An Efficient Algorithm for the Piecewise Affine-Linear Mumford-Shah Model Based on a Taylor Jet Splitting

Lukas Kiefer, Martin Storath, Andreas Weinmann

***Abstract*—We propose an algorithm to efficiently compute approximate solutions of the piecewise affine Mumford-Shah model. The algorithm is based on a novel reformulation of the underlying optimization problem in terms of Taylor jets. A splitting approach leads to linewise segmented jet estimation problems for which we propose an exact and efficient solver. The proposed method has the combined advantages of prior algorithms: it directly yields a partition, it does not need an initialization procedure, and it is highly parallelizable. The experiments show that the algorithm has lower computation times and that the solutions often have lower functional values than the state-of-the-art.**

I. Introduction

Partitioning of images is a central task in computer vision. In a supervised setup, learning based approaches, most notably convolutional neural networks, have become standard. If not sufficiently many training data are available, model-based methods are employed. Classical methods are k-means clustering, region growing (see e.g. [1]), and the watershed algorithm (see e.g. [2]). Another popular technique for clustering (without having to fix the number of clusters in advance) is based on the mean shift filtering [3], [4]. See [5] for an overview of image filtering methods.

An active field in image partitioning are variational methods. The idea is to model the partitioning as the result of an optimization problem which imposes regularity on both regions and boundaries [6]. A popular variational approach to image partitioning is the *piecewise constant Mumford-Shah model.* The central idea is to partition the image domain such that the total boundary length of all segments is small and such that the induced piecewise constant function approximates the image well. The model is frequently applied for unsupervised segmentation of color images [7], [8], depth images [8], texture images [9], and medical images [10], [11], to mention only some examples. However, the piecewise constant model assumption is often restrictive: many types of images possess linear trends within their segments: consider for example the sky in a landscape image, or an object with an illumination gradient in a conventional image; further, the bias in a magnetic resonance image, as well as depth and motion fields show linear trends. In such situations applying the piecewise constant Mumford-Shah model might lead to oversegmentation since spurious segments are produced at steeper slopes; see Figures 1 and 2. A natural way of accounting for that is to employ the *piecewise affine Mumford-Shah model*. As the name suggests, it is based on affine-linear approximations on the segments. More precisely, it is given as the minimization problem

$$\operatorname*{argmin}_{u,\mathcal{P}} \sum_{P\in\mathcal{P}} \left\{ \frac{\gamma}{2}\operatorname{length}(\partial P) + \int_P |u(x)-f(x)|^2\,dx \right\}, \quad \text{subject to } u|_P \text{ is affine linear for all } P\in\mathcal{P}. \tag{1}$$

The minimum is taken w.r.t. all partitions $\mathcal{P}$ of the image domain $\Omega \subset \mathbb{R}^2$ and all corresponding piecewise affine functions u on Ω. Here, $f : \Omega \to \mathbb{R}^K$ is a vector-valued input image with K channels, and $|\cdot|^2$ denotes the squared Euclidean norm in $\mathbb{R}^K$. The parameter $\gamma > 0$ controls the tradeoff between the length penalty and the data fidelity term. Figure 1 illustrates the piecewise constant and the piecewise affine Mumford-Shah model for a natural image with color gradients.

Another possibility to account for color gradients in the image is the *piecewise smooth Mumford-Shah model* which penalizes the deviations from piecewise constant functions controlled by an additional hyperparameter β (instead of enforcing piecewise constancy). However, the piecewise smooth model comes with side-effects: the estimated discontinuity set is in general not the boundary set of a partition as it can have open ends (so-called "crack tips"), and it suffers from the gradient limit effect, i.e. the creation of spurious discontinuities at steep slopes; see [12], [13] and Figure 2. Even though linear trends are better representable by the piecewise smooth Mumford-Shah model than by the piecewise constant variant, approximating them increases its energy (in contrast to the piecewise affine variant). Therefore the solution space of the piecewise smooth model does not contain genuine piecewise affine functions but only approximations of them.

Depending on the application, the principal interest in solving (1) can be to find an optimal partition $\mathcal{P}^*$, which may be used for segmentation or superpixelations as for instance in [14], [15]; or one can be interested in finding an optimal piecewise affine function u^*, which may be used for the

L. Kiefer is with the Mathematical Imaging Group, Universität Heidelberg, and with the Department of Mathematics and Natural Sciences, Hochschule Darmstadt, Germany

M. Storath is with the Image Analysis and Learning Group, Heidelberg Collaboratory for Image Processing, Universität Heidelberg, and with the Department of Applied Natural Sciences and Humanities, University of Applied Sciences Würzburg-Schweinfurt, Germany

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

(a) Input (b) Pcw. constant model (c) Pcw. affine model

Fig. 1: The piecewise affine Mumford-Shah model provides improved partitioning results for images with color gradients compared with the piecewise constant variant.

regularization of flow fields as in [16] or for the guidance of image filters as in [17]. The focus of this paper is the efficient computation of the partition.

A. Related Work

Despite their early introduction [12], [13], [18], [19] and their relatively simple formulation, Mumford-Shah models are still a very active topic of research; see [15], [20], [21], [22], [23], [24] to mention only some examples. This is because they lead to challenging (NP-hard) optimization problems [8], [25].

We first discuss work related with the piecewise constant variant which is also widely known as the Potts model and recently also called ℓ_0 gradient smoothing. Classical algorithms for the piecewise constant Mumford-Shah model are based on simulated annealing [18] and on approximations by elliptic functionals [26]. Currently popular approaches are active contours [7], [27], [28], [29], graph cuts [8], [30], convex relaxations [31], [32], [33], [34], semi-global matching [35], [36], fused coordinate descent [37], region fusion [22], iterative thresholding type techniques [20], [38], and alternating direction method of multipliers [39]. The latter approach is currently the benchmark in terms of quality according to the comparison in [22]. Since the problem is computationally rather expensive, parallelization has recently received a lot of attention; see [31], [34], [40], [41].

Besides the mentioned challenges of computing the piecewise constant Mumford-Shah model, the piecewise affine case comes with an additional complication: a commonly used reformulation, which eliminates the dependance on the partition and which is used in many algorithmic approaches for the piecewise constant variant, has no direct analogy for the piecewise affine case. In fact, the variety of algorithms is much smaller for the piecewise affine case. One can find approaches based on graduated non-convexity [12], active contours [14], graph cuts [8], [42], [43], [44] and ADMM splitting [16]. We discuss these algorithms in more detail in the next subsection.

Further related work is the total generalized variation model (TGV) [45] which can be seen as an ℓ^1-based convex relative of the piecewise affine Mumford-Shah model. The results of TGV are approximately piecewise affine functions. In contrast to the piecewise affine Mumford-Shah model, the solution of TGV does not yield a partition of the image. In [46], the authors apply a piecewise linear image model in the context of the analysis sparsity model in the sense of compressed sensing. They consider an anisotropic error constraint model. We notice that the solutions do not yield a partition directly. We finally mention the related recent work [47] on piecewise affine estimation in connection with deep edge aware filters.

B. Algorithms for the Piecewise Affine Mumford-Shah Model

One can sort the approaches to the piecewise affine Mumford-Shah problem into two classes: those that are mainly interested in finding an approximately optimal partition $\mathcal{P}^*$ and those that are mainly interested in computing an approximately piecewise affine estimate u^*. The latter ones, in general, produce an approximation which does not yield a partition as we will see below.

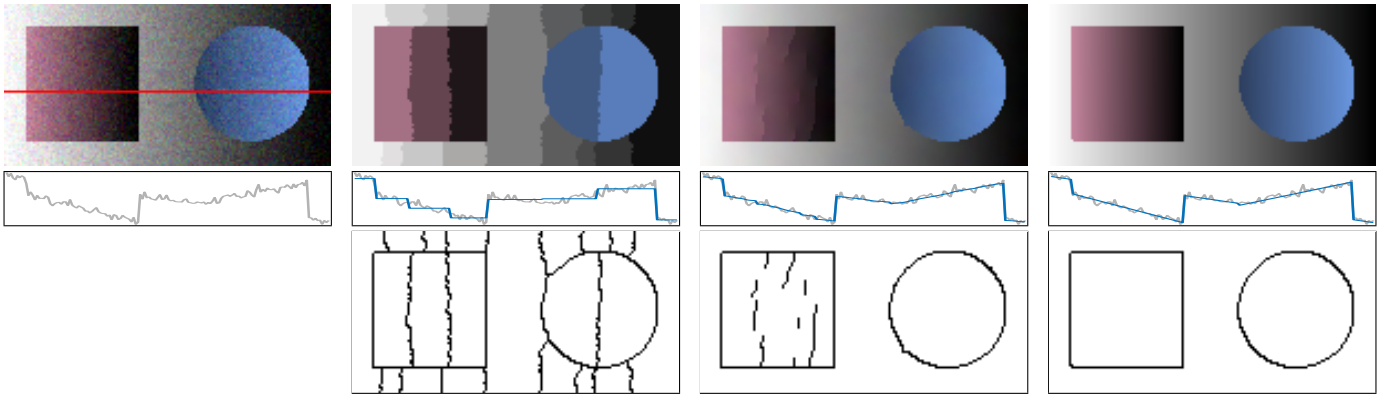
Partition-Based Methods: The common starting point is the formulation of the piecewise affine Mumford-Shah problem in terms of partitions:

$$\begin{aligned}\operatorname*{argmin}_{\mathcal{P}} \sum_{P\in\mathcal{P}} \Big\{ &\frac{\gamma}{2}\operatorname{length}(\partial P) \\ &+ \sum_{k=1}^{K} \min_{a,b,c} \int_P (ax_1+bx_2+c-f_k(x))^2\,dx \Big\}.\end{aligned} \tag{2}$$

This formulation gives rise to the following iterative repartitioning scheme

1. Choose an initial partition $\mathcal{P}^0$.
2. Iterate until convergence:
   a. for all $P\in\mathcal{P}^j$, and $k=1,\dots,K$, compute the optimal affine coefficients,
   $$(a^*,b^*,c^*) = \operatorname*{argmin}_{a,b,c} \sum_{x\in P} (ax_1+bx_2+c-f_k(x))^2;$$
   b. compute a partition $\mathcal{P}^{j+1}$ which is (approximately) optimal for the determined set of affine coefficients.

