J Math Imaging Vis
DOI 10.1007/s10851-016-0702-4

Image Labeling by Assignment

Freddie Åström[1] · **Stefania Petra**[2] · **Bernhard Schmitzer**[3] · **Christoph Schnörr**[4]

Received: 25 March 2016 / Accepted: 30 December 2016

Abstract We introduce a novel geometric approach to the image labeling problem. Abstracting from specific labeling applications, a general objective function is defined on a manifold of stochastic matrices, whose elements assign prior data that are given in any metric space, to observed image measurements. The corresponding Riemannian gradient flow entails a set of replicator equations, one for each data point, that are spatially coupled by geometric averaging on the manifold. Starting from uniform assignments at the barycenter as natural initialization, the flow terminates at some global maximum, each of which corresponds to an image labeling that uniquely assigns the prior data. Our geometric variational approach constitutes a smooth non-convex inner approximation of the general image labeling problem, implemented with sparse interior-point numerics in terms of parallel multiplicative updates that converge efficiently.

Keywords Image labeling · Assignment manifold · Fisher–Rao metric · Riemannian gradient flow · Replicator equations · Information geometry · Neighborhood filters · Nonlinear diffusion

Mathematics Subject Classification 62H35 · 65K05 · 68U10 · 62M40

✉ Christoph Schnörr
schnoerr@math.uni-heidelberg.de

Freddie Åström
freddie.astroem@iwr.uni-heidelberg.de

Stefania Petra
petra@math.uni-heidelberg.de

Bernhard Schmitzer
schmitzer@ceremade.dauphine.fr

[1] Heidelberg Collaboratory for Image Processing, Heidelberg University, Heidelberg, Germany

[2] Mathematical Imaging Group, Heidelberg University, Heidelberg, Germany

[3] CEREMADE, University Paris-Dauphine, Paris, France

[4] Image and Pattern Analysis Group, Heidelberg University, Heidelberg, Germany

1 Introduction

1.1 Motivation

Image Labeling is a basic problem of variational low-level image analysis. It amounts to determining a *partition* of the image domain by uniquely assigning to each pixel a single element from a finite set of labels. Most applications require such decisions to be made depending on other decisions. This gives rise to a global objective function whose minima correspond to favorable label assignments and partitions. Because the problem of computing globally optimal partitions generally is NP hard, *relaxations* of the variational problem only define computationally feasible optimization approaches.

Continuous Models and relaxations of the image labeling problem were studied, e.g., in [13,32], including the specific binary case, where two labels are only assigned [14] and the convex relaxation is tight, such that the global optimum can be determined by convex programming. *Discrete models* prevail in the field of computer vision. They lead to polyhedral relaxations of the image partitioning problem that are tighter than those obtained from continuous models after discretization. We refer to [22] for a comprehensive survey and evaluation. Similar to the continuous case, the binary partition problem can be efficiently and globally optimal solved using a subclass of binary discrete models [29].

Relaxations of the variational image labeling problem fall into two categories: *convex and non-convex relaxations*. The

Published online: 12 January 2017

dominant *convex approach* is based on the local polytope relaxation, a particular linear programming (LP-) relaxation [49]. This has spurred a lot of research on developing specific algorithms for efficiently solving large problem instances, as they often occur in applications. We mention [28] as a prominent example and otherwise refer again to [22]. Yet, models with higher connectivity in terms of objective functions with local potentials that are defined on larger cliques are still difficult to solve efficiently. A major reason that has been largely motivating our present work is the *non-smoothness* of optimization problems resulting from convex relaxation—the price to pay for convexity.

Major classes of *non-convex relaxations* are based on the mean-field approach [39], [47, Section 5] or on approximations of the intractable entropy of the probability distribution whose negative logarithm equals the functional to be minimized [50]. Examples for early applications of relaxations of the former approach include [15,18]. The basic instance of the latter class of approaches is known as the Bethe approximation. In connection with image labeling, all these approaches amount to *non-convex inner* relaxations of the combinatorially complex set of feasible solutions (the so-called marginal polytope), in contrast to the *convex outer* relaxations in terms of the local polytope discussed above. As a consequence, the non-convex approaches provide a mathematically valid basis for *probabilistic inference* like computing marginal distributions, which in principle enables a more sophisticated data analysis than mere energy minimization or maximum a posteriori inference, to which energy minimization corresponds from a probabilistic viewpoint.

On the other hand, like non-convex optimization problems in general, these relaxations are plagued by the problem of avoiding poor local minima. Although attempts were made to tame this problem by local convexification [16], the class of *convex* relaxation approaches has become dominant in the field, because the ability to solve the relaxed problem for a global optimum is a much better basis for research on algorithms and also results in more reliable software for users and applications.

Both classes of convex and non-convex approaches to the image labeling problem motivate the present work as an attempt to address the following two issues.

– **Smoothness versus Non-Smoothness** Regarding convex approaches and the development of efficient algorithms, a major obstacle stems from the inherent non-smoothness of the corresponding optimization problems. This issue becomes particularly visible in connection with decompositions of the optimization task into simpler problems by dropping complicating constraints, at the cost of a non-smooth dual master problem where these constraints have to be enforced. Advanced bundle methods [23] then seem to be among the most efficient methods. Yet, how to make rapid progress in systematic way does not seem obvious.
On the other hand, since the early days of linear programming, e.g., [4,5], it has been known that endowing the feasible set with a proper *smooth* geometry enables efficient numerics. Yet, such *interior-point* methods [38] are considered as not applicable for large-scale problems of variational image analysis, due to dense numerical linear algebra steps that are both too expensive and too memory intensive.

  In view of these aspects, **our approach** may be seen as a *smooth geometric approach* to image labeling based on *first-order, sparse* numerical operations.
– **Local versus Global Optimality** Global optimality distinguishes convex approaches from other ones and is the major argument for the former ones. Yet, having computed a global optimum of the relaxed problem, it has to be projected to the feasible set of combinatorial solutions (labelings) in a post-processing step. While the inherent suboptimality of this step can be bounded [31], and despite progress has been made to recover the true combinatorial optimum as least partially [46], it is clear that the benefit of global optimality of convex optimization has to be relativized when it constitutes a relaxation of an intractable optimization problem. Turning to non-convex problems, on the other hand, raises the two well-known issues: local optimality of solutions instead of global optimality, and susceptibility to initialization.
  In view of these aspects, **our approach** enjoys the following properties. While being non-convex, there is a *single natural* initialization only which makes obsolete the need to search for a good initialization. Furthermore, the approach returns a *global* optimum (out of many), which corresponds to an image labeling (combinatorial solution) without the need of further post-processing.
  Clearly, the latter property is typical for concave minimization formulations of combinatorial optimization problems [19] where solutions of the latter problem are enforced by weighting the concave penalty sufficiently large. Yet, in such cases, and in particular so when working in high dimensions as in image analysis, the problem persists to determine good initializations and to carefully design the numerics (search direction, step-size selection, etc.), in order to ensure convergence and a reasonable convergence rate.

1.2 Approach: Overview

Figure 1 illustrates our setup and the approach. We distinguish the feature space $\mathscr{F}$ that models all application-specific aspects, and the assignment manifold $\mathscr{W}$ used for mod-

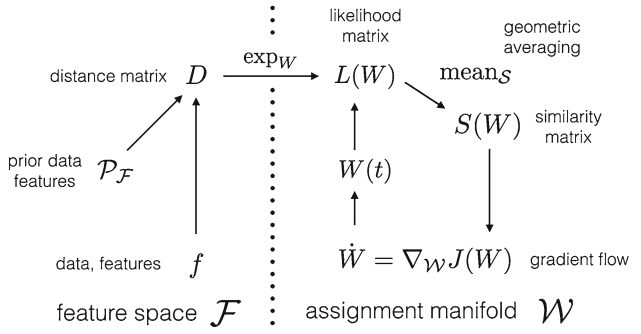


Fig. 1 Overview of the variational approach. Given data and prior features in a metric space $\mathcal{F}$, inference corresponds to a Riemannian gradient flow with respect to an objective function $J(W)$ on the assignment manifold $\mathcal{W}$. The curve of matrices $W(t)$ assigns at each t prior data $\mathcal{P}_\mathcal{F}$ to observed data f and terminates at a global maximum W^* that constitutes a labeling, i.e., a unique assignment of a single prior datum to each data point. Spatial coherence of the labeling field is enforced by geometric averaging over spatial neighborhoods. The entire dynamic process on the assignment manifold achieves a MAP labeling in a smooth, geometrical setting, realized with sparse interior-point numerics in terms of parallel multiplicative updates

eling the image labeling problem and for computing a solution. This distinction avoids to mix up physical dimensions, specific data formats, etc., with the representation of the inference problem. It ensures broad applicability to any application domain that can be equipped with a metric which properly reflects data similarity. It also enables to normalize the representation used for inference, so as to remove any bias toward a solution *not* induced by the data at hand.

We consider *image labeling* as the task to assign to the image data an arbitrary prior data set $\mathcal{P}_\mathcal{F}$, provided the distance of its elements to any given data element can be measured by a distance function $d_\mathcal{F}$, which the user has to supply. Basic examples for the elements of $\mathcal{P}_\mathcal{F}$ include prototypical feature vectors, patches, etc. Collecting all pairwise distance data into a distance matrix D, which could be computed on the fly for extremely large problem sizes, provides the input data to the inference problem.

The mapping $\exp_W$ lifts the distance matrix to the assignment manifold $\mathcal{W}$. The resulting likelihood matrix L constitutes a normalized version of the distance matrix D that reflects the initial feature space geometry as given by the distance function $d_\mathcal{F}$. Each point on $\mathcal{W}$, like the matrices L, S, and W, is *stochastic matrix* with strictly positive entries, that is, with row vectors that are discrete probability distributions having full support. Each such row vector indexed by i represents the *assignment* of prior elements of $\mathcal{P}_\mathcal{F}$ to the given datum a location i, in other words the *labeling* of datum i. We equip the set of all such matrices with the geometry induced by the Fisher–Rao metric and call it *assignment manifold*.

The inference task (image labeling) is accomplished by *geometric averaging* in terms of Riemannian means of assignment vectors over spatial neighborhoods. This step transforms the likelihood matrix L into the similarity matrix S. It also induces a dependency of labeling decisions on each other, akin to the prior (regularization) terms of the established variational approaches to image labeling, as discussed in the preceding section. These dependencies are resolved by maximizing the correlation (inner product) between the assignment in terms of the matrix W and the similarity matrix S, where the latter matrix is induced by W as well. The Riemannian gradient flow of the corresponding objective function $J(W)$, that is highly nonlinear but smooth, evolves $W(t)$ on the manifold $\mathcal{W}$ until a fixed point is reached which terminates the loop on the right-hand side of Fig. 1. The resulting fixed point corresponds to an *image labeling* which *uniquely* assigns to each datum a prior element of $\mathcal{P}_\mathcal{F}$.

Adopting a probabilistic Bayesian viewpoint, this fixed-point iteration may be viewed as maximum a posteriori inference carried out in a geometric setting with multiplicative, sparse, and highly parallel numerical operations.

1.3 Further Related Work

Besides current research on image labeling, there are further classes of approaches that resemble our approach. We briefly sketch each of them in turn and highlight similarities and differences.

Neighborhood Filters. A large class of approaches to *denoising* of given image data f are defined in terms of neighborhood filters that iteratively perform operations of the form

$$u_i^{(k+1)} = \sum_j \frac{K\left(x_i, x_j, u_i^{(k)}, u_j^{(k)}\right)}{\sum_l K\left(x_i, x_l, u_i^{(k)}, u_l^{(k)}\right)} u_j^{(k)}, \quad u(0) = f, \quad \forall i, \tag{1.1}$$

where K is a nonnegative kernel function that is symmetric with respect to the two indexed locations (e.g., i, j in the numerator) and may depend on both the spatial distance $\|x^i - x^j\|$ and the values $|u_i - u_j|$ of pairs of pixels. Maybe the most prominent example is the nonlocal means filter [9] where K depends on the distance of *patches* centered at i and j, respectively. We refer to [35] for a recent survey.

Noting that (1.1) is a linear operation with a row-normalized nonnegative (i.e., stochastic) matrix, a similar situation would be

$$u_i = \sum_j L_{ij}(W) u_j, \tag{1.2}$$

with the likelihood matrix from Fig. 1, if we would replace the prior data $\mathcal{P}_\mathcal{F}$ with the given image data f itself and

adopt a distance function $d_{\mathscr{F}}$, in order to mimic the kernel function K of (1.1).

