Signal Processing Society. He has held the position of associate Editor-in-Chief (2003–2005) for the IEEE TRANSACTIONS ON MEDICAL IMAGING. He is currently a member of the editorial boards of *SIAM Journal on Imaging Sciences*, the IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING, and *Foundations and Trends in Signal Processing*.