While step 2a is a simple linear regression, step 2b is an NP-hard optimization problem. One possibility to tackle this problem is to represent the partition by level set functions and to evolve them guided by a PDE (active contours approach), see e.g. [14]. Another possibility is to compute the partition by a min-cut/max-flow algorithm (graph cuts approach) which



(a) Input (b) Pcw. constant, $\gamma = 0.3$ (c) Pcw. smooth, $\gamma = 0.03$, $\beta = 10$ (d) Pcw. affine, $\gamma = 0.4$

Fig. 2: **Piecewise affine vs. piecewise smooth Mumford-Shah model for color gradients.** *Top:* solution u^*; *center:* profiles of the blue channel, *bottom:* discontinuity sets. The pcw. constant model yields oversegmentation. The pcw. smooth model shows spurious discontinuities at steeper slopes ("gradient limit effect"), and the discontinuity set is not the boundary set of a partition ("crack tips"). The pcw. affine variant recovers the steep gradients and the produced discontinuity set forms a partition.

is used for instance in [48], [49], [44]. The graph cut approach is often employed because a highly efficient library exists [8], [50], [51]. We use this method (which uses graph cuts partitioning) as the benchmark method.

Function-Based Methods without Explicit Partitioning: In their early work, Blake and Zisserman [12] proposed to use a graduated non-convexity approach (GNC). The first step of GNC is to approximate the nonconvex energy function F of the closely related weak plate model by a convex function F^*. Then a whole sequence of energy functions $F^{(p)}, 1 \geq p \geq 0$ are constructed such that $F^{(1)} = F^*$ and $F^{(0)} = F$ and $F^{(p)}$ changes in a continuous way from $F^{(1)}$ to $F^{(0)}$. One sequentially solves the corresponding minimization problems (e.g. by gradient descent) for $1 \geq p > 0$ and decreasing p using the results of a higher p as starting point for the next lower p. Blake and Zisserman point out that GNC shows good performance for small variation penalties of the weak plate model. Unfortunately, the piecewise affine Mumford-Shah is the limiting case of the weak plate model for infinitely large variation penalty, and the GNC results are less favorable in this case.

Recently, Fortun et al. [16] proposed a method using a splitting into coupled subproblems. The splitting is such that the subproblems boil down to univariate piecewise affine estimation problems which can be solved efficiently. We will elaborate on the relations to the proposed method later on (Section II-F). For now, the important point is that the final results do not provide partitions.

Advantages and Tradeoffs: The central advantage of the partition-based approaches is that they explicitly yield a partition which can be directly employed in a segmentation pipeline or as a basis for a superpixel generator. Also, if desired, a corresponding piecewise affine function can be easily computed by linear regression on the segments. On the flipside, the existing partition-based approaches need an initialization procedure that yields an initial partitioning. Further, their runtime grows with the number of labels. Additionally, parallelization of the algorithms is more involved. The advantages of function-based methods are that they do not need an initialization procedure in the sense that they do not require an initialization by an initial partitioning, and that they can be easily parallelized. The main drawback of function-based methods is that they only provide a not necessarily piecewise affine approximation of the piecewise affine function u whose discontinuity set is in general not the boundary set of a partition. This implies, as an undesirable side-effect, that the functional value in (2) cannot be evaluated directly which makes it difficult to quantify the quality of the algorithms.

C. Contribution

In this paper, we propose a novel algorithm to efficiently compute approximate solutions of the piecewise affine Mumford-Shah model (1). The key novelties of our method are as follows:

- we propose a formulation of the piecewise affine Mumford-Shah model as a minimization problem with respect to Taylor jets, i.e., fields of local Taylor polynomials. This eliminates the dependance on the partition $\mathcal{P}$ in (1);
- we propose a novel splitting approach based on a jet coupling which leads to linewise segmented jet estimation problems;
- we propose an efficient and exact solver for the linewise segmented jet estimation problems.

The proposed algorithm unifies the advantages of partition-based methods and those of function-based methods:

- it directly yields a partition $\mathcal{P}$;
- it does not need an initialization procedure;
- its average computation times are almost independent of the choice of the edge penalty;
- it is highly parallelizable.

The experimental results show that the proposed method improves upon the state-of-the-art approach to the piecewise affine Mumford-Shah model in the sense that the algorithm has lower computation times and often achieves lower functional values.

II. Proposed Method

To keep the presentation focused on the novel concepts, we carry out the derivations for single-channel images ($K = 1$) from Section II-A to Section II-D. The extension for multi-channel images ($K > 1$) is discussed in Section II-E.

A. A Jet Formulation of the Piecewise Affine Mumford-Shah Model

A key point in our derivation is the formulation of problem (1) in terms of Taylor jets.

The *Taylor jet* of a function u is a field of Taylor polynomials, i.e., with every point x in the domain Ω of u we associate the truncated Taylor expansion of u in the point x up to a certain order k, where k is called the order of the jet. In the following, we will need the (first order) jet $\mathcal{J}u$ of a function u at the point $x \in \Omega$ which we define as the first order polynomial $\mathcal{J}u(x)$ given by

$$\mathcal{J}u(x)(z) := T_x u(z) = u(x) + \tfrac{\partial u(x)}{\partial x_1}(z_1 - x_1) + \tfrac{\partial u(x)}{\partial x_2}(z_2 - x_2) \tag{3}$$

where $T_x u$ is the (first order) Taylor polynomial of u centered at x and where $z \in \mathbb{R}^2$ is the argument of the polynomial. Summing up, the jet $\mathcal{J}u : \Omega \to \Pi_1$ is a function on Ω with values in the space Π_1 of bivariate polynomials of order one; in particular, $\mathcal{J}u(x)$ is a polynomial, and $\mathcal{J}u(x)(z)$ denotes the point evaluation of the polynomial $\mathcal{J}u(x)$ at the point $z \in \mathbb{R}^2$. We note that the variables z in the above definition (3) are shifted versions of the ones typically used for the jet formulation. The reason for our choice here is that we will later need the notion of equality of the polynomials in the way defined above; this will become clear in the following. It is a basic but important observation that u is piecewise affine linear if and only if the jet $\mathcal{J}u$ is a piecewise constant field.

This allows us to rewrite (1) in the form

$$\begin{aligned} \operatorname*{argmin}_{u,\mathcal{P}} \; & \sum_{P \in \mathcal{P}} \left\{ \frac{\gamma}{2} \operatorname{length}(\partial P) + \int_P (u(x) - f(x))^2 \, dx \right\}, \\ & \text{subject to } \mathcal{J}u|_P \text{ is constant for all } P \in \mathcal{P}. \end{aligned} \tag{4}$$

If we denote the length of the jump set of the jet $\mathcal{J}u$ by $\|\nabla \mathcal{J}u\|_0$, we can cast (4) into the form

$$u^* = \operatorname*{argmin}_u \; \gamma \, \|\nabla \mathcal{J}u\|_0 + \int_\Omega (u(x) - f(x))^2 \, dx. \tag{5}$$

(We note that the factor $1/2$ in the first term of (4), which compensated for the double counting of the boundaries, vanishes in this formulation.) Instead of minimizing over u directly, we lift the problem to the jet space. This has the advantage that we may access the regularizing term more explicitly, while the data term still has a concrete representation in terms of jets. An important point is that a jet recovers its underlying function at x by evaluating the polynomial $\mathcal{J}u(x)$ at (the point) x, i.e. $\mathcal{J}u(x)(x) = u(x)$. Using this fact, we can formulate (5), and thus the initial problem (1), as a minimization problem over polynomial fields. We obtain the following jet formulation:

$$J^* = \operatorname*{argmin}_{J \in \mathrm{PC}(\Omega;\Pi_1)} \; \gamma \, \|\nabla J\|_0 + \int_\Omega (J(x)(x) - f(x))^2 \, dx. \tag{6}$$

Here, $\mathrm{PC}(\Omega;\Pi_1)$ denotes the space of piecewise constant fields of first order polynomials. Note that, as J is a piecewise constant polynomial field, there is a corresponding piecewise affine function u with $\mathcal{J}u = J$ on the complement of the discontinuity set; in other words, J is the jet of a piecewise affine function. This reveals the equivalence of the problems (6) and (1).

The key advantage of the jet formulation (6) is that the piecewise affine problem in u has transformed into a piecewise constant problem in J. In particular, the connected components of the level sets of J^* are exactly the segments of the corresponding partition $\mathcal{P}^*$, and we have $u^*(x) = J^*(x)(x)$ for all $x \in \Omega$.