In our approach, however, the likelihood matrix along with its nonlinear geometric transformation, the similarity matrix $S(W)$, evolves along with the evolution of assignment matrix W, so as to determine a labeling with *unique* assignments to each pixel i, rather than convex combinations as required for denoising. Furthermore, the prior data set $\mathscr{P}_{\mathscr{F}}$ that is assigned in our case may be very different from the given image data and, accordingly, the assignment matrix may have any rectangular shape rather than being a quadratic $m \times m$ matrix.

Conceptually, we are concerned with *decision making* (labeling, partitioning, unique assignments) rather than with mapping one image to another one. Whenever the prior data $\mathscr{P}_{\mathscr{F}}$ comprise a finite set of *prototypical* image values or patches, such that a mapping of the form

$$u_i = \sum_j W_{ij} f_j^*, \qquad f_j^* \in \mathscr{P}_{\mathscr{F}}, \qquad \forall i, \tag{1.3}$$

is well defined, then this does result in a transformed image u after having reached a fixed point of the evolution of W. This result then should not be considered as a denoised image, however. Rather, it merely illustrates the interpretation of the given data f in terms of the prior data $\mathscr{P}_{\mathscr{F}}$ and a corresponding optimal assignment.

Nonlinear Diffusion. Neighborhood filters are closely related to iterative algorithms for numerically solving discretized diffusion equations. Just think of the basic 5-point stencil of the discrete Laplacian, the iterative averaging of nearest neighbor differences, and the large class of adaptive generalizations in terms of nonlinear diffusion filters [48]. More recent work directly addressing this connection includes [10,36,44]. The author of [36], for instance, advocates the approximation of the matrix of (1.1) by a *symmetric* (hence, doubly stochastic) *positive-definite* matrix, in order to enable interpretations of the denoising operation in terms of the spectral decomposition of the assignment matrix, and to make the connection to diffusion mappings on graphs.

The connection to our work is implicitly given by the discussion of the previous point, the relation of our approach to neighborhood filters. Roughly speaking, the application of our approach in the *specific* case of assigning image data to image data may be seen as some kind of nonlinear diffusion that results in an image whose degrees of freedom are given by the cardinality of the prior set $\mathscr{P}_{\mathscr{F}}$. We plan to explore the exact nature of this connection in more detail in our future work.

Replicator Dynamics. Replicator dynamics and the corresponding equations are well known [17]. They play a major role in models of various disciplines, including theoretical biology and applications of game theory to economy. In the field of image analysis, such models have been promoted by Pelillo and co-workers, mainly to efficiently determine by continuous optimization techniques good local optima of intractable problems, like matchings through maximum-clique search in an association graph [42]. Although the corresponding objective functions are merely quadratic, the analysis of the corresponding equations is rather involved [8]. Accordingly, clever heuristics have been suggested to tame related problems of non-convex optimization [7].

Regarding our approach, we aim to get rid of these issues—see the discussion of "Global optimality" in Sect. 1.1—through three ingredients: (1) a unique natural initialization, (2) spatial averaging that removes spurious local affects of noisy data, and (3) adopting the Riemannian geometry which determines the structure of the replicator equations, for both geometric spatial averaging and numerical optimization.

Relaxation Labeling. The task of labeling primitives in images has been formulated as a problem of contextual decision making already 40 years ago [20,43]. Originally, update rules were merely formulated in order to find mutually consistent individual label assignments. Subsequent research related these labeling rules to optimization tasks. We refer to [41] for a concise account of the literature and for putting the approach on mathematically solid ground. Specifically, the so-called Baum–Eager theorem was applied in order to show that updates increase the mutual consistency of label assignments. Applications include pairwise clustering [40] that boils down to determining a local optimum by continuous optimization of a non-convex quadratic form, similar to the optimization tasks considered in [8,42]. We attribute the fact that these approaches have not been widely applied to the problems of non-convex optimization discussed above.

The measure of mutual consistency of our approach is non-quadratic, and the Baum–Eager theorem about polynomial growth transforms does not apply. Increasing consistency follows from the Riemannian gradient flow that governs the evolution of label assignments. Regarding the non-convexity from the viewpoint of optimization, we believe that the setup of our approach displayed by Fig. 1 significantly alleviates these problems, in particular through the geometric averaging of assignments that emanates from a natural initialization.

We address again some of these points that are relevant for our future work, in Sect. 5.

1.4 Organization

Section 2 summarizes the geometry of the probability simplex in order to define the assignment manifold, which is the basis of our variational approach. The approach is presented in Sect. 3 by repeating the discussion of Fig. 1, together with the mathematical details. Finally, several numerical experiments are reported in Sect. 4. They are academical, yet non-trivial, and supposed to illustrate properties of the approach as claimed in the preceding sections. Specific applications of image labeling are not within the scope of this paper. We conclude and indicate further directions of research in Sect. 5.

Major symbols and the basic notation used in this paper are listed in "Appendix 1." In order not to disrupt the flow of reading and reasoning, proofs, and technical details, all of which are elementary but essentially complement the presentation and make this paper self-contained, are listed as "Appendix 2."

2 The Assignment Manifold

In this section, we define the feasible set for representing and computating image labelings in terms of assignment matrices $W \in \mathcal{W}$, the assignment manifold $\mathcal{W}$. The basic building block is the open probability simplex $\mathcal{S}$ equipped with the Fisher–Rao metric. We collect below and in "Proofs of Section 2 of Appendix 2" corresponding definitions and properties.

For background reading and much more details on information and Riemannian geometry, we refer to [1,21].

2.1 Geometry of the Probability Simplex

The relative interior $\mathcal{S} = \mathring{\Delta}_{n-1}$ of the probability simplex given by (6.8a) becomes a differentiable Riemannian manifold when endowed with the Fisher–Rao metric. In the present particular case, it reads (cf. the notation (6.16))

$$\langle u, v\rangle_p := \Big\langle \frac{u}{\sqrt{p}}, \frac{v}{\sqrt{p}} \Big\rangle, \qquad \forall u, v \in T_p\mathcal{S}, \tag{2.1}$$

with tangent spaces given by

$$T_p\mathcal{S} = \{v \in \mathbb{R}^n : \langle \mathbb{1}, v\rangle = 0\}, \qquad p \in \mathcal{S}. \tag{2.2}$$

We regard the scaled sphere $\mathcal{N} = 2\mathbb{S}^{n-1}$ as manifold with Riemannian metric induced by the Euclidean inner product of $\mathbb{R}^n$. The following diffeomorphism ψ between $\mathcal{S}_n$ and the open subset $\psi(\mathcal{S}_n) \subset \mathcal{N}$ was suggested, e.g., by [27, Section 2.1] and [1, Section 2.5].

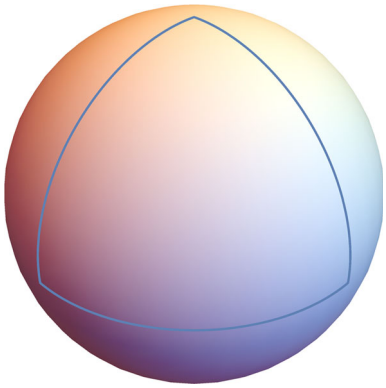


Fig. 2 The *Triangle* encloses the image $\psi(\mathcal{S}_2) \subset 2\mathbb{S}^2$ of the simplex $\mathcal{S}_2$ under the sphere map (2.3)

Definition 1 (*Sphere Map*) We call the diffeomorphism

$$\psi : \mathcal{S} \to \mathcal{N}, \qquad p \mapsto s = \psi(p) := 2\sqrt{p}, \tag{2.3}$$

sphere map (see Fig. 2).

The sphere map enables to compute the geometry of $\mathcal{S}$ from the geometry of the 2-sphere.

Lemma 1 *The sphere map ψ (2.3) is an isometry, i.e., the Riemannian metric is preserved. Consequently, lenghts of tangent vectors and curves are preserved as well.*

Proof See "Proofs of Section 2" in Appendix 2. □

In particular, geodesics as critical points of length functionals are mapped by ψ to geodesics. As a consequence, we have

Lemma 2 [Riemannian Distance on $\mathcal{S}$] *The Riemannian distance on $\mathcal{S}$ is given by*

$$d_{\mathcal{S}}(p, q) = 2 \arccos\Big(\sum_{i\in[n]} \sqrt{p_i q_i} \Big) \in [0, \pi). \tag{2.4}$$

The objective function for computing Riemannian means (geometric averaging; see Definition 2 and Eq. (2.8) below) is based on the distance (2.4). Figure 3 visualizes corresponding geodesics and level sets on $\mathcal{S}_3$ that differ for discrete distributions $p \in \mathcal{S}_3$ close to the barycenter and for low-entropy distributions close to the vertices. See also the caption of Fig. 3.

It is well known from the literature (e.g., [3,30]) that geometries may considerably change in higher dimensions. Figure 4 displays the Riemannian distances of points on curves that connect the barycenter and vertices on $\overline{\mathcal{S}}_n$ (to which the distance (2.4) extends), depending on the dimension n. The normalizing effect on geometric averaging, further discussed in the caption, increases with n and is relevant to image labeling, where large values of n may occur in applications.

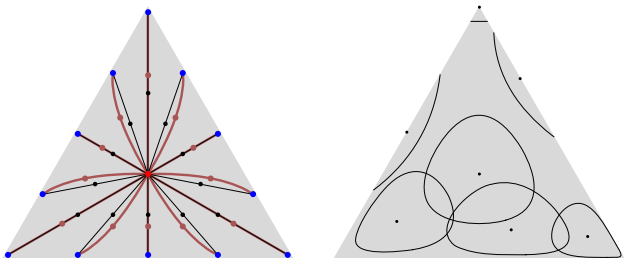


Fig. 3 Geometry of the probability simplex induced by the Fisher–Rao Metric. The *left panel* shows Euclidean (*black*) and non-Euclidean geodesics (*brown*) connecting the barycenter (*red*) and the *blue points*, along with the corresponding Euclidean and Riemannian means: In comparison with Euclidean means, geometric averaging pushes toward the boundary. The *right panel* shows contour lines of points that have the same Riemannian distance from the respective center point (*black dots*). The different sizes of these regions indicate that geometric averaging causes a larger effect around the barycenters of both the simplex and its faces, where such points represent fuzzy labelings, and a smaller effect within regions close to the vertices (unit vectors)

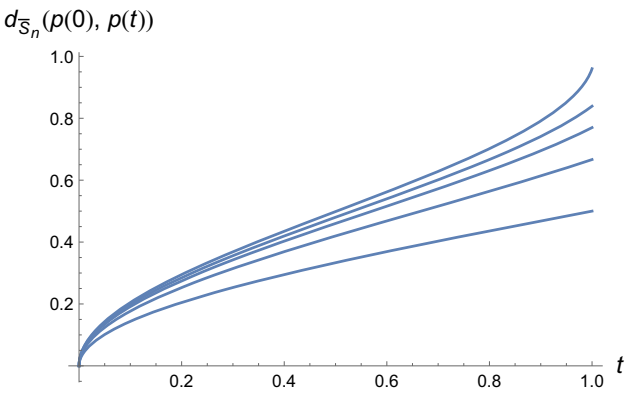


Fig. 4 Each curve, from *bottom to top*, represents the Riemannian distances $d_{\overline{\mathcal{S}}_n}\big(p(0), p(t)\big)$ (normalized to [0,1]; Eq. (2.4)) of points on the curve $\{p(t)\}_{t\in[0,1]}$ that linearly (i.e., Euclidean) interpolates between the fixed vertex $p(0) = e^1$ of the simplex $\overline{\mathcal{S}}_n = \Delta_{n-1}$ and the barycenter $p(1) = \overline{p}$, for dimensions $n = 2^k$, $k \in \{1, 2, 3, 4, 8\}$. As the dimension n grows, the barycenter is located as far away from e^1 as all other boundary points e^i, $te^i + (1-t)e^j$, $t \in [0, 1]$, $i, j \neq 1$, etc., which have *disjoint* supports. This entails a normalizing effect on the Riemannian mean of points that are far away, unlike with Euclidean averaging where this influence increases with the Euclidean distance

Let $\mathscr{M}$ be a smooth Riemannian manifold (see the paragraph around Eq. (6.14) introducing our notation). The Riemannian gradient $\nabla_{\mathscr{M}} f(p) \in T_p\mathscr{M}$ of a smooth function $f : \mathscr{M} \to \mathbb{R}$ at $p \in \mathscr{M}$ is the tangent vector defined by [21, p. 89]

$$\langle \nabla_{\mathscr{M}} f(p), v\rangle_p = Df(p)[v] = \langle \nabla f(p), v\rangle, \quad \forall v \in T_p\mathscr{M}. \tag{2.5}$$

We consider next the specific case $\mathscr{M} = \mathscr{S} = \mathscr{S}_n$.

Proposition 1 *(Riemannian Gradient on $\mathscr{S}_n$) For any smooth function $f : \mathscr{S} \to \mathbb{R}$, the Riemannian gradient of f at $p \in \mathscr{S}$ is given by*

$$\nabla_{\mathscr{S}} f(p) = p\big(\nabla f(p) - \langle p, \nabla f(p)\rangle \mathbb{1}\big). \tag{2.6}$$

Proof See "Proofs of Section 2" in Appendix 2. □

The exponential map associated with the open probability simplex $\mathscr{S}$ is detailed next.

Proposition 2 *(Exponential Map (Manifold $\mathscr{S}$)) The exponential mapping*

$$\mathrm{Exp}_p : V_p \to \mathscr{S}, \quad v \mapsto \mathrm{Exp}_p(v) = \gamma_v(1), \quad p \in \mathscr{S}, \tag{2.7a}$$

is given by

$$\gamma_v(t) = \frac{1}{2}\Big(p + \frac{v_p^2}{\|v_p\|^2}\Big) + \frac{1}{2}\Big(p - \frac{v_p^2}{\|v_p\|^2}\Big)\cos\big(\|v_p\|t\big) \tag{2.7b}$$

$$+ \frac{v_p}{\|v_p\|}\sqrt{p}\sin\big(\|v_p\|t\big), \tag{2.7c}$$

with $t = 1$, $v_p = v/\sqrt{p}$, $p = \gamma(0)$, $\dot\gamma_v(0) = v$ and

$$V_p = \big\{v \in T_p\mathscr{S} : \gamma_v(t) \in \mathscr{S},\ t \in [0, 1]\big\}. \tag{2.7d}$$

Proof See "Proofs of Section 2" of Appendix 2. □

Remark 1 Checking the inclusion $v \in V_p$ due to (2.7d), for a given tangent vector $v \in T_p\mathscr{S}$, is inconvenient for applications. Therefore, the mapping exp is defined below by Eq. (3.8a) which approximates the exponential mapping Exp, with the feasible set V_p replaced by the entire space $T_p\mathscr{S}$ (Lemma 3).

Accordingly, geometric averaging as defined next (Sect. 2.2) based on Exp can be approximated as well using the mapping exp. This is discussed in Sect. 3.3.2.

2.2 Riemannian Means

The *Riemannian center of mass* is commonly called *Karcher mean* or *Fréchet mean* in the more recent literature, in particular outside the field of mathematics. We prefer—cf. [26]—the former notion and use the shorter term *Riemannian mean*.

Definition 2 (*Riemannian Mean, Geometric Averaging*) The *Riemannian mean* $\overline{p}$ of a set of points $\{p^i\}_{i\in[N]} \subset \mathscr{S}$ with corresponding weights $w \in \Delta_{N-1}$ minimizes the objective function

$$p \mapsto \frac{1}{2}\sum_{i\in[N]} w_i d^2_{\mathscr{S}}(p, p^i) \tag{2.8}$$

and satisfies the optimality condition [21, Lemma 4.8.4]

$$\sum_{i\in[N]} w_i \,\mathrm{Exp}_{\overline{p}}^{-1}(p^i) = 0, \tag{2.9}$$

with the inverse of the exponential mapping $\mathrm{Exp}_p^{-1}\colon \mathcal{S} \to T_p\mathcal{S}$. We denote the Riemannian mean by

$$\mathrm{mean}_{\mathcal{S},w}(\mathcal{P}), \qquad w \in \Delta_{N-1}, \quad \mathcal{P} = \{p^1, \ldots, p^N\}, \tag{2.10}$$

and drop the subscript w in the case of uniform weights $w = \frac{1}{N}\mathbb{1}_N$.

Lemma 3 *The Riemannian mean* (2.10) *defined as minimizer of* (2.8) *is unique for any data* $\mathcal{P} = \{p^i\}_{i\in[N]} \subset \mathcal{S}$ *and weights* $w \in \Delta_{N-1}$.

Proof Using the isometry ψ given by (2.3), we may consider the scenario transferred to the domain on the 2-sphere depicted in Fig. 2. Due to [25, Thm. 1.2], the objective (2.8) is convex along geodesics and has a unique minimizer within any geodesic Ball $\mathbb{B}_r$ with diameter upper bounded by $2r \leq \frac{\pi}{2\sqrt{\kappa}}$, where κ upper bounds the sectional curvatures in $\mathbb{B}_r$. For the 2-sphere $\mathcal{N}$, we have $\kappa = 1/4$ constant, and hence the inequality is satisfied for the domain $\psi(\mathcal{S}) \subset \mathcal{N}$ which has geodesic diameter π. □

We call the computation of Riemannian means *geometric averaging*. The implementation of this iterative operation and its efficient approximation by a closed-form expression are addressed in Sect. 3.3.

2.3 Assignment Matrices and Manifold

A natural question is how to extend the geometry of $\mathcal{S}$ to stochastic matrices $W \in \mathbb{R}^{m\times n}$ with $W_i \in \mathcal{S}$, $i \in [m]$, so as to preserve the information-theoretic properties induced by this metric (that we do not discuss here—cf. [1,12]).

This problem was recently studied by [37]. The authors suggested three natural definitions of manifolds. It turned out that all of them are slight variations of taking the product of $\mathcal{S}$, differing only by the scaling of the resulting product metric. As a consequence, we make the following

Definition 3 (*Assignment Manifold*) The manifold of assignment matrices, called *assignment manifold*, is the set

$$\mathcal{W} = \{W \in \mathbb{R}^{m\times n}\colon W_i \in \mathcal{S},\ i \in [m]\}. \tag{2.11}$$

According to this product structure and based on (2.1), the Riemannian metric is given by

$$\langle U, V\rangle_W := \sum_{i\in[m]} \langle U_i, V_i\rangle_{W_i}, \qquad U, V \in T_W\mathcal{W}. \tag{2.12}$$

Note that $V \in T_W\mathcal{W}$ means $V_i \in T_{W_i}\mathcal{S}$, $i \in [m]$.

Remark 2 We call stochastic matrices contained in $\mathcal{W}$ *assignment matrices*, due to their role in the variational approach (Sect. 3).

3 Variational Approach

We introduce in this section the basic components of the variational approach and the corresponding optimization task, as illustrated in Fig. 1.

3.1 Basic Components

3.1.1 Features, Distance Function, Assignment Task

Let

$$f\colon \mathcal{V} \to \mathcal{F}, \qquad i \mapsto f_i, \qquad i \in \mathcal{V} = [m], \tag{3.1}$$

denote any given data, either raw image data or features extracted from the data in a preprocessing step. In any case, we call f *feature*. At this point, we do not make any assumption about the *feature space* $\mathcal{F}$ except that a *distance function*

$$d_{\mathcal{F}}\colon \mathcal{F} \times \mathcal{F} \to \mathbb{R}, \tag{3.2}$$

is specified. We assume that a finite subset of $\mathcal{F}$

$$\mathcal{P}_{\mathcal{F}} := \{f_j^*\}_{j\in[n]}, \tag{3.3}$$

additionally is given, called *prior set*. We are interested in the assignment of the prior set to the data in terms of an *assignment matrix*

$$W \in \mathcal{W} \subset \mathbb{R}^{m\times n}, \tag{3.4}$$

with the manifold $\mathcal{W}$ defined by (2.11). Thus, by definition, every row vector $0 < W_i \in \mathcal{S}$ is a discrete distribution with full support $\mathrm{supp}(W_i) = [n]$. The element

$$W_{ij} = \Pr(f_j^*|f_i), \qquad i \in [m], \quad j \in [n], \tag{3.5}$$

quantifies the assignment of prior item f_j^* to the observed data point f_i. We may think of this number as the *posterior probability* that f_j^* generated the observation f_i.

The *assignment task* asks for determining an optimal assignment W^*, considered as "explanation" of the data based on the prior data $\mathcal{P}_{\mathcal{F}}$. We discuss next the ingredients of the objective function that will be used to solve assignment tasks.

3.1.2 Distance Matrix

Given $\mathcal{F}$, $d_{\mathcal{F}}$ and $\mathcal{P}_{\mathcal{F}}$, we compute the *distance matrix*

$$D \in \mathbb{R}^{m\times n}, \quad D_i \in \mathbb{R}^n, \quad D_{ij} = \frac{1}{\rho} d_{\mathcal{F}}(f_i, f_j^*), \tag{3.6a}$$

$$\rho > 0, \quad i \in [m], \quad j \in [n], \tag{3.6b}$$

where ρ is the first (from two) *user parameters* to be set. This parameter serves two purposes. It accounts for the unknown scale of the data f that depends on the application and hence cannot be known beforehand. Furthermore, its value determines what subset of the prior features f_j^*, $j \in [n]$ effectively affects the process of determining the assignment matrix W. This becomes explicit through the definition of the next processing stage, given by Eq. (3.12) below, that uses D as input. We call ρ *selectivity parameter*.

Furthermore, we set the initial value

$$W = W(0), \qquad W_i(0) := \frac{1}{n}\mathbb{1}_n, \quad i \in [m]. \tag{3.7}$$

of the flow (3.21) determining $W(t)$ that is introduced and discussed below in Sect. 3.2.3.

Note that W is initialized with the uninformative *uniform assignment* that is not biased toward a solution in any way.

3.1.3 Likelihood Matrix

The next processing step is based on the following

Definition 4 (*Lifting Map* (**Manifolds** $\mathcal{S}, \mathcal{W}$)) The lifting mapping is defined by

$$\exp\colon T\mathcal{S} \to \mathcal{S}, \qquad (p,u) \mapsto \exp_p(u) = \frac{pe^u}{\langle p, e^u\rangle}, \tag{3.8a}$$

$$\exp\colon T\mathcal{W} \to \mathcal{W}, \qquad (W,U) \mapsto \exp_W(U) = \begin{pmatrix} \exp_{W_1}(U_1) \\ \dots \\ \exp_{W_m}(U_m) \end{pmatrix}, \tag{3.8b}$$

where $U_i, W_i, i \in [m]$ index the row vectors of the matrices U, W, and where the argument decides which of the two mappings exp applies.

Remark 3 After replacing the arbitrary point $p \in \mathcal{S}$ by the barycenter $\frac{1}{n}\mathbb{1}_n$, readers will recognize the *softmax function* in (3.8a), i.e., $\langle \frac{1}{n}\mathbb{1}_n, e^u\rangle^{-1}\big(\frac{1}{n}\mathbb{1}_n e^u\big) = \frac{e^u}{\langle \mathbb{1}, e^u\rangle}$. This function is widely used in various application fields of applied statistics (e.g., [45]), ranging from parametrizations of distributions, e.g., for logistic classification [6], to other problems of modeling [34] not related to our approach.

The lifting mapping generalizes the softmax function through the dependency on the base point p. In addition, it approximates geodesics and accordingly the exponential mapping Exp, as stated next. We therefore use the symbol exp as mnemonic. Unlike Exp_p, the mapping $\exp_p$ is defined on the entire tangent space, cf. Remark 1.

Proposition 3 *Let*

$$v = \big(\mathrm{Diag}(p) - pp^\top\big)u, \qquad v \in T_p\mathcal{S}. \tag{3.9}$$

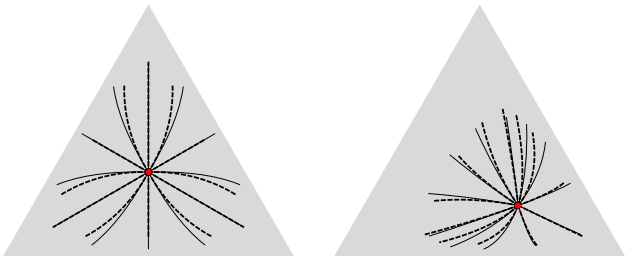


Fig. 5 Illustration of Proposition 3. Various geodesics $\gamma_{v^i}(t)$, $i \in [k]$, $t \in [t, t_{\max}]$ (*solid lines*) emanating from p (*red point*) with the same speed $\|v^i\|_p = \|v^j\|_p$, $\forall i, j$, are displayed together with the curves $\exp_p(u^i t)$, $i \in [k]$, $t \in [t, t_{\max}]$, where the vectors u^i, v^i, $i \in [k]$ satisfy (3.9)

Then $\exp_p(ut)$ *given by* (3.8a) *solves*

$$\dot p(t) = p(t)u - \langle p(t), u\rangle p(t), \qquad p(0) = p, \tag{3.10}$$

and provides a first-order approximation of the geodesic $\gamma_v(t)$ *from* (2.7a)

$$\exp_p(ut) \approx p + vt, \qquad \|\gamma_v(t) - \exp_p(ut)\| = \mathcal{O}(t^2). \tag{3.11}$$

Proof See "Proofs of Section 3 and Further Details" of Appendix 2. □

Figure 5 illustrates the approximation of geodesics γ_v and the exponential mapping Exp_p, respectively, by the lifting mapping $\exp_p$.