B. Discretization of the Proposed Jet Formulation

We may adopt a common way of discretizing the length penalty term in (6),

$$\|\nabla J\|_0 \approx \sum\nolimits_{s=1}^{S} \omega_s \|\nabla_{d^s} J\|_0, \tag{7}$$

see [52], [53], [39]. The terms on the right hand side count the number of changes of the field J w.r.t. to a direction $d^s \in \mathbb{Z}^2$; that is,

$$\|\nabla_{d^s} J\|_0 = |\{x \in \Omega' : J(x) \neq J(x+d^s), \; x+d^s \in \Omega'\}|, \tag{8}$$

where $\Omega' = \{1, \dots, M\} \times \{1, \dots, N\}$ denotes the discretized domain. The directions d^s form a neighborhood system $\{d^1, \dots d^S\}$, with $S \geq 2$. The simplest choice is the system $\{e_1, e_2\}$, $e_1 = (1,0)^T, e_2 = (0,1)^T$, with the weights $\omega_{1,2} = 1$. To reduce anisotropy effects, we here focus on the more isotropic discretization $\{e_1, e_2, e_1+e_2, e_1-e_2\}$. We here use the weights $\omega_{1,2} = \sqrt{2} - 1$ and $\omega_{3,4} = 1 - \frac{\sqrt{2}}{2}$ used in [11] which coincide up to normalization with those of [54], [53].

C. Proposed Jet-Based Splitting Approach

Using the discretization (7) for the proposed jet formulation (6) we obtain the discretized problem

$$J^* = \operatorname*{argmin}_{J:\Omega' \to \Pi_1} \gamma \sum_{s=1}^{S} \omega_s \|\nabla_{d^s} J\|_0 + \sum_{x \in \Omega'} (J(x)(x) - f(x))^2. \tag{9}$$

Our goal is to compute approximate solutions of (9). Towards an optimization algorithm, we first split up the target J into S polynomial fields, $J^1, \dots, J^S$, subject to the constraints that they are all equal:

$$\begin{aligned} \operatorname*{argmin}_{J^1,\dots,J^S} \; & \sum_{s=1}^{S} \left\{ \gamma \omega_s \|\nabla_{d^s} J^s\|_0 + \frac{1}{S} \sum_{x \in \Omega'} (J^s(x)(x) - f(x))^2 \right\} \\ & \text{subject to } J^s = J^t \text{ for all } 1 \leq s < t \leq S. \end{aligned} \tag{10}$$

Since two bivariate first-order polynomials are equal if and only if their evaluation in three non-collinear points in $\mathbb{R}^2$ is equal, we can rewrite (10) as

$$\begin{aligned}\operatorname*{argmin}_{J^1,\dots,J^S} \sum_{s=1}^{S}\Big\{\gamma\omega_s\|\nabla_{d^s}J^s\|_0+\frac{1}{S}\sum_{x\in\Omega'}(J^s(x)(x)-f(x))^2\Big\}\\ \text{s.t. } J^s(x)(x)=J^t(x)(x),\\ J^s(x)(x+e_1)-J^s(x)(x)=J^t(x)(x+e_1)-J^t(x)(x),\\ J^s(x)(x+e_2)-J^s(x)(x)=J^t(x)(x+e_2)-J^t(x)(x),\\ \text{for all } x\in\Omega', \text{and all } s,t \text{ with } 1\le s<t\le S.\end{aligned} \tag{11}$$

Recall $e_1=(1,0)^T, e_2=(0,1)^T$ denote the coordinate directions. It is convenient to introduce the following abbreviations

$$\begin{aligned}u(x)&:=u_J(x):=J(x)(x),\\ a(x)&:=a_J(x):=J(x)(x+\mathrm{e}_1)-J(x)(x),\\ b(x)&:=b_J(x):=J(x)(x+e_2)-J(x)(x),\end{aligned}$$

for the function value at the base point x and the slopes of the polynomial $J(x)$. Then, the constraint in (11) reads

$$u^s(x)=u^t(x),\ a^s(x)=a^t(x),\ b^s(x)=b^t(x), \tag{12}$$

for all $x\in\Omega'$, and all s,t with $1\le s<t\le S$. We note that the presented jet formulation results in rather strong coupling incorporating also the slopes/derivatives of the considered polynomials.

We now decompose the constrained problem into coupled subproblems using an ADMM approach. That is, we form the augmented Lagrangian form of (11) and iteratively perform a block-coordinate-wise minimization and gradient ascent steps on the Lagrange multipliers. (See [55] for an overview on ADMM.) We note that ADMM schemes often work well for nonconvex problems; see, e.g., [56], [57], [58], [24]. The augmented Lagrangian form of (11) is given by

$$\begin{aligned}&\mathcal{L}_{\mu,\nu}(\{J^s\},\{\lambda^{s,t}\},\{\tau^{s,t}\},\{\rho^{s,t}\})\\ &=\sum_{s=1}^{S}\Big\{\omega_s\gamma\|\nabla_{d_s}J^s\|_0+\tfrac{1}{S}\|u^s-f\|^2\\ &\quad+\sum_{t=s+1}^{S}\Big(\tfrac{\mu}{2}\big\|u^s-(u^t-\tfrac{\lambda^{s,t}}{\mu})\big\|^2-\tfrac{1}{2\mu}\|\lambda^{s,t}\|^2\\ &\qquad+\tfrac{\nu}{2}\big\|a^s-(a^t-\tfrac{\tau^{s,t}}{\nu})\big\|^2-\tfrac{1}{2\nu}\|\tau^{s,t}\|^2\\ &\qquad+\tfrac{\nu}{2}\big\|b^s-(b^t-\tfrac{\rho^{s,t}}{\nu})\big\|^2-\tfrac{1}{2\nu}\|\rho^{s,t}\|^2\Big)\Big\},\end{aligned} \tag{13}$$

where the $\lambda^{s,t},\tau^{s,t},\rho^{s,t}\in\mathbb{R}^{M\times N}$ denote the Lagrange multipliers and $\|\cdot\|$ is the Frobenius norm, i.e. $\|u\|^2=(\sum_{i,j}u_{ij}^2)$ for $u\in\mathbb{R}^{M\times N}$. The parameters $\mu,\nu>0$ determine how strong differences between the split variables are penalized. Note that we utilize two different coupling parameters: one for the slope variables a,b and one for the base point variable u. This is because slopes and base points typically live on different scales.

In each iteration we minimize the Lagrangian $\mathcal{L}$ w.r.t. the polynomial fields $J^1,\dots,J^S$. For a fixed s the according minimization problem reads

$$\begin{aligned}\operatorname*{argmin}_J\mathcal{L}_{\mu,\nu}=\operatorname*{argmin}_J\Big\{&\omega_s\gamma\|\nabla_{d_s}J\|_0+\tfrac{1}{S}\|u-f\|^2\\ &+\sum_{t=s+1}^{S}\Big(\tfrac{\mu}{2}\|u-(u^t-\tfrac{\lambda^{s,t}}{\mu})\|^2+\tfrac{\nu}{2}\|a-(a^t-\tfrac{\tau^{s,t}}{\nu})\|^2\\ &\qquad+\tfrac{\nu}{2}\|b-(b^t-\tfrac{\rho^{s,t}}{\nu})\|^2\Big)\\ &+\sum_{r=1}^{s-1}\Big(\tfrac{\mu}{2}\|u-(u^r+\tfrac{\lambda^{r,s}}{\mu})\|^2+\tfrac{\nu}{2}\|a-(a^r+\tfrac{\tau^{r,s}}{\nu})\|^2\\ &\qquad+\tfrac{\nu}{2}\|b-(b^r+\tfrac{\rho^{r,s}}{\nu})\|^2\Big)\Big\}.\end{aligned} \tag{14}$$

Note that all other terms in (13) do not depend on J^s, so we dropped them. In the following we bring (14) into a convenient form. To this end, we will use repeatedly the fact that

$$\sum\nolimits_{i=1}^{N}x_i(p-t_i)^2=\big(\sum\nolimits_{i=1}^{N}x_i\big)\big(p-\tfrac{\sum_{i=1}^{N}t_ix_i}{\sum_{i=1}^{N}x_i}\big)^2+C \tag{15}$$

holds for $p,t_1,\dots,t_N\in\mathbb{R}$ and $x_1,\dots,x_N>0$ and a constant C that does not depend on p. Initially, this allows us to rewrite the summands in (14) and we get

$$\begin{aligned}\operatorname*{argmin}_J\Big\{&\omega_s\gamma\|\nabla_{d_s}J\|_0+\tfrac{1}{S}\|u-f\|^2\\ &+\tfrac{(S-s)\mu}{2}\big\|u-\frac{\sum_{t=s+1}^{S}(u^t-\frac{\lambda^{s,t}}{\mu})}{(S-s)}\big\|^2\\ &+\tfrac{(s-1)\mu}{2}\big\|u-\frac{\sum_{r=1}^{s-1}(u^r+\frac{\lambda^{r,s}}{\mu})}{(s-1)}\big\|^2\\ &+\tfrac{(S-s)\nu}{2}\big\|a-\frac{\sum_{t=s+1}^{S}(a^t-\frac{\tau^{s,t}}{\nu})}{(S-s)}\big\|^2\\ &+\tfrac{(s-1)\nu}{2}\big\|a-\frac{\sum_{r=1}^{s-1}(a^r+\frac{\tau^{r,s}}{\nu})}{(s-1)}\big\|^2\\ &+\tfrac{(S-s)\nu}{2}\big\|b-\frac{\sum_{t=s+1}^{S}(b^t-\frac{\rho^{s,t}}{\nu})}{(S-s)}\big\|^2\\ &+\tfrac{(s-1)\nu}{2}\big\|b-\frac{\sum_{r=1}^{s-1}(b^r+\frac{\rho^{r,s}}{\nu})}{(s-1)}\big\|^2\Big\}.\end{aligned} \tag{16}$$