Remark 4 Note that adding any constant vector $c\mathbb{1}$, $c \in \mathbb{R}$ to a vector u does not change $\exp_p(u)$: $\frac{pe^{u+c\mathbb{1}}}{\langle p, e^{u+c\mathbb{1}}\rangle} = \frac{p(e^c\mathbb{1})e^u}{\langle p, (e^c\mathbb{1})e^u\rangle} = \frac{pe^u}{\langle p, e^u\rangle} = \exp_p(u)$. Accordingly, the same vector v is generated by (3.9). While the definition (3.8a) removes this ambiguity, there is no need to remove the mean of the vector u in numerical computations.

Given D and W as described in Sect. 3.1.2, we lift the matrix D to the manifold $\mathcal{W}$ by

$$L = L(W) := \exp_W(-U) \in \mathcal{W}, \tag{3.12a}$$

$$U_i = D_i - \frac{1}{n}\langle \mathbb{1}, D_i\rangle \mathbb{1}, \quad i \in [m], \tag{3.12b}$$

with exp defined by (3.8b). We call L *likelihood matrix* because the row vectors are discrete probability distributions which separately represent the similarity of each observation f_i to the prior data $\mathcal{P}_{\mathcal{F}}$, as measured by the distance $d_{\mathcal{F}}$ in (3.6).

Note that the operation (3.12) depends on the assignment matrix $W \in \mathcal{W}$.

3.1.4 Similarity Matrix

Based on the likelihood matrix L, we define the *similarity matrix*

$$S = S(W) \in \mathcal{W}, \tag{3.13a}$$

$$S_i = \mathrm{mean}_{\mathcal{S}}\{L_j\}_{j \in \tilde{\mathcal{N}}_{\mathcal{E}}(i)}, \quad i \in [m], \tag{3.13b}$$

where each row is the Riemannian mean (2.10) (using uniform weights) of the likelihood vectors, indexed by the neighborhoods as specified by the underlying graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$,

$$\tilde{\mathcal{N}}_{\mathcal{E}}(i) = \{i\} \cup \mathcal{N}_{\mathcal{E}}(i), \quad \mathcal{N}_{\mathcal{E}}(i) = \{j \in \mathcal{V} : ij \in \mathcal{E}\}. \tag{3.14}$$

Thus, S represents the similarity of the data within a local spatial neighborhood to the prior data $\mathcal{P}_{\mathcal{F}}$.

Note that S depends on W because L does so by (3.12). The *size* of the neighborhoods $|\tilde{\mathcal{N}}_{\mathcal{E}}(i)|$ is the *second-user parameter*, besides the selectivity parameter ρ for scaling the distance matrix (3.6). Typically, each $\tilde{\mathcal{N}}_{\mathcal{E}}(i)$ indexes the same local "window" around pixel location i. We then call the window size $|\tilde{\mathcal{N}}_{\mathcal{E}}(i)|$ *scale parameter*.

Remark 5 In basic applications, the distance matrix D will not change once the features and the feature distance $d_{\mathcal{F}}$ are determined. On the other hand, the likelihood matrix $L(W)$ and the similarity matrix $S(W)$ have to be recomputed as the assignment W evolves, as part of any numerical algorithm used to compute an optimal assignment W^*.

We point out, however, that more general scenarios are conceivable —without essentially changing the overall approach—where $D = D(W)$ depends on the assignment as well and hence has to be updated too, as part of the optimization process. Section 4.5 provides an example.

3.2 Objective Function, Optimal Assignment

We specify next the objective function as criterion for assignments and the gradient flow on the assignment manifold, to compute an optimal assignment W^*. Finally, based on W^*, the so-called assignment mapping is defined.

3.2.1 Objective Function

Getting back to the interpretation from Sect. 3.1.1 of the assignment matrix $W \in \mathcal{W}$ as *posterior probabilities*,

$$W_{ij} = \Pr(f_j^* | f_i), \tag{3.15}$$

of assigning prior feature f_j^* to the observed feature f_i, a natural *objective function* to be maximized is

$$\max_{W \in \mathcal{W}} J(W), \qquad J(W) := \langle S(W), W \rangle. \tag{3.16}$$

The functional J together with the feasible set $\mathcal{W}$ formalizes the following objectives:

1. Assignments W should *maximally correlate* with the feature-induced similarities $S = S(W)$, as measured by the inner product which defines the objective function $J(W)$.
2. Assignments of prior data to observations should be done in a *spatially coherent* way. This is accomplished by *geometric averaging* of likelihood vectors over local spatial neighborhoods, which turns the likelihood matrix $L(W)$ into the similarity matrix $S(W)$, *depending* on W.
3. Maximizers W^* should define *image labelings* in terms of rows $\overline{W}_i^* = e^{k_i} \in \{0,1\}^n$, $i, k_i \in [m]$, that are indicator vectors. While the latter matrices are not contained in the assignment manifold $\mathcal{W}$ as feasible set, we compute in practice assignments $W^* \approx \overline{W}^*$ arbitrarily close to such points. It will turn out below that the *geometry enforces* this approximation.
   As a consequence, in view of (3.15), such points W^* *maximize posterior probabilities*, akin to the interpretation of MAP inference with discrete graphical models by minimizing corresponding energy functionals. As discussed in Sect. 1, however, the mathematical structure of the optimization task of our approach and the way of fusing data and prior information are quite different.

The following statement formalizes the discussion of the form of desired maximizers W^*.

Lemma 4 *We have*

$$\sup_{W \in \mathcal{W}} J(W) = m, \tag{3.17}$$

and the supremum is attained at the extreme points

$$\overline{\mathcal{W}}^* := \big\{\overline{W}^* \in \{0,1\}^{m \times n} : \overline{W}_i^* = e^{k_i}, \tag{3.18a}$$

$$i \in [m],\ k_1, \ldots, k_m \in [n]\big\} \subset \overline{\mathcal{W}}, \tag{3.18b}$$

corresponding to matrices with unit vectors as row vectors.

Proof See "Proofs of Section 3 and Further Details" of Appendix 2. □

3.2.2 Assignment Mapping

Regarding the feature space $\mathcal{F}$, no assumptions were made so far, except for specifying a distance function $d_{\mathcal{F}}$. We have to be more specific about $\mathcal{F}$ only if we wish to *synthesize* the approximation to the given data f, in terms of an assignment W^* that optimizes (3.16) and the prior data $\mathcal{P}_{\mathcal{F}}$. We denote the corresponding approximation by

$$u : \mathcal{W} \to \mathcal{F}^{|\mathcal{V}|}, \qquad W \mapsto u(W), \qquad u^* := u(W^*), \tag{3.19}$$

and call it *assignment mapping*.

A trivial example of such a mapping concerns cases where prototypical feature vectors f^{*j}, $j \in [n]$ are assigned to data vectors f^i, $i \in [m]$: the mapping $u(W^*)$ then simply replaces each data vector by the convex combination of prior vectors assigned to it,

$$u^{*i} = \sum_{j \in [n]} W^*_{ij} f^{*j}, \qquad i \in [m]. \tag{3.20}$$

And if W^* approximates a global maximum $\overline{W}^*$ as characterized by Lemma 4, then each f_i is (almost) uniquely replaced by some $u^{*k_i} = f^{*k_i}$.

A less trivial example is the case of prior information in terms of patches. We specify the mapping u for this case and further concrete scenarios in Sect. 4.

3.2.3 Optimization Approach

The optimization task (3.16) does not admit a closed-form solution. We therefore compute the assignment by the *Riemannian gradient ascent flow* on the manifold $\mathscr{W}$,

$$\dot{W}_{ij} = \big(\nabla_{\mathscr{W}} J(W)\big)_{ij} \tag{3.21a}$$

$$= W_{ij}\Big(\big(\nabla_i J(W)\big)_j - \big\langle W_i, \nabla_i J(W)\big\rangle\Big), \tag{3.21b}$$

$$W_i(0) = \frac{1}{n}\mathbb{1}, \quad j \in [n], \tag{3.21c}$$

with

$$\nabla_i J(W) := \frac{\partial}{\partial W_i} J(W) \tag{3.21d}$$

$$= \Big(\frac{\partial}{\partial W_{i1}} J(W), \ldots, \frac{\partial}{\partial W_{in}} J(W)\Big)^\top, \ i \in [m], \tag{3.21e}$$

which results from applying (2.6) to the objective (3.16). The flows (3.21), for $i \in [m]$, are *not* independent as the product structure of $\mathscr{W}$ (cf. Sect. 2.3) might suggest. Rather, they are coupled through the gradient $\nabla J(W)$ which reflects the interaction of the distributions W_i, $i \in [m]$, due to the geometric averaging which results in the similarity matrix (3.13).

Observe that, by (3.21a) and $\langle \mathbb{1}, W_i\rangle = 1$,

$$\langle \mathbb{1}, \dot{W}_i\rangle = \langle \mathbb{1}, W_i \nabla_i J(W)\rangle \tag{3.22a}$$

$$- \langle W_i, \nabla_i J(W)\rangle \langle \mathbb{1}, W_i\rangle = 0, \quad i \in [m], \tag{3.22b}$$

that is $\nabla_{\mathscr{W}} J(W) \in T_W \mathscr{W}$, and thus the flow (3.21a) evolves on $\mathscr{W}$. Let $W(t) \in \mathscr{W}$, $t \geq 0$ solve (3.21a). Then, with the Riemannian metric (2.12),

$$\frac{\mathrm{d}}{\mathrm{d}t} J\big(W(t)\big) = \big\langle \nabla_{\mathscr{W}} J\big(W(t)\big), \dot{W}(t)\big\rangle_{W(t)} \tag{3.23a}$$

$$\overset{(3.21a)}{=} \big\|\nabla_{\mathscr{W}} J\big(W(t)\big)\big\|^2_{W(t)} \geq 0, \tag{3.23b}$$

that is, the objectivefunction value *increases* until a stationary point is reached where the Riemannian gradient vanishes. Clearly, we expect $W(t)$ to approximate a global maximum due to Lemma 4, which all satisfy the condition for stationary points $\overline{W}$,

$$0 = \dot{\overline{W}}_i = \overline{W}_i\big(\nabla_i J(\overline{W}) - \langle \overline{W}_i, \nabla_i J(\overline{W})\rangle \mathbb{1}\big), \quad i \in [m], \tag{3.24}$$

because replacing $\overline{W}_i$ in (3.24) by $\overline{W}^*_i = e^{k_i}$ for some $k_i \in [n]$ makes the bracket vanish for the k_i-th equation, whereas all other equations indexed by $j \neq k_i$, $j \in [n]$ are satisfied due to $\overline{W}^*_{ij} = 0$.

Regarding *interior* stationary points $\overline{W} \in \mathscr{W}$ with $\overline{W} \geq 0$ due to the definition of $\mathscr{W}$, all brackets $(\cdots)$ on the r.h.s. of (3.24) must vanish, which can only happen if the Euclidean gradient satisfies

$$\nabla_i J(\overline{W}) = \langle \overline{W}_i, \nabla_i J(\overline{W})\rangle \mathbb{1}, \qquad i \in [m] \tag{3.25}$$

including the case $\nabla J(\overline{W}) = 0$. Inspecting the gradient of the objective function (3.16), we get

$$\frac{\partial}{\partial W_{ij}} J(W) = \frac{\partial}{\partial W_{ij}} \langle S(W), W\rangle = \sum_{k,l} \frac{\partial}{\partial W_{ij}} \big(S_{kl}(W) W_{kl}\big) \tag{3.26a}$$

$$= \sum_{k,l} \Big(\frac{\partial}{\partial W_{ij}} S_{kl}(W)\Big) W_{kl} + S_{ij}(W) \tag{3.26b}$$

$$= \langle T^{ij}(W), W\rangle + S_{ij}(W), \tag{3.26c}$$

where both matrices $S(W)$ and $T^{ij}(W) = \frac{\partial}{\partial W_{ij}} S(W)$ depend in a smooth way on the data (3.1) and the prior set (3.3) through the distance matrix (3.6), the likelihood matrix (3.12) and the geometric averaging (3.13) which forms the similarity matrix $S(W)$. Regarding the second term on the r.h.s. of (3.26b), a computation relegated to "Proofs of Section 3 and Further Details of Appendix 2" yields

$$\langle T^{ij}(W), W\rangle = \sum_{k,l} -\Big(\big(H^k(W)\big)^{-1} h^{k,ij}(W)\Big)_l W_{kl}. \tag{3.27}$$

The way to compute the somewhat unwieldy explicit form of the r.h.s. is explained by (7.14f) and the corresponding appendix. In terms of these quantities, condition (3.25) for

stationary interior points translates to

$$\langle T^{ij}(\overline{W}), \overline{W}\rangle + S_{ij}(\overline{W}) \tag{3.28a}$$

$$= \sum_j \big(\langle T^{ij}(\overline{W}), \overline{W}\rangle + S_{ij}(\overline{W})\big)\overline{W}_{ij}, \tag{3.28b}$$

$$\forall i \in [m], \quad \forall j \in [n] \tag{3.28c}$$

including the special case $S_{ij}(W) = -\langle T^{ij}(W), W\rangle$, $\forall i \in [m]$, $j \in [n]$, corresponding to $\nabla J(\overline{W}) = 0$. Note that condition (3.28) requires that for every $i \in [m]$, the l.h.s. takes the *same* value for every $j \in [n]$, such that averaging with respect to W_i on the r.h.s. causes no change.