Again we dropped terms that do not depend on J. For readability we introduce the following abbreviations for the sums in (16):

$$\begin{aligned}\Lambda&=\sum\nolimits_{t=s+1}^{S}(u^t-\lambda^{s,t}/\mu),\quad \Psi=\sum\nolimits_{t=s+1}^{S}(a^t-\tau^{s,t}/\nu),\\ \Delta&=\sum\nolimits_{t=s+1}^{S}(b^t-\rho^{s,t}/\nu),\quad \Gamma=\sum\nolimits_{r=1}^{s-1}(u^r+\lambda^{r,s}/\mu),\\ \Phi&=\sum\nolimits_{r=1}^{s-1}(a^r+\tau^{r,s}/\nu),\qquad \Theta=\sum\nolimits_{r=1}^{s-1}(b^r+\rho^{r,s}/\nu).\end{aligned}$$

After applying (15) to all but the first line of (16) we obtain

$$\begin{aligned}\operatorname*{argmin}_J\ &\omega_s\gamma\|\nabla_{d_s}J\|_0+\tfrac{1}{S}\|u-f\|^2+\tfrac{(S-1)\mu}{2}\big\|u-\tfrac{\Lambda+\Gamma}{S-1}\big\|^2\\ &+\tfrac{(S-1)\nu}{2}\big\|a-\tfrac{\Psi+\Phi}{S-1}\big\|^2+\tfrac{(S-1)\nu}{2}\big\|b-\tfrac{\Delta+\Theta}{S-1}\big\|^2.\end{aligned} \tag{17}$$

A final application of (15) to both remaining terms depending on u in (17) leads to

$$\begin{aligned}\operatorname*{argmin}_J\, &\omega_s\gamma\|\nabla_{d_s}J\|_0 + \tfrac{2+\mu S(S-1)}{2S}\big\|u - \tfrac{2f+\mu S\left(\Lambda+\Gamma\right)}{2+\mu S(S-1)}\big\|^2 \\ &+ \tfrac{(S-1)\nu}{2}\big\|a - \tfrac{\Psi+\Phi}{S-1}\big\|^2 + \tfrac{(S-1)\nu}{2}\big\|b - \tfrac{\Delta+\Theta}{S-1}\big\|^2.\end{aligned} \tag{18}$$

Multiplying (18) by $\frac{2}{(S-1)\nu}$, we get

$$\begin{aligned}\operatorname*{argmin}_J\, &\tfrac{2\omega_s\gamma}{(S-1)\nu}\|\nabla_{d_s}J\|_0 + \tfrac{2+\mu S}{\nu S}\big\|u - \tfrac{2f+\mu S\left(\Lambda+\Gamma\right)}{2+\mu S(S-1)}\big\|^2 \\ &+ \big\|a - \tfrac{\Psi+\Phi}{S-1}\big\|^2 + \big\|b - \tfrac{\Delta+\Theta}{S-1}\big\|^2.\end{aligned} \tag{19}$$

Together with the gradient ascents on the Lagrange multipliers we derive the iterative procedure

$$\begin{aligned}(J^s)^{j+1} &= \\ &\operatorname*{argmin}_J \tfrac{2\omega_s\gamma}{(S-1)\nu_j}\|\nabla_{d_s}J\|_0 + \tfrac{2+\mu_j S}{\nu_j S}\|u-(w^s)^j\|^2 \\ &+ \|a-(y^s)^j\|^2 + \|b-(z^s)^j\|^2 \quad \forall s=1,...,S, \\ (\lambda^{s,t})^{j+1} &= (\lambda^{s,t})^j + \mu_j((u^s)^{j+1} - (u^t)^{j+1}) \quad \forall s<t, \\ (\tau^{s,t})^{j+1} &= (\tau^{s,t})^j + \nu_j((a^s)^{j+1} - (a^t)^{j+1}) \quad \forall s<t, \\ (\rho^{s,t})^{j+1} &= (\rho^{s,t})^j + \nu_j((b^s)^{j+1} - (b^t)^{j+1}) \quad \forall s<t,\end{aligned} \tag{20}$$

where the superscript j denotes the iteration number and where we used the abbrevations

$$(w^s)^j = \frac{2f+\mu_j S\left(\sum_{t=s+1}^{S}\left((u^t)^j - \frac{(\lambda^{s,t})^j}{\mu_j}\right) + \sum_{r=1}^{s-1}\left((u^r)^{j+1} + \frac{(\lambda^{r,s})^{j+1}}{\mu_j}\right)\right)}{2+\mu_j S(S-1)},$$

$$\begin{aligned}(y^s)^j &= \tfrac{1}{S-1}\sum\nolimits_{t=s+1}^{S}\left((a^t)^j - \tfrac{(\tau^{s,t})^j}{\nu_j}\right) \\ &+ \tfrac{1}{S-1}\sum\nolimits_{r=1}^{s-1}\left((a^r)^{j+1} + \tfrac{(\tau^{r,s})^{j+1}}{\nu_j}\right), \\ (z^s)^j &= \tfrac{1}{S-1}\sum\nolimits_{t=s+1}^{S}\left((b^t)^j - \tfrac{(\rho^{s,t})^j}{\nu_j}\right) \\ &+ \tfrac{1}{S-1}\sum\nolimits_{r=1}^{s-1}\left((b^r)^{j+1} + \tfrac{(\rho^{r,s})^{j+1}}{\nu_j}\right).\end{aligned}$$

As it is common when dealing with non-convex problems, we employ increasing coupling sequences $(\mu_j)_{j\in\mathbb{N}}, (\nu_j)_{j\in\mathbb{N}}$ as coupling parameters. Note that the computation of w^s, y^s, z^s and the Lagrange multiplier updates only involve pointwise basic arithmetic operations. We provide a pseudocode for the ADMM strategy in Algorithm 1 in the appendix.

The expensive part of (20) consists of solving the S non-convex problems w.r.t. the targets J (first equation of (20)) which are – omitting the indices – of the prototypical form

$$\operatorname{argmin}_J\ \gamma'\,\|\nabla_d J\|_0 + \eta^2\|u-w\|^2 + \|a-y\|^2 + \|b-z\|^2, \tag{21}$$

for some direction d of the neighborhood system and some $\gamma', \eta > 0$. First, the problem (21) can be decomposed into pathwise subproblems as follows: let L_x be the straight line passing through x in direction d and let $\mathcal{Q}$ be the set of all such lines, $\mathcal{Q} = \{L_x : x \in \Omega'\}$. (Note that $\mathcal{Q}$ – as a set – contains each line exactly once.) As $\mathcal{Q}$ forms a partition of the image domain, we can reformulate (21) as

$$\begin{aligned}(J')^* = &\\ \operatorname*{argmin}_{J:\Omega'\to\Pi_1} &\sum_{Q\in\mathcal{Q}}\Big\{\gamma'\,\big|\{x\in Q : J(x)\neq J(x+d),\ x+d\in Q\}\big| \\ &+ \sum_{x\in Q}\Big(\eta^2(u(x)-w(x))^2 + (a(x)-y(x))^2 \\ &+ (b(x)-z(x))^2\Big)\Big\}.\end{aligned} \tag{22}$$

This reveals that the functional consists of summands that operate exclusively within their respective lines. Therefore it is sufficient to minimize the functional for each line separately; that is, computing for all $Q \in \mathcal{Q}$

$$\begin{aligned}(J'_Q)^* = &\\ \operatorname*{argmin}_{J':Q\to\Pi_1} &\gamma'\,\big|\{x\in Q : J'(x)\neq J'(x+d),\ x+d\in Q\}\big| \\ &+ \sum_{x\in Q}\Big(\eta^2(u(x)-w(x))^2 + (a(x)-y(x))^2 \\ &+ (b(x)-z(x))^2\Big),\end{aligned} \tag{23}$$

and setting $(J')^*(q) = (J'_Q)^*(q)$ for all $q \in Q$. Note that we can solve (23) for all lines $Q \in \mathcal{Q}$ in parallel. So, a number of $\min\{M, N\}$ problems can be solved in parallel. Here, we exploit this by using multicore CPU processing. Parallelization on the GPU gets reasonable for larger image sizes.

D. Efficient Solution of Linewise Segmented Jet Problems

The computational effort of the proposed algorithm is dominated by solving the linewise jet problems (23). To obtain an efficient overall algorithm, we here propose a fast solver.