We do not have evidence for the nonexistence of specific data configurations, for which the flow (3.21) may reach such very specific stationary interior points. Any such point, however, will not be a maximum and be isolated, by virtue of the local strict convexity of the objective function (2.8) for Riemannian means (cf. Lemma 3 below), which determines the similarity matrix (3.13). Consequently, any perturbation (e.g., by numerical computation) will let the flow escape from such a point, in order to maximize the objective due to (3.23).

We summarize this reasoning by the

Conjecture 1 *For any data* (3.1) *and prior sets (3.3), up to a subset of* $\mathscr{W}$ *of measure zero, the flow* $W(t)$ *generated by (3.21) approximates a global maximum as defined by (3.18) in the sense that, for any* $0 < \varepsilon \ll 1$, *there is a* $t = t(\varepsilon)$ *such that*

$$\big\|W\big(t(\varepsilon)\big) - \overline{W}^*\big\| \leq \varepsilon, \qquad \textit{for some} \quad \overline{W}^* \in \overline{\mathscr{W}}^*. \tag{3.29}$$

Remark 6

1. Since $\overline{\mathscr{W}}^* \notin \mathscr{W}$, the flow $W(t)$ cannot converge to a global maximum, and numerical problems arise when (3.29) holds for ε very close to zero. Our strategy to avoid such problems is described in Sect. 3.3.1.
2. Although global maxima are not attained, we agree to call a point $W^* = W(t)$ *maximum* and *optimal assignment* that satisfies (3.29) for some fixed small ε. The criterion which terminates our algorithm is specified in Sect. 3.3.4.
3. Our numerical approximation of the flow (3.21) is detailed in Sect. 3.3.3.

3.3 Implementation

We discuss in this section specific aspects of the implementation of the variational approach.

3.3.1 Assignment Normalization

Because each vector W_i approaches some vertex $\overline{W}^* \in \overline{\mathscr{W}}^*$ by construction, and because the numerical computations are designed to evolve on $\mathscr{W}$, we avoid numerical issues by checking for each $i \in [m]$ every entry W_{ij}, $j \in [n]$, after each iteration of the algorithm (3.36) below. Whenever an entry drops below $\varepsilon = 10^{-10}$, we rectify W_i by

$$W_i \leftarrow \frac{1}{\langle \mathbb{1}, \tilde{W}_i\rangle}\tilde{W}_i, \tag{3.30a}$$

$$\tilde{W}_i = W_i - \min_{j\in[n]} W_{ij} + \varepsilon, \qquad \varepsilon = 10^{-10}. \tag{3.30b}$$

In other words, the number ε plays the role of 0 in our implementation. Our numerical experiments (Sect. 4) showed that this operation removed any numerical issues without affecting convergence in terms of the criterion specified in Sect. 3.3.4.

3.3.2 Computing Riemannian Means

Computation of the similarity matrix $S(W)$ due to Eq. (3.13) involves the computation of Riemannian means. In view of Definition 2, we compute the Riemannian mean $\mathrm{mean}_{\mathscr{S}}(\mathscr{P})$ of given points $\mathscr{P} = \{p^i\}_{i\in[N]} \subset \mathscr{S}$, using uniform weights, as fixed point $p^{(\infty)}$ by iterating the following steps.

$$\text{(1)} \quad \text{Set } p^{(0)} = \frac{1}{n}\mathbb{1}. \tag{3.31a}$$

Given $p^{(k)}$, $k \geq 0$, compute (cf. the explicit expressions (7.16b) and (2.7))

$$\text{(2)} \quad v^i = \mathrm{Exp}^{-1}_{p^{(k)}}(p^i), \quad i \in [N], \tag{3.31b}$$

$$\text{(3)} \quad v = \frac{1}{N}\sum_{i\in[N]} v^i, \tag{3.31c}$$

$$\text{(4)} \quad p^{(k+1)} = \mathrm{Exp}_{p^{(k)}}(v), \tag{3.31d}$$

and continue with step (2) until convergence. In view of the optimality condition (2.9), our implementation returns $p^{(k+1)}$ as a result if after carrying out step (3) the condition $\|v\|_\infty \leq 10^{-3}$ holds.

We point out that numerical problems arise at step (2) if *identical* vectors are averaged, as the expression (7.16b) shows. Such situations may occur, e.g., when computer-generated images are processed. Setting $\varepsilon = 1 - \langle\sqrt{p}, \sqrt{q}\rangle$ for two vectors $p, q \in \mathscr{S}$, we replace the expression (7.16b) by

$$\mathrm{Exp}_p^{-1}(q) \approx \frac{9\varepsilon^2 + 40\varepsilon + 480}{240\sqrt{1-\varepsilon/2}}\big(\sqrt{pq} - (1-\varepsilon)p\big) \quad \text{if } \varepsilon < 10^{-3}. \tag{3.32}$$

Although the iteration (3.31) converges quickly, carrying out such iterations as a subroutine, at each pixel and iterative step of the outer iteration (3.36), increases runtime

(of non-parallel implementations) noticeably. In view of the approximation of the exponential map $\mathrm{Exp}_p(v) = \gamma_v(1)$ by (3.11), it seems natural to approximate the Riemannian mean as well by modifying steps (2) and (4) above accordingly.

Lemma 5 *Replacing in the iteration (3.31) above the exponential mapping* Exp_p *by the lifting map* $\exp_p$ (3.8a) *yields the closed-form expression*

$$\begin{aligned} \mathrm{mean}_{\mathscr{S}}(\mathscr{P}) &\approx \frac{\mathrm{mean}_g(\mathscr{P})}{\langle \mathbb{1}, \mathrm{mean}_g(\mathscr{P})\rangle}, \\ \mathrm{mean}_g(\mathscr{P}) &= \Big(\prod_{i\in[N]} p^i\Big)^{\frac{1}{N}} \end{aligned} \tag{3.33}$$

as approximation of the Riemannian mean $\mathrm{mean}_{\mathscr{S}}(\mathscr{P})$*, with the geometric mean* $\mathrm{mean}_g(\mathscr{P})$ *applied componentwise to the vectors in* $\mathscr{P}$*.*

Proof See "Proofs of Section 3 and Further Details" of Appendix 2. □

Remark 7 Taking into account non-uniform weights $w \in \Delta_{N-1}$, according to Definition 2, is straightforward. We briefly take up this point in Sect. 5: see Eq. (5.2) and the corresponding paragraph together with figure 14.

3.3.3 Optimization Algorithm

A thorough analysis of various discrete schemes for numerically integrating the gradient flow (3.21), including stability estimates, is beyond the scope of this paper and will be separately addressed in follow-up work (see Sect. 5 for a short discussion).

Here, we merely adopted the following basic strategy from [33] that has been widely applied in the literature and performed remarkably well in our experiments. Approximating the flow (3.21) for each vector W_i, $i \in [m]$, by the time-discrete scheme

$$\frac{W_i^{(k+1)} - W_i^{(k)}}{t_i^{(k+1)} - t_i^{(k)}} = W_i^{(k)}\big(\nabla_i J(W^{(k)}) - \langle W_i^{(k)}, \nabla_i J(W^{(k)})\rangle \mathbb{1}\big), \tag{3.34a}$$

$$W_i^{(k)} := W_i\left(t_i^{(k)}\right), \tag{3.34b}$$

and choosing the adaptive step sizes $t_i^{(k+1)} - t_i^{(k)} = \frac{1}{\langle W_i^{(k)}, \nabla_i J(W^{(k)})\rangle}$, yields the multiplicative updates

$$W_i^{(k+1)} = \frac{W_i^{(k)}\big(\nabla_i J(W^{(k)})\big)}{\langle W_i^{(k)}, \nabla_i J(W^{(k)})\rangle}, \qquad i \in [m]. \tag{3.35}$$

We further simplify this update in view of the explicit expression (3.26) of the gradient $\nabla_i J(W)$ of the objective function that comprises two terms. The first one contributes the derivative of $S(W)$ with respect to W_i, which is significantly smaller than the second term $S_i(W)$ of (3.26), because $S_i(W)$ results from *averaging* (3.13) the likelihood vectors $L_j(W_j)$ over spatial neighborhoods and hence changes slowly. As a consequence, we simply drop this first term which, as a by-product, avoids the numerical evaluation of the expensive expressions (3.27) specifying the first term.

Thus, for computing the numerical results reported in this paper, we used the fixed-point iteration

$$W_i^{(k+1)} = \frac{W_i^{(k)}\big(S_i(W^{(k)})\big)}{\langle W_i^{(k)}, S_i(W^{(k)})\rangle}, \quad W_i^{(0)} = \frac{1}{n}\mathbb{1}, \quad i \in [m] \tag{3.36}$$

together with the approximation due to Lemma 5 for computing Riemannian means, which define by (3.13) the similarity matrices $S(W^{(k)})$. Note that this requires to recompute the likelihood matrices (3.12) as well, at each iteration k (see Fig. 1).

3.3.4 Termination Criterion

Algorithm (3.36) was terminated if the average entropy

$$-\frac{1}{m}\sum_{i\in[m]}\sum_{j\in[n]} W_{ij}^{(k)} \log W_{ij}^{(k)} \tag{3.37}$$

dropped below a threshold. For example, a threshold value 10^{-3} means in practice that, up to a tiny fraction of indices $i \subset [m]$ that should not matter for a subsequent further analysis, all vectors W_i are very close to unit vectors, thus indicating an almost unique assignment of prior items f_j^*, $j \in [n]$ to the data f_i, $i \in [m]$. Note that this termination criterion conforms to Conjecture 1 and was met in all experiments.

4 Illustrative Applications and Discussion

We focus in this section on few academical, yet non-trivial numerical examples, to illustrate and discuss basic properties of the approach. Elaborating any specific application is outside the scope of this paper.