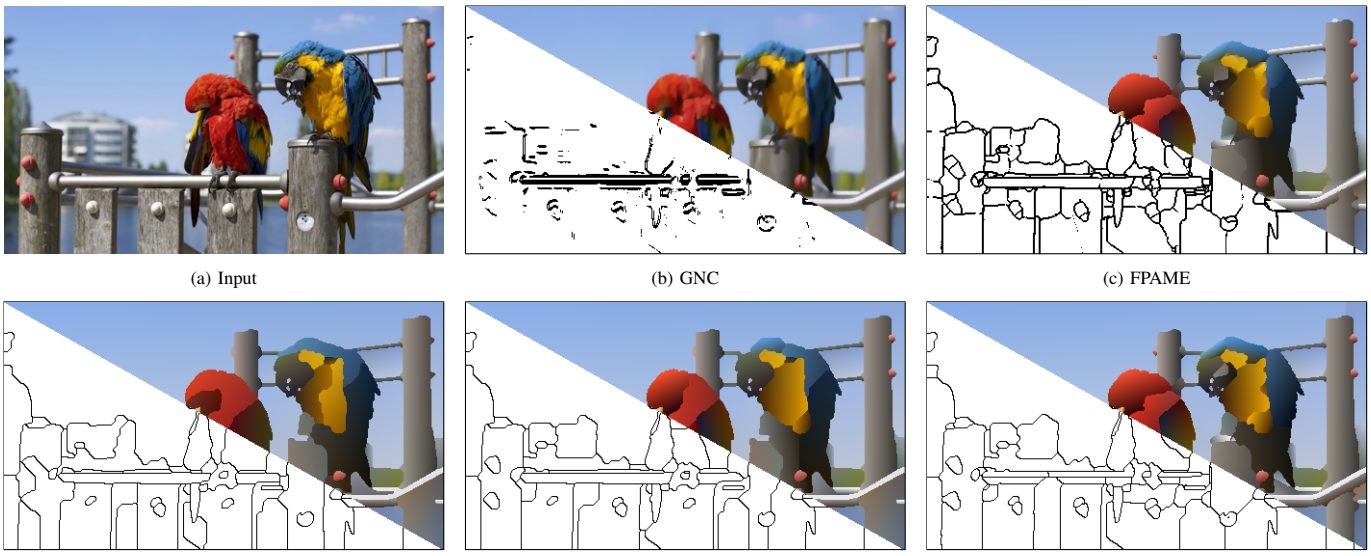
Let Q be an arbitrary line in direction d, parametrized as $Q = (q_1, q_2 \ldots, q_L)$ with $q_l = q_1 + (l-1)d$. Our first crucial observation is that (23) essentially is a one-dimensional segmented least squares problem on the line Q. Solving it consists of two steps. At first, we compute an optimal partition $\mathcal{I}^*$ on the line Q, i.e., a minimizer of

$$\mathcal{I}^* = \operatorname*{argmin}_{\mathcal{I}\text{ partition of }Q}\ \gamma'\,|\mathcal{I}| + \sum\nolimits_{I\in\mathcal{I}}\mathcal{E}^I. \tag{24}$$

Here $\mathcal{E}^I$ denotes the residual sum of squares of the best approximating bivariate linear polynomial p with $p(x) = \alpha x_1 + \beta x_2 + \delta$ on the "interval" $I = \{q_l, q_{l+1}, \ldots, q_r\}$ of the data term in (23):

$$\mathcal{E}^I = \min_{p\in\Pi_1}\sum_{x\in I}\eta^2(p(x)-w(x))^2 + (\alpha - y(x))^2 + (\beta - z(x))^2. \tag{25}$$

After having determined an optimal partition $\mathcal{I}^*$ and the minimizing argument in (25), denoted by p_I^* for all $I \in \mathcal{I}^*$, the solution $(J'_Q)^*$ is given on each segment $I \in \mathcal{I}^*$ by $(J'_Q)^*(q) = p_I^*$ for all $q \in I$. This constitutes the second step and yields a solution of (23) provided we have a solution for (24).



(a) Input (b) GNC (c) FPAME

(d) GC (64 labels) (e) GC (256 labels) (f) Proposed

Fig. 3: **Qualitative comparison of algorithms.** *Upper half:* results u^*; *lower half:* edge sets derived from u^* for the partition-free methods (GNC and FPAME), and boundaries of partition $\mathcal{P}^*$ for the partition-based methods (GC and the proposed method), respectively. The edge sets of the partition-free methods do not always consist of closed curves, and are not the boundaries of a partition.

Univariate partitioning problems of the form (24) can be solved by a standard dynamic programming scheme; see e.g. [59], [60]. Its basic idea is that the functional value in (24), denoted by B^*, satisfies the Bellman equation

$$B_r^* = \min_{l=1,\dots,r} \left\{ \mathcal{E}^{\{q_l,\dots,q_r\}} + \gamma + B_{l-1}^* \right\}, \tag{26}$$

where we let $B_0^* = -\gamma$. By the dynamic programming principle, we successively compute B_1^*, B_2^*, until we reach B_L^*. As our primary interest is the optimal partition $\mathcal{I}^*$, rather than the minimal functional value B_L^*, we keep track of a corresponding partition. An economic way to do so is to store at step r the minimizing argument l^* of (26) as the value C_r so that C encodes the boundaries of an optimal partition. We refer to [60] for a detailed description.

To implement the dynamic program for problem (24) according to (26) we have to solve $\mathcal{O}(L^2)$ problems of the form (25). More precisely, (26) reveals that the polynomial approximation errors $\mathcal{E}^I$ have to be computed for all $\mathcal{O}(L^2)$ "intervals" of the line Q. As (25) leads to a linear regression problem we could use a standard solver. But since we have to compute $\mathcal{O}(L^2)$ such problems, there is need for a tailored computation scheme to obtain an efficient overall algorithm. We here utilize such a tailored scheme exploiting additional structure of the problem inspired by [61]. To this end, we formulate (25) as the least square problem

$$\mathcal{E}^{\{q_l,\dots,q_r\}} = \min_{\alpha,\beta,\delta} \|A^h(\alpha,\beta,\delta)^T - g^{lr}\|^2, \tag{27}$$

where $h = r-l+1$ and the $(3h\times 3)$-matrix A^h and the vector g^{lr} are given by

$$A^h = \begin{pmatrix} \eta d_1 & \eta d_2 & \eta \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ & \vdots & \\ hd_1\eta & hd_2\eta & \eta \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad \text{and} \quad g^{lr} = \begin{pmatrix} \eta w_l \\ y_l \\ z_l \\ \vdots \\ \eta w_r \\ y_r \\ z_r \end{pmatrix}. \tag{28}$$

We use the QR algorithm to solve (27). Two points are crucial for fast solver: first, we exploit that the QR algorithm directly computes $\mathcal{E}^{\{q_l,\dots,q_r\}}$ which appears as the squared residual norm after the elimination. This way we do not have to compute a minimizer $(\alpha^*,\beta^*,\delta^*)$ explicitly. Secondly, we get $\mathcal{E}^{\{q_{l-1},\dots,q_r\}}$ from $\mathcal{E}^{\{q_l,\dots,q_r\}}$ by updating the QR decomposition of A^h to one of A^{h+1} in $\mathcal{O}(1)$ by employing Givens rotations. Thus, we obtain an $\mathcal{O}(1)$ scheme for computing the $\mathcal{E}^I$. In total the worst case complexity of solving (23) for a line Q is $\mathcal{O}(L^2)$.

E. Extension for Multichannel Images

We first note that solving the piecewise affine Mumford-Shah problem (1) for multichannel images $f : \Omega \to \mathbb{R}^K$ is not equivalent to solving the single-channel variant channel by channel. The latter approach typically leads to undesired artifacts as the partition boundaries are not enforced to be aligned over the channels.

To extend the proposed approach to the multichannel piecewise affine Mumford-Shah problem, we consider the first order Taylor jet of $u : \Omega \to \mathbb{R}^K$ which is given by

$$\mathcal{J}u(x) = T_x u = (T_x u_1, ..., T_x u_K)^T, \tag{29}$$

Fig. 4: **Qualitative comparison to graph cuts.** The proposed approach and the graph cut approach both provide reasonable partitions yet with some slight differences. For example at $\gamma = 0.5$, graph cuts still opens a segment for the water, but not for the fence. In the proposed approach, this is vice versa. As a general tendency, we observe that the proposed approach results in more segments throughout all γ-levels.

and which consists of the first order Taylor expansions of the component functions $u_i : \Omega \to \mathbb{R}$ of u. Then, the multichannel version of the jet formulation (6) is given by

$$J^* = \operatorname*{argmin}_{J \in \mathrm{PC}(\Omega;(\Pi_1)^K)} \gamma \, \|\nabla J\|_0 + \sum_{k=1}^{K} \int_\Omega (J_k(x)(x) - f_k(x))^2 \, dx. \tag{30}$$

where the minimum is taken over piecewise constant multichannel jets. As the domain of the image does not change in the multichannel case, the discretization of the jump penalty in (7) remains unchanged. Note that the alignment of the partition boundaries is enforced which can be best seen when looking at the counting of the directional differences in (8): in the multichannel case, we have that $J(x) \neq J(x + d^s)$ if and only if they are different in *at least one component,* i.e., if and only if $J_k(x) \neq J_k(x + d^s)$ for at least one $k \in \{1, \dots, K\}$. That means, opening a "jump" between x and $x + d^s$ in all channels has no extra costs compared to opening a jump in a single channel. The further derivation of the algorithm is analogous to the single channel case and can be found in the appendix.

F. Relation to other Approaches

In contrast to this work, the method in [16] is based on a coupling of the function values u only. In view of the jet formulation of the piecewise affine model a "complete" coupling requires also to couple the slopes. Hence, the method in [16] is the straightforward extension of the method in [39] and can be seen as a relaxation of the proposed method. The approach in [16] leads to standard linewise piecewise affine subproblems instead of the jet estimation problems needed in this work. In contrast, the proposed method requires a tailored subproblem solver, see Section II-D. Compared with [16] a main advantage of the proposed method is that it yields a partition directly.

Another related problem is given (in a discretized version) by

$$\operatorname*{argmin}_{u:\Omega' \to \mathbb{R}} \gamma \sum_{s=1}^{S} \omega_s \|\nabla^2_{d^s} u\|_0 + \sum_{x \in \Omega'} (u(x) - f(x))^2. \tag{31}$$

We stress that this is not equivalent to the piecewise affine Mumford-Shah model. The relevant difference is that $\|\nabla^2_{d^s} u\|_0$ counts the number of kinks of u instead of the number of affine coefficient changes. The practical consequence is that the functional favors solutions which tend to have kinks rather than jumps.