4.1 Parameters, Empirical Convergence Rate

Figure 6 shows a color image and a noisy version of it. The latter image was used as input data of a labeling problem. Both images comprise 31 color vectors forming the prior data set $\mathscr{P}_{\mathscr{F}} = \{f^{1*}, \ldots, f^{31*}\}$. The labeling task is to assign

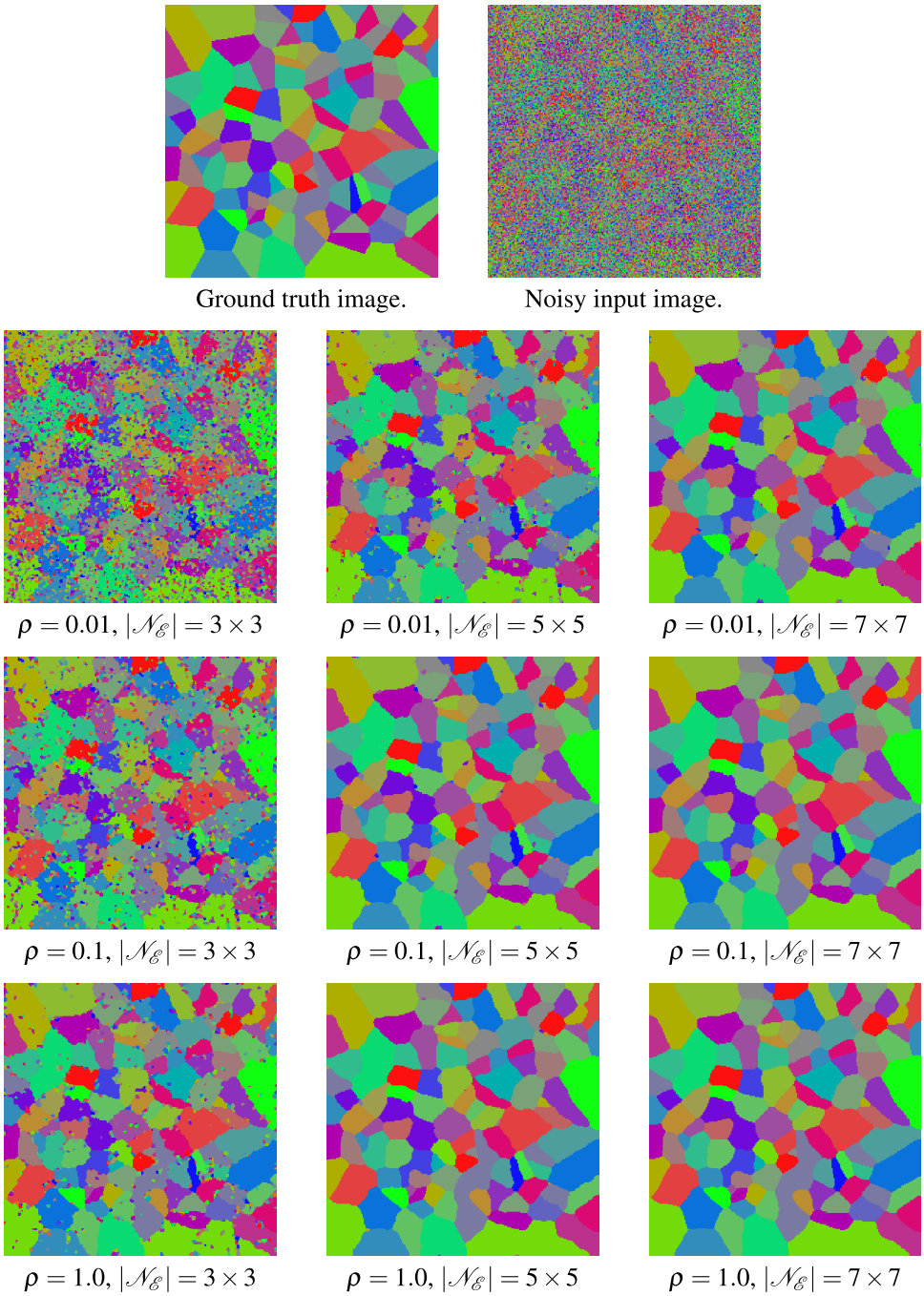


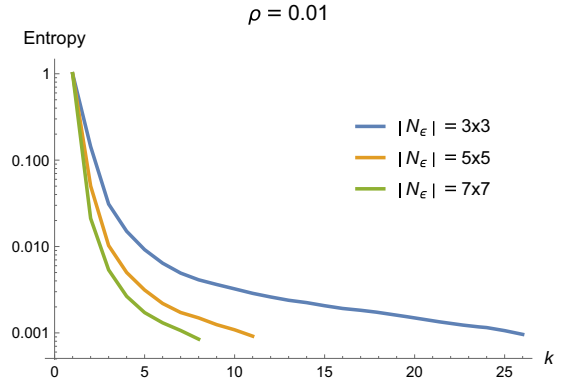
Fig. 6 Parameter influence on labeling. The *top row* shows a ground-truth image and noisy input data. Both images and the prior data set $\mathscr{P}_\mathscr{F}$ are composed of 31 color vectors. Each *color vector*is encoded as a vertex of the simplex Δ_{30}. This results in unit distances between all colors and thus enables an unbiased assessment of the impact of geometric averaging and the two parameter values ρ, $|\mathcal{N}_\varepsilon|$. The remaining panels show the assignments $u(W^*)$ for various parameter values where W^* maximizes the objective function (3.16). The spatial scale $|\mathcal{N}_\varepsilon|$ increases from *left to right*. The parameter ρ increases downwards. The results illustrate the compromise between sensitivity to noise and to the geometry of signal transitions. The selectivity parameter ρ increases from *top to bottom*. If ρ is chosen too small, then there is a tendency to noise-induced oversegmentation, in particular at small spatial scales $|\mathcal{N}_\varepsilon|$. Depending on the application, however, the ability to separate the physical and the spatial scale in order to recognize outliers with small spatial support, while performing diffusion at a larger spatial scale as in the panels of the *left column*, may be beneficial

these vectors in a spatially coherent way to the input data so as to recover the ground-truth image.

This tasks should not be confused with image denoising in the traditional sense [9] where noise has to be removed from *real-valued* image data. Rather, the experiment depicted by Fig. 6 represents difficult *classification* tasks where the assignment process is essential in order to cope with the high noise level.

Every color vector was encoded by the vertices of the simplex Δ_{30}, that is, by the unit vectors $\{e^1, \ldots, e^{31}\} \subset \{0, 1\}^{31}$. Choosing the distance $d_\mathscr{F}(f^i, f^j) := \|f^i - f^j\|_1$, this results in unit distances between all pairs of data points and hence enables to assess most clearly the impact of geometric spatial averaging and the influence of the two parameters ρ and $|\mathcal{N}_\varepsilon|$, introduced in Sects. 3.1.2 and 3.1.4, respectively. We refer to the caption for a brief discussion of the selectivity parameter ρ and the spatial scale in terms of $|\mathcal{N}_\varepsilon|$.

The reader familiar with total variation-based denoising, where a *single* parameter is only used to control the influence of regularization, may ask why *two* parameters are used in



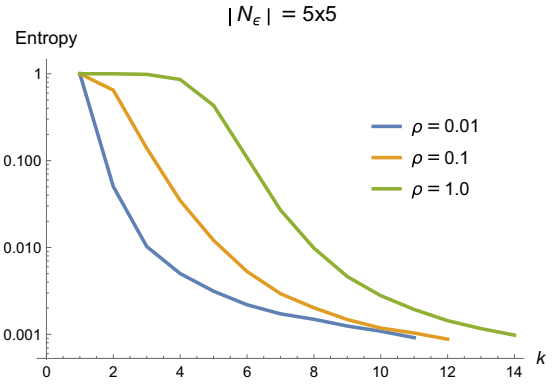


Fig. 7 Parameter values and convergence rate. Average entropy (3.37) of the assignment vectors $W_i^{(k)}$ as a function of the iteration counter k and the two parameters ρ and $|\mathcal{N}_\varepsilon|$, for the labeling task illustrated in Fig. 6. The *left panel* shows that despite high selectivity in terms of a small value of ρ, small spatial scales necessitate to resolve more conflicting assignments through propagating information by geometric spatial averaging. As a consequence, more iterations are needed to achieve convergence and a labeling. The *right panel*, on the other hand, shows that at a fixed spatial scale $|\mathcal{N}_\varepsilon|$, higher selectivity leads to faster convergence, because outliers are simply removed from the averaging process, whereas low selectivity leads to an assignment (labeling) taking all data into account

the present approach and if they are necessary. We refer again to Fig. 6 and the caption where the separation of the physical and spatial scale based on different parameter choices is demonstrated. The total variation measure couples these scales as the co-area formula explicitly shows. As a consequence, a single parameter is only needed. On the other hand, larger values of this parameter lead to the well-known loss-of-contrast effect, which using the present approach can be avoided by properly choosing the parameters ρ, $|\mathcal{N}_\varepsilon|$ corresponding to these two scales.

Figure 7 shows how convergence of the iterative algorithm (3.36) is affected by these two parameters. It also demonstrates that few tens of massively parallel outer iterations suffice to reach the termination criterion of Sect. 3.3.4. A parallel implementation only has to take into account the spatial neighborhood (3.14) where pixel locations directly interact in order to compute by geometric averaging the likelihood matrix (3.13).

All results were computed using the assignment mapping (3.20) *without* rounding. This shows that the termination criterion of Sect. 3.3.4, illustrated in Fig. 7, leads to (almost) unique assignments .

4.2 Vector-Valued Data

Let $f^i \in \mathbb{R}^d$ denote vector-valued image data or extracted feature vectors at locations $i \in [m]$, and let

$$\mathscr{P}_{\mathscr{F}} = \{f^{*1}, \ldots, f^{*n}\} \tag{4.1}$$

denote the prior information given by prototypical feature vectors. In the example that follows below, f^i will be a RGB color vector. It should be clear, however, that *any* feature vector of arbitrary dimension d could be used instead, depending on the application at hand. We used the distance function

$$d_{\mathscr{F}}(f^i, f^{*j}) = \frac{1}{d}\|f^i - f^{*j}\|_1, \tag{4.2}$$

with the normalizing factor $1/d$ to make the choice of the parameter ρ insensitive with respect to the dimension d of the feature space. Given an optimal assignment matrix W^* as solution to (3.16), the prior information assigned to the data is given by the assignment mapping

$$u^i = u^i(W^*) = \mathbb{E}_{W_i^*}[\mathscr{P}_{\mathscr{F}}], \qquad i \in [m], \tag{4.3}$$

which merely replaces each data vector f^i by the prior vector f^{*j} assigned to it through W_i^*.

Figure 8 shows the assignment of 20 prototypical color vectors to a color image for various values of the spatial scale parameter $|\mathcal{N}_\varepsilon|$, while keeping the selectivity parameter ρ fixed. As a consequence, the induced assignments and image partitions exhibit a natural coarsening effect in the spatial domain.

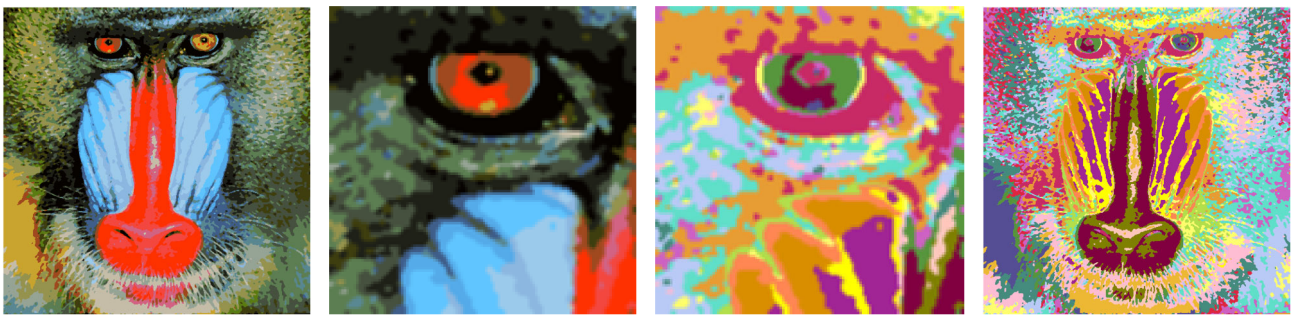
4.3 Patches

Let f^i denote a patch of raw image data (or, more generally, a patch of features vectors)

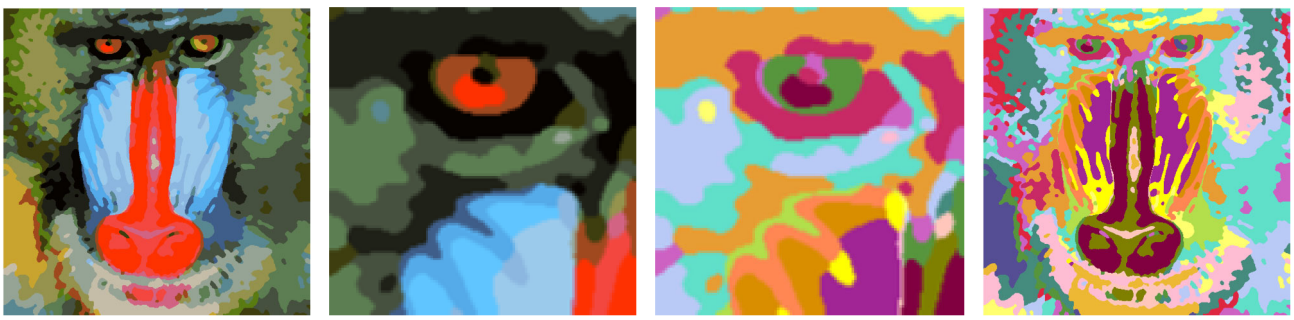
$$f^{ij} \in \mathbb{R}^d, \qquad j \in \mathcal{N}_p(i), \qquad i \in [m], \tag{4.4}$$

centered at location $i \in [m]$ and indexed by $\mathcal{N}_p(i) \subset \mathcal{V}$ (subscript p indicates neighborhoods for patches). With each entry $j \in \mathcal{N}_p(i)$, we associate the Gaussian weight

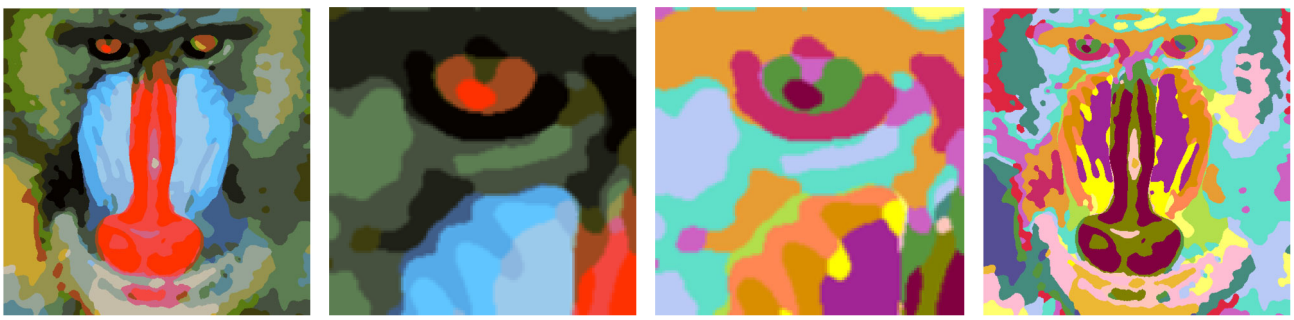
(a)



(b)



(c)



(d)

Fig. 8 Image labeling at different spatial scales. The *two rightmost columns* show the same information using a random color code for the assignment of the 20 prior vectors to pixel locations, to highlight the induced image partitions. Increasing the spatial scale $|\mathcal{N}_\varepsilon|$ for a *fixed* value of the selectivity parameter ρ induces a natural coarsening of the assignments and the corresponding image partitions along the spatial scale. **a** Input image (*left panel*) and a section of it. Twenty color vectors (*right panel*) forming the set prior data set $\mathscr{P}_{\mathscr{F}}$ according to Eq. (4.1). **b** Assignment $u(W^*)$, $|\mathcal{N}_\varepsilon| = 3 \times 3$, $\rho = 0.01$. **c** Assignment $u(W^*)$, $|\mathcal{N}_\varepsilon| = 7 \times 7$, $\rho = 0.01$. **d** Assignment $u(W^*)$, $|\mathcal{N}_\varepsilon| = 11 \times 11$, $\rho = 0.01$

$$w_{ij}^p := G_\sigma(\|x^i - x^j\|), \qquad i, j \in \mathcal{N}_p(i), \tag{4.5}$$

where the vectors $x^i, x^j \in \mathbb{R}^d$ correspond to the locations in the image domain indexed by $i, j \in \mathcal{V}$. Specifically, w^p is chosen to be the discrete impulse response of a Gaussian low-pass filter supported on $\mathcal{N}_p(i)$, so that the scale σ directly depends on the patch size and does not need to be chosen by hand. Such downweighting of values that are less close to the center location of a patch is an established elementary technique for reducing boundary and ringing effects of patch ("window")-based image processing.

The prior information is given in terms of n prototypical patches

$$\mathscr{P}_{\mathscr{F}} = \{f^{*1}, \ldots, f^{*n}\}, \tag{4.6}$$

and a corresponding distance

$$d_{\mathscr{F}}(f^i, f^{*j}), \qquad i \in [m], \quad j \in [n]. \tag{4.7}$$

There are many ways to choose this distance depending on the application at hand. We refer to the Examples 1 and 2 below. Expression (4.7) is based on the tacit assumption that patch f^{*j} is centered at i and indexed by $\mathcal{N}_p(i)$ as well.

Given an optimal assignment matrix W^*, it remains to specify how prior information is assigned to every location $i \in \mathcal{V}$, resulting in a vector $u^i = u^i(W^*)$ that is the overall result of processing the input image f. Location i is affected by patches that overlap with i. Let us denote the indices of these patches by

$$\mathcal{N}_p^{i\leftarrow j} := \{j \in \mathcal{V} : i \in \mathcal{N}_p(j)\}. \tag{4.8}$$

Every such patch is centered at location j to which prior patches are assigned by

$$\mathbb{E}_{W_j^*}[\mathscr{P}_{\mathscr{F}}] = \sum_{k\in[n]} W_{jk}^* f^{*k}. \tag{4.9}$$

Let location i be indexed by i_j in patch j (local coordinate inside patch j). Then, by summing over all patches indexed by $\mathcal{N}_p^{i\leftarrow j}$ whose supports include location i, and by weighting the contributions to location i by the corresponding weights (4.5), we obtain the vector

$$u^i = u^i(W^*) = \frac{1}{\sum_{j'\in\mathcal{N}_p^{i\leftarrow j}} w_{j'i_j}^p} \sum_{j\in\mathcal{N}_p^{i\leftarrow j}} w_{ji_j}^p \sum_{k\in[n]} W_{jk}^* f^{*ki_j} \in \mathbb{R}^d, \tag{4.10}$$

that is assigned by W^* to location i. This expression looks more clumsy than it actually is. In words, the vector u^i assigned to location i is the convex combination of vectors contributed from patches overlapping with i that itself are formed as convex combinations of prior patches. In particular, if we consider the common case of *equal* patch supports $\mathcal{N}_p(i)$ for every i that additionally are *symmetric* with respect to the center location i, then $\mathcal{N}_p^{i\leftarrow j} = \mathcal{N}_p(i)$. As a consequence, due to the symmetry of the weights (4.5), the first sum of (4.10) sums up all weights w_{ij}^p. Hence, the normalization factor on the right-hand side of (4.10) equals 1, because the low-pass filter w^p preserves the zero-order moment (mean) of signals. Furthermore, it then makes sense to denote by $(-i)$ the location i_p corresponding to i in patch j. Thus (4.10) becomes

$$u^i = u^i(W^*) = \sum_{j\in\mathcal{N}_p(i)} w_{j(-i)}^p \sum_{k\in[n]} W_{jk}^* f^{*k(-i)}. \tag{4.11}$$

Introducing in view of (4.9) the shorthand

$$\mathbb{E}_{W_j^*}^i[\mathscr{P}_{\mathscr{F}}] := \sum_{k\in[n]} W_{jk}^* f^{*k(-i)} \tag{4.12}$$

for the vector assigned to i by the convex combination of prior patches assigned to j, we finally rewrite (4.10) due the symmetry $w_{j(-i)}^p = w_{ji}^p = w_{ij}^p$ in the more handy form[1]

$$u^i = u^i(W^*) = \mathbb{E}_{w^p}\big[\mathbb{E}_{W_j^*}^i[\mathscr{P}_{\mathscr{F}}]\big]. \tag{4.13}$$

The inner expression represents the assignment of prior vectors to location i by fitting prior patches to all locations $j \in \mathcal{N}(i)$. The outer expression fuses the assigned vectors. If they were all the same, the outer operation would have no effect, of course.

We discuss further properties of this approach by concrete examples.

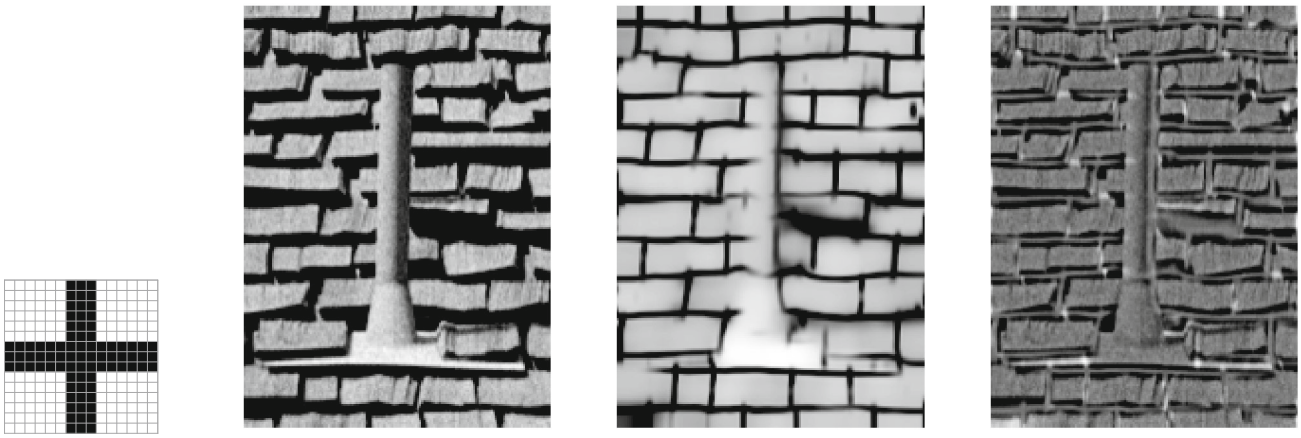
Example 1 (Patch Assignment) Figure 9 shows an image f and the corresponding assignment $u(W^*)$ based on a patch dictionary $\mathscr{P}_{\mathscr{F}}$ that was formed as explained in the caption.

We chose the distance $d_{\mathscr{F}}$ of Eq. (4.2),

$$d_{\mathscr{F}}(f^i, f^{*j}) = \frac{1}{|\mathcal{N}_p(i)|}\|f^i - f^{*j(i)}\|_1, \tag{4.14}$$

where here the arguments f^i, f^{*j} stand for the vectorized scalar-valued patches centered at location i, after adapting each prior template f^{*j} at each pixel location i to the data f, denoted by $f^{*j} = f^{*j(i)}$ in (4.14). Each such template takes two values that were adapted to the template f^i to which it is compared, i.e.,

[1] For locations i close to the boundary of the image domain where patch supports $\mathcal{N}_p(i)$ shrink, the definition of the vector w^p has to be adapted accordingly.



(a) **(b)** **(c)** **(d)**

Fig. 9 **a** A patch supposed to represent prior knowledge about the structure of an image f (**b**). The dictionary $\mathscr{P}_{\mathscr{F}}$ of Eq. (4.6) was generated by all translations of (**a**) and assigned to the image (**b**), using a distance $d_{\mathscr{F}}$ that adapts the two *grayvalues* of each template to the data—see Eqs. (4.14) and (4.15). The resulting assignment $u(W^*)$ is depicted by (**c**). **d** The residual image $v(W^*) := f - u(W^*)$ by subtracting (**c**) from (**b**) (rescaled for better visibility)

$$f_k^{*j(i)} \in \left\{ f^i_{\text{low}}, f^i_{\text{high}} \right\}, \quad \forall k, \tag{4.15a}$$

where

$$f^i_{\text{low}} = \text{median}\left\{ f^i_j : \, j \in \mathscr{N}_p(i), \;\; f^i_j < \text{median}\{f^i_j\}_{j\in\mathscr{N}_p(i)} \right\}, \tag{4.15b}$$

$$f^i_{\text{high}} = \text{median}\left\{ f^i_j : \, j \in \mathscr{N}_p(i), \;\; f^i_j \geq \text{median}\{f^i_j\}_{j\in\mathscr{N}_p(i)} \right\}. \tag{4.15c}$$

The result in Fig. 9c illustrates how the approximation $u(W^*)$ of f is restricted by the prior knowledge, leading to normalized signal transitions regarding both the spatial geometry and the signal values. By maximizing the objective (3.16), a patch-consistent and dense cover of the image is computed. It induces a strong nonlinear image filtering effect by fusing through assignment for each single pixel value more than 200 predictions of possible values based on the patch dictionary $\mathscr{P}_{\mathscr{F}}$.

The approach enables to model additive image decompositions

$$f = u(W^*) + v(W^*), \tag{4.16}$$

that is, image = geometry + texture & noise, for specific image classes, which are implicitly represented by the dictionary $\mathscr{P}_{\mathscr{F}}$. Such decomposition appears to be more discriminative than additive image decompositions achieved by convex variational approaches (see, e.g., [2]) that employ various regularizing norms, for this purpose.

Example 2 (Patch Assignment) Figure 10 shows a fingerprint image characterized by two gray values $f^*_{\text{dark}}, f^*_{\text{bright}}$ that were extracted from the histogram of f after removing a smooth function of the spatially varying mean value (panel (b)). The latter was computed by interpolating the median values for each patch of a coarse 16×16 partition of the entire image.

Figure 10c shows the dictionary of patches modeling the remaining binary signal transitions. An essential difference to Example 1 is the *subdivision of the dictionary into classes of equivalent patches* corresponding to each orientation. The averaging process was set up to distinguish only the assignment of patches of *different* patch classes and to treat patches of the same class equally. This makes geometric averaging particularly effective if signal structures conform to a single class on larger spatial connected supports. Moreover, it reduces the problem size to merely 13 class labels: 12 orientations at $k \cdot 30°$, $k \in [12]$ degrees, together with the single constant patch complementing the dictionary.