III. Experimental Results

We have implemented the proposed method in C++. As initial parameters we set $\mu_0 = 0.1\gamma, \nu_0 = 150\gamma\mu_0$. In each iteration, we repeated scheme (20) 5 times with the same value for μ, ν. Then we continued with the intermediate result

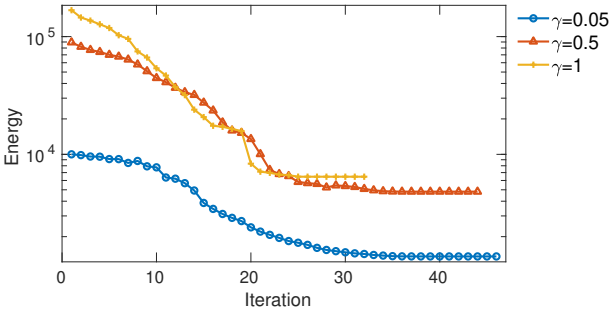


Fig. 5: **Quantitative evaluation: iterations vs. energies.** The plot shows the energy value after each iteration for the results of the proposed method in Figure 4. We observe an overall energy decrease in the iterations.

TABLE I: **Quantitative evaluation: energies.** Mean energies over the Berkeley test set for the methods which allow to directly evaluate the functional value.

	$\gamma = 0.05$	0.15	0.50	1.00	2.00
GC (64 labels)	1802.2	3180.3	5484.8	7324.7	9551.5
GC (256 labels)	1802.1	**3178.2**	5480.7	7317.5	9526.4
Proposed	**1801.6**	3196.2	**5480.1**	**7267.7**	**9468.6**

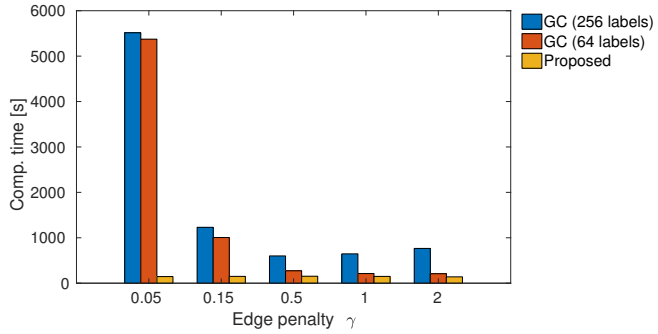


Fig. 6: **Quantitative evaluation: runtimes.** Average runtimes for the Berkeley test set. The average computation times of the proposed approach are almost constant w.r.t. the edge penalty.

with lowest energy and increased μ, ν by the factor φ to improve the results w.r.t. the energy value. (Note that we also always observe convergence when we omit such repetitions.) We observed both visually and quantitatively appealing results for the initial choice of $\varphi = 1.7$ which is set to $\varphi = 1.3$ after a relative decrease of energy greater than $2 \cdot 10^{-2}$ was observed. The iteration was stopped when the maximum norm of the difference of the split variables of the compass directions e_1, e_2 was smaller than 10^{-4}.

We compare the proposed approach to graduated non-convexity (GNC), the method of [16] (FPAME), and the iterative repartitioning scheme based on graph cuts (GC). We have implemented GNC following the description in [12]. Recall that GNC solves a relaxed problem depending on a variation parameter. We tested several such parameters, and obtained the best visual results for the value 5^4. For FPAME, we adapted the method [16] to the ℓ^2 data term. Since GNC and FPAME do not directly yield a partition we extracted their edge sets by thresholding on the second order finite differences as in [12], [16]. For the iterative repartitioning scheme, we used the α-expansion algorithm of the toolbox GCO v3.0 [8], [50], [51]. To get a reasonable initial partition $\mathcal{P}^0$ we extracted a subset of L color values from the image and performed a piecewise constant partitioning with these initial labels. Then we proceeded with the iterative repartitioning approach until the relative improvement of the energy was less than 10^{-3} or after a maximum number of 10 iterations was reached. All experiments were conducted on a workstation (Intel XeonE5-2620v4, 2.10GHz, 16 cores, 256 GB RAM).

A qualitative comparison is given in Figure 3. We observe that GNC does not give a genuine piecewise affine estimate. The edge sets of GNC and FPAME have open ends. Hence, they are not the boundaries of a partition, and so GNC and FPAME do not allow for an evaluation of the functional value which hinders a quantitative evaluation in terms of the energy. (Note that [12], [16] also do not provide a postprocessing which we could use to obtain a partition.) Thus, in the following quantitative evaluation, we focus on a detailed comparison to GC.

For the quantitative evaluation, we compare our method to GC with respect to energy value and runtime. We first observe that the results of GC are influenced by the number of chosen initial labels; in most yet not all cases, choosing more labels leads to a lower energy value but also increases the runtime; see Figures 3, 4. Figure 5 shows the value of the energy after each iteration of the proposed method for the results in Figure 4. In the corresponding experiments, we compare our approach to GC with a smaller and a larger number of initial labels (64 and 256). As data set we used the 200 test images of the Berkeley segmentation data set [62]. Table I reveals that the proposed method gives lower mean energies (w.r.t. the considered piecewise affine Mumford-Shah model) than GC for 4 out of 5 edge penalties. Further, the runtimes of the proposed method are lower for all edge penalties, most notably for the small ones, see Figure 6. Selected examples are shown in Figure 7. We refer to the supplementary for further results.

In all our experiments, we observed the convergence of the proposed scheme. We point out that for this kind of problems theoretical convergence results are rather weak in general. In particular, also for the state-of-the-art method based on iterative application of graph cuts (we compare with here) the convergence is observed only empirically as well. Further theoretical investigation of convergence is an interesting topic for future work.

IV. Discussion

We have studied the piecewise affine Mumford-Shah model which is a variational model for image partitioning. Determining approximate solutions of this model is a challenging problem. It requires a higher computational effort than employing classical non-variational clustering methods such as the k-means algorithm or the mean shift algorithm for partitioning. However, in contrast to these methods, the piecewise affine Mumford-Shah model incorporates the total length of the segment boundaries in its regularizing term and, on each segment, the input image is approximated by an affine function.

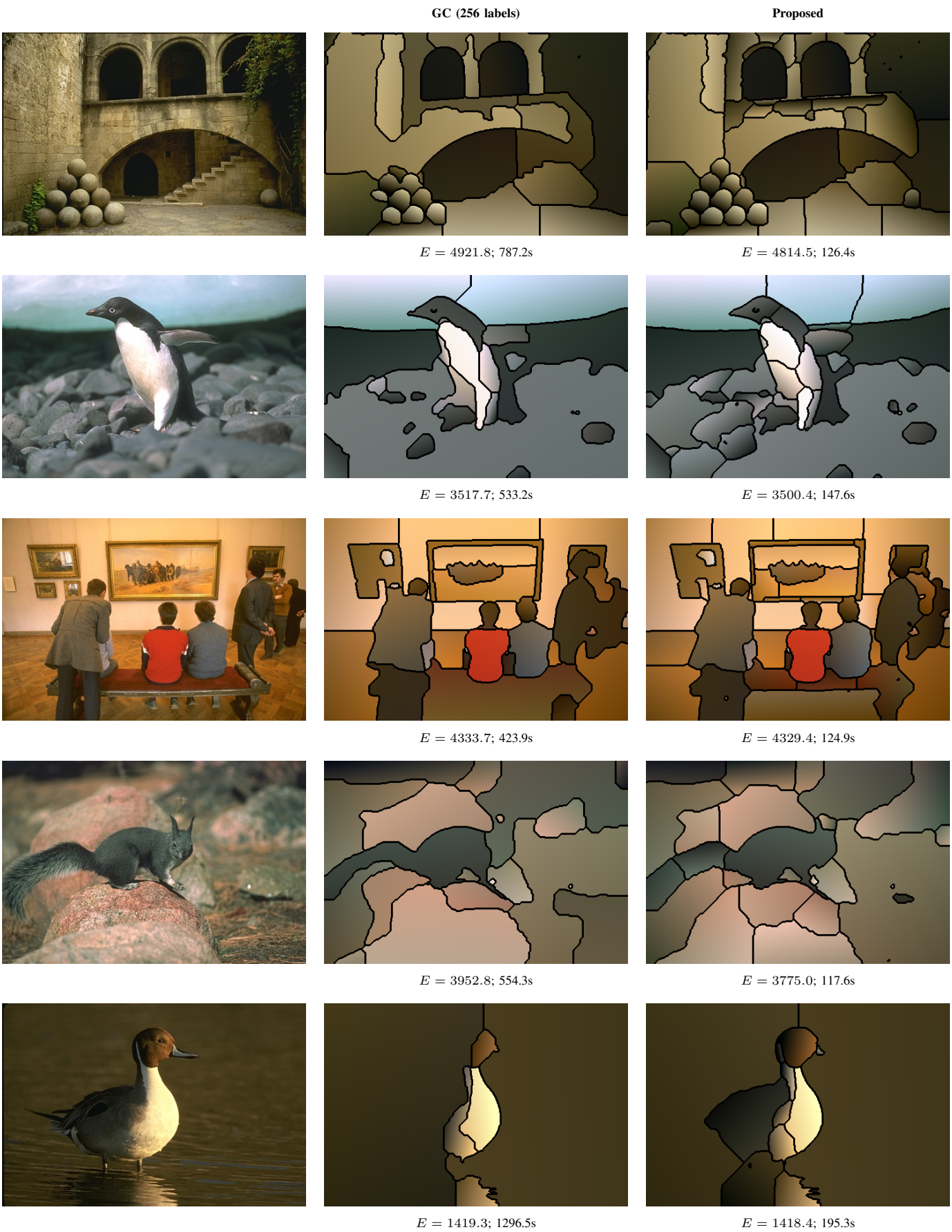


Fig. 7: **Example results for the Berkeley data set.** *From left to right:* five images from the BSD500, results of the graph cut approach (256 initial labels), and results of the proposed approach. The model parameter is $\gamma = 0.5$. Energies and runtimes are reported below the images.

We have proposed an efficient algorithm for computing approximate solutions of the piecewise affine Mumford-Shah model. A novel formulation of the problem in terms of Taylor jets allowed us to invoke a splitting into few tightly coupled jet-subproblems for which we developed an efficient and exact solver. We have seen that, in contrast to GNC and FPAME, the proposed method provides a genuine partition of the image. The experiments showed that the proposed approach yields results with mostly lower average model energies than those of the benchmark approach based on graph cuts. Hence, the model energy is efficiently minimized. At the same time, it needs less computation time than the benchmark method, especially for small edge penalties.

ACKNOWLEDGEMENT

This work was supported by the German Research Foundation (DFG STO1126/2-1 & WE5886/4-1).

APPENDIX

Pseudo Code for the ADMM Strategy

Algorithm 1: ADMM strategy for the piecewise affine-linear Mumford-Shah problem

Input: Image $f \in \mathbb{R}^{m\times n}$; model parameter $\gamma > 0$; neighborhood system $\{d^1, \dots, d^S\}$ and weights $\{\omega_1, \dots, \omega_S\}$; stopping paramter ξ; coupling sequences $(\mu_j)_{j\in\mathbb{N}}, (\nu_j)_{j\in\mathbb{N}}$

Output: Piecewise constant first order jet $J^* = (u^*, a^*, b^*)$

Initialize $J^s = (u^s, a^s, b^s) = (f, 0, 0)$ for all $s = 1, \dots, S$; $\lambda^{s,t}, \tau^{s,t}, \rho^{s,t} = 0$ for all $1 \le s < t \le S$

$j \leftarrow 1$

repeat

  for $s = 1, \dots, S$ **do**

    /* Compute data for jet subproblems */

    $w^s \leftarrow \frac{2f+\mu_j S\left(\sum_{t=s+1}^{S}\left(u^t - \frac{\lambda^{s,t}}{\mu_j}\right) + \sum_{r=1}^{s-1}\left(u^r + \frac{\lambda^{r,s}}{\mu_j}\right)\right)}{2+\mu_j S(S-1)}$

    $y^s \leftarrow \frac{1}{S-1}\sum_{t=s+1}^{S}\left(a^t - \frac{\tau^{s,t}}{\nu_j}\right) + \frac{1}{S-1}\sum_{r=1}^{s-1}\left(a^r + \frac{\tau^{r,s}}{\nu_j}\right)$

    $z^s \leftarrow \frac{1}{S-1}\sum_{t=s+1}^{S}\left(b^t - \frac{\rho^{s,t}}{\nu_j}\right) + \frac{1}{S-1}\sum_{r=1}^{s-1}\left(b^r + \frac{\rho^{r,s}}{\nu_j}\right)$

    /* Solve jet subproblems linewise w.r.t. direction d_s */

    $J^s \leftarrow \operatorname{argmin}_J \frac{2\omega_s\gamma}{(S-1)\nu_j}\|\nabla_{d_s} J\|_0 + \frac{2+\mu_j S}{\nu_j S}\|u - w^s\|^2 + \|a - y^s\|^2 + \|b - z^s\|^2$

  end

  /* Lagrange multiplier update */

  for $s = 1, \dots, S$ **do**

    for $t = s+1, \dots, S$ **do**

      $\lambda^{s,t} \leftarrow \lambda^{s,t} + \mu_j(u^s - u^t)$

      $\tau^{s,t} \leftarrow \tau^{s,t} + \nu_j(a^s - a^t)$

      $\rho^{s,t} \leftarrow \rho^{s,t} + \nu_j(b^s - b^t)$

    end

  end

  /* Increase coupling parameters */

  $\mu_j \leftarrow \mu_{j+1}$, $\nu_j \leftarrow \nu_{j+1}$

  $j \leftarrow j + 1$

until $\|J^1 - J^2\|_\infty < \xi$;

return $J^* = \frac{1}{S}\sum_{s=1}^{S} J^s$

Details for the Piecewise-Affine Mumford-Shah Model in the Multivariate Case

Discretized Problem: Starting from (30) in analogy to the derivation starting from (6) in the single channel case we obtain the discretized problem

$$J^* = \operatorname*{argmin}_{J:\Omega'\to(\Pi_1)^K} \gamma\sum_{s=1}^{S}\omega_s\|\nabla_{d^s} J\|_0 + \sum_{x\in\Omega'}\sum_k (J_k(x)(x) - f_k(x))^2. \tag{32}$$

Then the analogue of the splitting (10) in single channel case becomes

$$\operatorname*{argmin}_{J^1,\dots,J^S} \gamma\sum_s \omega_s\|\nabla_{d^s} J^s\|_0 + \frac{1}{S}\sum_{x\in\Omega'}\sum_k (J_k^s(x)(x) - f_k(x))^2 \quad \text{s.t. } J^s = J^t \quad \text{for all } 1 \le s < t \le S. \tag{33}$$

We rewrite (33) in the way of (11) which gives

$$\begin{aligned}\operatorname*{argmin}_{J^1,\dots,J^S} \sum_{s=1}^{S}\Big\{&\gamma\omega_s\|\nabla_{d^s} J^s\|_0 + \frac{1}{S}\sum_{x\in\Omega'}\sum_k (J_k^s(x)(x) - f_k(x))^2\Big\}\\ &\text{s.t. } J^s(x)(x) = J^t(x)(x),\\ &J^s(x)(x+e_1) - J^s(x)(x) = J^t(x)(x+e_1) - J^t(x)(x),\\ &J^s(x)(x+e_2) - J^s(x)(x) = J^t(x)(x+e_2) - J^t(x)(x),\\ &\text{for all } x \in \Omega', \text{and all } s,t \text{ with } 1 \le s < t \le S.\end{aligned} \tag{34}$$

The Lagrangian of (34) which corresponds to (13) in the univariate case is now understood w.r.t. the Lagrange multipliers $\lambda^{s,t}, \tau^{s,t}, \rho^{s,t} \in \mathbb{R}^{M\times N\times K}$ and the squared Frobenius norm $\|u\|^2 = \sum_{i,j,k} u_{ijk}^2$.

Univariate Subproblems.: The resulting univariate subproblems in (23) for a fixed line Q have now the prototypical form

$$\begin{aligned}&(J'_Q)^*\\ &= \operatorname*{argmin}_{J':Q\to(\Pi_1)^K} \gamma'|\{x \in Q : J'(x) \ne J'(x+d),\ x+d \in Q\}|\\ &\quad + \sum_{x\in Q}\sum_k \Big(\eta^2(u_k(x) - w_k(x))^2 + (a_k(x) - y_k(x))^2\\ &\quad + (b_k(x) - z_k(x))^2\Big).\end{aligned} \tag{35}$$

Concerning efficiently solving linewise segmented multichannel jet problems as demonstrated section 2.4 in the main text one has to mind that $\mathcal{E}^I$ denotes the sum over all channels k of the residual sum of squares of the best approximating bivariate polynomial, that is,

$$\mathcal{E}^I = \sum_k \min_{p\in\Pi_1} \eta^2(p(x) - w_k(x))^2 + (\alpha - y_k(x))^2 + (\beta - z_k(x))^2 \tag{36}$$

The least squares formulation of (36) on the discrete "interval" $(q_l, ..., q_r)$ is

$$\mathcal{E}^{\{q_l,\dots,q_r\}} = \sum_k \min_{\alpha,\beta,\delta} \|A^h(\alpha,\beta,\delta)^T - g^{lr}_{:,k}\|^2 \tag{37}$$

with A^h like in the main text (recall $h = r - l + 1$) and

$$g^{lr} = \begin{pmatrix} \eta w_{11} & \dots & \eta w_{1K} \\ y_{11} & \dots & y_{1K} \\ z_{11} & \dots & z_{1K} \\ \vdots & & \vdots \\ \eta w_{l1} & \dots & \eta w_{lK} \\ y_{l1} & \dots & y_{lK} \\ z_{l1} & \dots & z_{lK} \end{pmatrix}. \tag{38}$$

REFERENCES

[1] P. J. Besl and R. C. Jain, "Segmentation through variable-order surface fitting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 2, pp. 167–192, 1988.

[2] S. Beucher, "Use of watersheds in contour detection," in *Proceedings of the International Workshop on Image Processing*. CCETT, 1979.

[3] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 5, pp. 603–619, 2002.

[4] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on information theory*, vol. 21, no. 1, pp. 32–40, 1975.

[5] P. Milanfar, "A tour of modern image filtering: New insights and methods, both practical and theoretical," *IEEE signal processing magazine*, vol. 30, no. 1, pp. 106–128, 2012.

[6] J.-M. Morel and S. Solimini, *Variational methods in image segmentation: with seven image processing experiments*. Springer Science & Business Media, 2012, vol. 14.

[7] T. Chan and L. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.

[8] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[9] Z. Kato and T.-C. Pong, "A Markov random field image segmentation model for color textured images," *Image and Vision Computing*, vol. 24, no. 10, pp. 1103–1114, 2006.

[10] R. Ramlau and W. Ring, "A Mumford–Shah level-set approach for the inversion and segmentation of X-ray tomography data," *Journal of Computational Physics*, vol. 221, no. 2, pp. 539–557, 2007.

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

[12] A. Blake and A. Zisserman, *Visual reconstruction*. MIT Press Cambridge, 1987.

[13] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577–685, 1989.

[14] Y. Wang, S. Xiang, C. Pan, L. Wang, and G. Meng, "Level set evolution with locally linear classification for image segmentation," *Pattern Recognition*, vol. 46, no. 6, pp. 1734–1746, 2013.

[15] C. Chen, J. Leng, and G. Xu, "A general framework of piecewise-polynomial Mumford-Shah model for image segmentation," *International Journal of Computer Mathematics*, vol. 94, no. 10, pp. 1981–1997, 2017.

[16] D. Fortun, M. Storath, D. Rickert, A. Weinmann, and M. Unser, "Fast piecewise-affine motion estimation without segmentation," *IEEE Transactions on Image Processing*, To appear, preprint arXiv:1802.01872.

[17] W. Liu, W. Xu, X. Chen, X. Huang, C. Shen, and J. Yang, "Edge-preserving piecewise linear image smoothing using piecewise constant filters," *arXiv preprint arXiv:1801.06928*, 2018.

[18] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, 1984.

[19] D. Greig, B. Porteous, and A. Seheult, "Exact maximum a posteriori estimation for binary images," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 51, no. 2, pp. 271–279, 1989.

[20] L. Xu, C. Lu, Y. Xu, and J. Jia, "Image smoothing via L_0 gradient minimization," *ACM Transactions on Graphics (TOG)*, vol. 30, no. 6, p. 174, 2011.

[21] L. Xu, S. Zheng, and J. Jia, "Unnatural L_0 sparse representation for natural image deblurring," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 1107–1114.

[22] R. Nguyen and M. Brown, "Fast and effective L_0 gradient minimization by region fusion," in *IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 208–216.

[23] E. Strekalovskiy and D. Cremers, "Real-Time Minimization of the Piecewise Smooth Mumford-Shah Functional," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2014, pp. 127–141.

[24] 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.

[25] O. Veksler, "Efficient graph-based energy minimization methods in computer vision," Ph.D. dissertation, Cornell University, 1999.

[26] L. Ambrosio and V. Tortorelli, "Approximation of functional depending on jumps by elliptic functional via Γ-convergence," *Communications on Pure and Applied Mathematics*, vol. 43, no. 8, pp. 999–1036, 1990.

[27] A. Dufour, V. Shinin, S. Tajbakhsh, N. Guillén-Aghion, J.-C. Olivo-Marin, and C. Zimmer, "Segmenting and tracking fluorescent cells in dynamic 3-D microscopy with coupled active surfaces," *IEEE Transactions on Image Processing*, vol. 14, no. 9, pp. 1396–1410, 2005.

[28] C. Zimmer and J.-C. Olivo-Marin, "Coupled parametric active contours," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1838–1842, 2005.

[29] D. Cremers, M. Rousson, and R. Deriche, "A review of statistical approaches to level set segmentation: Integrating color, texture, motion and shape," *International Journal of Computer Vision*, vol. 72, no. 2, pp. 195–215, 2007.

[30] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient N-D image segmentation," *International Journal of Computer Vision*, vol. 70, no. 2, pp. 109–131, 2006.

[31] T. Pock, A. Chambolle, D. Cremers, and H. Bischof, "A convex relaxation approach for computing minimal partitions," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 810–817.

[32] A. Chambolle, D. Cremers, and T. Pock, "A convex approach to minimal partitions," *SIAM Journal on Imaging Sciences*, vol. 5, no. 4, pp. 1113–1158, 2012.

[33] J. Lellmann and C. Schnörr, "Continuous multiclass labeling approaches and algorithms," *SIAM Journal on Imaging Sciences*, vol. 4, no. 4, pp. 1049–1096, 2011.

[34] E. Strekalovskiy, A. Chambolle, and D. Cremers, "A convex representation for the vectorial Mumford-Shah functional," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 1712–1719.

[35] H. Hirschmüller, "Stereo processing by semiglobal matching and mutual information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2008.

[36] A. Drory, C. Haubold, S. Avidan, and F. A. Hamprecht, "Semi-global matching: a principled derivation in terms of message passing," in *Pattern Recognition*. Springer, 2014, pp. 43–53.

[37] X. Cheng, M. Zeng, and X. Liu, "Feature-preserving filtering with L_0 gradient minimization," *Computers & Graphics*, vol. 38, pp. 150–157, 2014.

[38] M. Fornasier and R. Ward, "Iterative thresholding meets free-discontinuity problems," *Foundations of Computational Mathematics*, vol. 10, no. 5, pp. 527–567, 2010.

[39] M. Storath and A. Weinmann, "Fast partitioning of vector-valued images," *SIAM Journal on Imaging Sciences*, vol. 7, no. 3, pp. 1826–1852, 2014.

[40] T. Pock, D. Cremers, H. Bischof, and A. Chambolle, "An algorithm for minimizing the Mumford-Shah functional," in *IEEE International Conference on Computer Vision (ICCV)*, 2009, pp. 1133–1140.

[41] M. Storath, D. Rickert, M. Unser, and A. Weinmann, "Fast segmentation from blurred data in 3d fluorescence microscopy," *IEEE Transactions on Image Processing*, vol. 26, no. 10, pp. 4856–4870, 2017.

[42] S. Birchfield and C. Tomasi, "Multiway cut for stereo and motion with slanted surfaces," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 1, 1999, pp. 489–495.

[43] A. Delong, L. Gorelick, F. R. Schmidt, O. Veksler, and Y. Boykov, "Interactive segmentation with super-labels," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2011, pp. 147–162.

[44] J. Yang and H. Li, "Dense, accurate optical flow estimation with piecewise parametric model," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1019–1027.

[45] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation," *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 492–526, 2010.

[46] R. Giryes, M. Elad, and A. M. Bruckstein, "Sparsity based methods for overparameterized variational problems," *SIAM Journal on Imaging Sciences*, vol. 8, no. 3, pp. 2133–2159, 2015.

[47] L. Xu, J. Ren, Q. Yan, R. Liao, and J. Jia, "Deep edge-aware filters," in *International Conference on Machine Learning*, 2015, pp. 1669–1678.

[48] H. Isack and Y. Boykov, "Energy-Based Geometric Multi-model Fitting," *Int J Comput Vis*, vol. 97, pp. 123–147, 2012.

[49] A. Delong, A. Osokin, H. N. Isack, Y. Boykov, A. Delong, H. N. Isack, Y. Boykov, and A. Osokin, "Fast Approximate Energy Minimization with Label Costs," *Int J Comput Vis*, vol. 96, pp. 1–27, 2012.

[50] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, 2004.

[51] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 2, pp. 147–159, 2004.

[52] A. Chambolle, "Image segmentation by variational methods: Mumford and Shah functional and the discrete approximations," *SIAM Journal on Applied Mathematics*, vol. 55, no. 3, pp. 827–863, 1995.

[53] ——, "Finite-differences discretizations of the Mumford-Shah functional," *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 33, no. 02, pp. 261–288, 1999.

[54] Y. Boykov and V. Kolmogorov, "Computing geodesics and minimal surfaces via graph cuts," in *Proceedings Ninth IEEE International Conference on Computer Vision*, 2003, pp. 26–33 vol.1.

[55] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[56] Y. Wang, W. Yin, and J. Zeng, "Global convergence of ADMM in nonconvex nonsmooth optimization," *Journal of Scientific Computing*, 2018.

[57] Z. Xu, S. De, M. Figueiredo, C. Studer, and T. Goldstein, "An empirical study of ADMM for nonconvex problems," *arXiv preprint arXiv:1612.03349*, 2016.

[58] R. Chartrand and B. Wohlberg, "A nonconvex ADMM algorithm for group sparsity with sparse groups," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6009–6013.

[59] G. Winkler and V. Liebscher, "Smoothers for discontinuous signals," *Journal of Nonparametric Statistics*, vol. 14, no. 1-2, pp. 203–222, 2002.

[60] F. Friedrich, A. Kempe, V. Liebscher, and G. Winkler, "Complexity penalized M-estimation," *Journal of Computational and Graphical Statistics*, vol. 17, no. 1, pp. 201–224, 2008.

[61] M. Storath, L. Kiefer, and A. Weinmann, "Smoothing for signals with discontinuities using higher order Mumford-Shah models," *arXiv preprint arXiv:1803.06156*, 2018.

[62] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.

Lukas Kiefer received the B.S. degree in business mathematics from Universität Mannheim and the M.S. degree in mathematics from Universität Heidelberg in 2014 and 2017, respectively. He is currently pursuing the Ph.D. degree in mathematics with the Mathematical Imaging Group at Universität Heidelberg and is further affiliated with the Hochschule Darmstadt. His research interests include image processing, variational methods, and inverse problems.

Martin Storath received his Diploma degree in Mathematics in 2008, his Honours degree in Technology Management in 2009, and his doctoral degree in Mathematics in 2013, all from Technische Universität München. He worked as researcher at the Helmholtz Zentrum München, at the Biomedical Imaging Group, EPFL, and at the Image Analysis and Learning Group, Universität Heidelberg. Currently, he is a professor at the University of Applied Sciences Würzburg-Schweinfurt. His research interests include image processing, variational methods, and inverse problems.

Andreas Weinmann received his Diploma degree in mathematics and computer science from Technische Universität München in 2006 and his Ph.D. degree from Technische Universität Graz in 2010 (both with highest distinction). He worked as a researcher at Technische Universtät München. Currently, he is affiliated with the Helmholtz Center Munich and the Hochschule Darmstadt. His research interests are applied analysis as well as signal and image processing.