The distance $d_{\mathscr{F}}(f^i, f^{*j})$ between the image patch centered at i and the j-th prior patch was chosen depending on both the prior patch and the data patch it was compared to: For the constant prior patch, the distance was

$$d_{\mathscr{F}}(f^i, f^{*j}) = \frac{1}{|\mathscr{N}_p(i)|} \| f^i - f^*_i f^{*j} \|_1 \tag{4.17a}$$

with

$$f^*_i = \begin{cases} f^*_{\text{dark}}, & \text{if } \text{med}\{f^i_j\}_{j\in\mathscr{N}_p(i)} \leq \frac{1}{2}\left(f^*_{\text{dark}} + f^*_{\text{bright}} \right), \\ f^*_{\text{bright}}, & \text{otherwise.} \end{cases} \tag{4.17b}$$

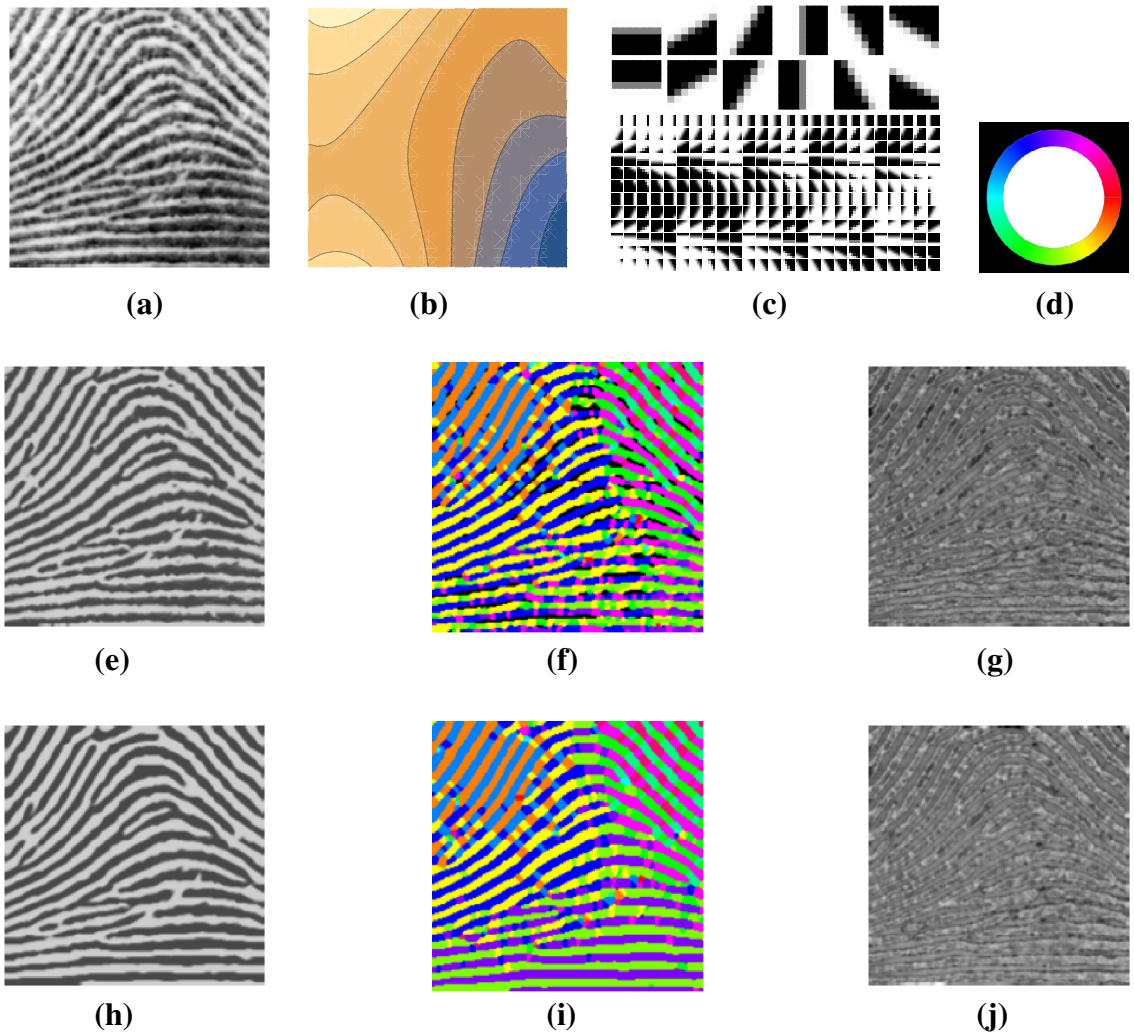


Fig. 10 Analysis of the local signal structure of image **a** by patch assignment. This process is twofold non-local: **i** through the assignment of 3×3 patches (*center row*) and 7×7 patches, respectively, and **ii** due to the gradient flow (3.21) that promotes the spatially coherent assignment of patches corresponding to different orientations of signal transitions, in order to maximize the similarity objective (3.16). **a** Input image f. **b** Contourplot of a smooth image computed and subtracted from f as a preprocessing step. **c** Prior patches representing binary signal transitions at orientations $0°, 30°, \ldots$ (*top row*), and the corresponding translation invariant dictionary (*bottom row*). Each row of patches constitutes an *equivalence class* of patches. **d** Color code indicating oriented bright-to-dark signal transitions. **e** Assignment $u(W^*)$ of 3×3 patches to image f from (**a**) ($\rho = 0.02$). **f** Class label of assigned patches encoded due to (**d**). *Black* means assignment of the constant template that was added to the dictionary (**c**). **g** Residual image $v(W^*) = f - u(W^*)$ (rescaled for visualization). **h** Assignment $u(W^*)$ of 7×7 patches to image f from (**a**) ($\rho = 0.02$). **i** Class label of assigned patches encoded due to (**d**). **j** Residual image $v(W^*) = f - u(W^*)$ (rescaled for visualization)

For all other prior patches, the distance was

$$d_{\mathscr{F}}\left(f^i, f^{*j}\right) = \frac{1}{|\mathscr{N}_p(i)|}\|f^i - f^{*j}\|_1. \tag{4.18}$$

The center and bottom rows in Fig. 10, respectively, show the assignment $u(W^*)$ of the dictionary of 3×3 patches (center row) and of 7×7 patches (bottom row). The center panels (f) and (i) depict the class labels of these assignments according to the color code of panel (d). These images display the interpretation of the image structure of f from panel (a). While the assignment of patches of size 3×3 is slightly noisy, which becomes visible through the assignment of the constant template marked by black in panel (f), the assignment of 5×5 or 7×7 patches results in a robust and spatially coherent, accurate representation of the local image structure. The corresponding pronounced nonlinear filtering effect is due to the consistent assignment of a large number of patches at each pixel location and fusing the corresponding predicted values.

Panels (g) and (j) show the resulting additive image decompositions (4.16) that seem difficult to achieve when using established convex variational approaches (see, e.g.,

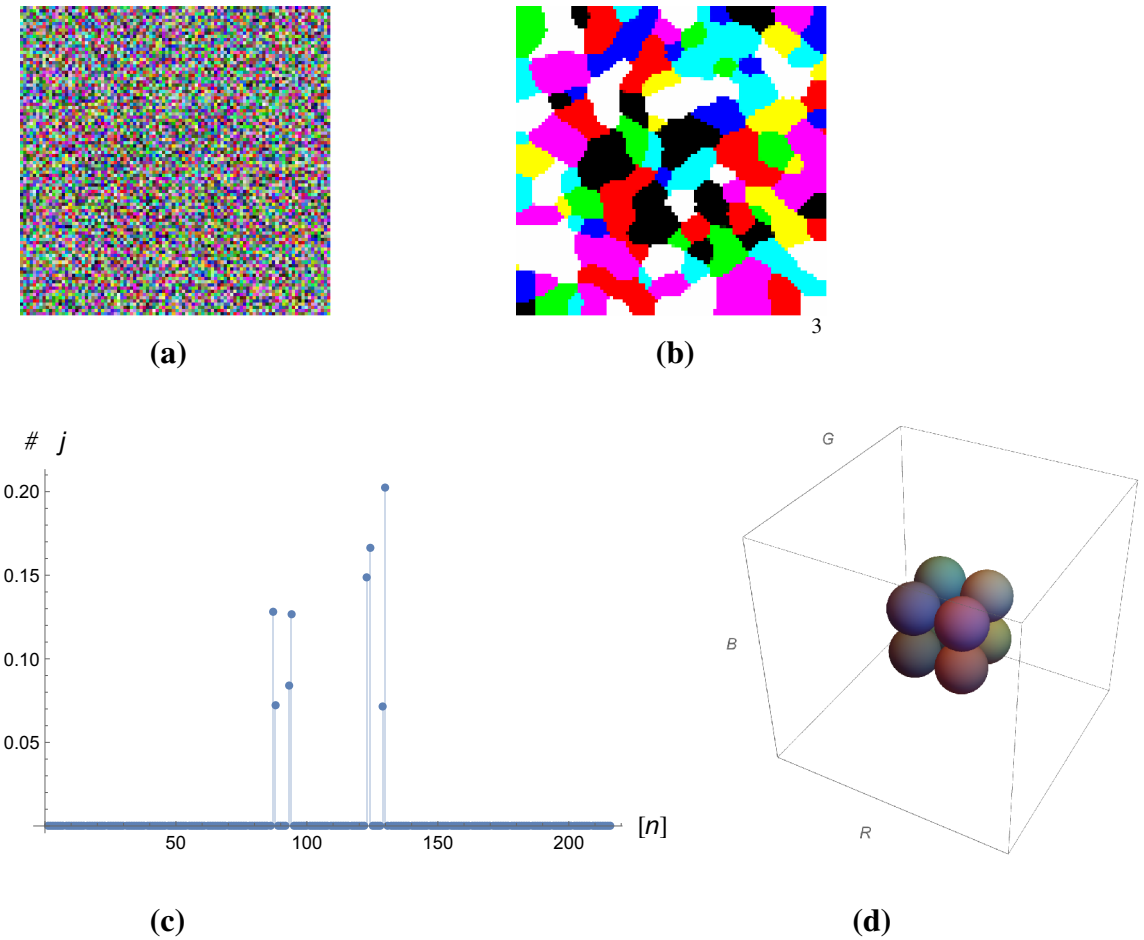


Fig. 11 Unsupervised assignment of uniform noise **a** to itself in terms of a uniform discretization of the *rgb color* cube $[0,1]^3$ that does not include the color *gray* $0.5(1,1,1)^\top$. The assignment selects the 8 colors (**d**) closest to *gray* with random frequencies (**c**) and a spatially random partition (**b**) (rescaled to highlight the partition). **a** Uniform noise, **b** sparse assignment $u(W^*)$ (displayed after rescaling) of 6^3 color vectors corresponding to a uniform discretization of the *rgb-cube* $[0,1]^3$ to the image (**a**) yields a noise-induced random piecewise constant partition through geometric averaging (parameters: $|\mathcal{N}_\varepsilon| = 7 \times 7$, $\rho = 0.01$), **c** Relative frequencies of assignment of the prior color vectors f^{*j}, $j \in [6^3]$. The 8 nonzero frequencies correspond to vectors indicated in the *color cube* (**d**), **d** 8 color vectors (out of 6^3) closest to *gray* (with equal distance) only were assigned to (**a**), resulting in (**b**). These colors look differently in (**b**) due to rescaling the image $u(W^*)$ to $[0,1]^3$ for better visibility

[2]) that employ various regularizing norms and duality, for this purpose.

Finally, we point out that it would be straightforward to add to the dictionary further patches modeling minutiae and other features relevant to fingerprint analysis. We do not consider in this paper any application-specific aspects, however.

4.4 Unsupervised Assignment

We consider the case that no prior information is available.

The simplest way to handle the absence of prior information is to use the given data themselves as prior information along with a suitable constraint, to enforce selection of the most important parts by *self-assignment*.

In order to illustrate this mechanism clearly, Fig. 11 shows as example the assignment of uniform noise to itself. As prior data $\mathscr{P}_\mathscr{F}$, we uniformly discretized the rgb color cube $[0,1]^3$ at $0, 0.2, 0.4, \ldots, 1$ along each axis, resulting in $|\mathscr{P}_\mathscr{F}| = 6^3 = 216$ color vectors. Because there is no preference for any of these vectors, spatial diffusion of uniform noise at any spatial scale will inherently end up with the average color gray, which however is excluded from the prior set, by construction. Accordingly, the process terminated with a spatially random assignment of the 8 color vectors closest to gray (Figs. 11b rescaled and 11d) solely induced by the input noise and geometric averaging at a certain scale. Figure 11c depicts the relative frequencies each prior vector is assigned to some location. Except for the 8 aforementioned vectors, all others are ignored.

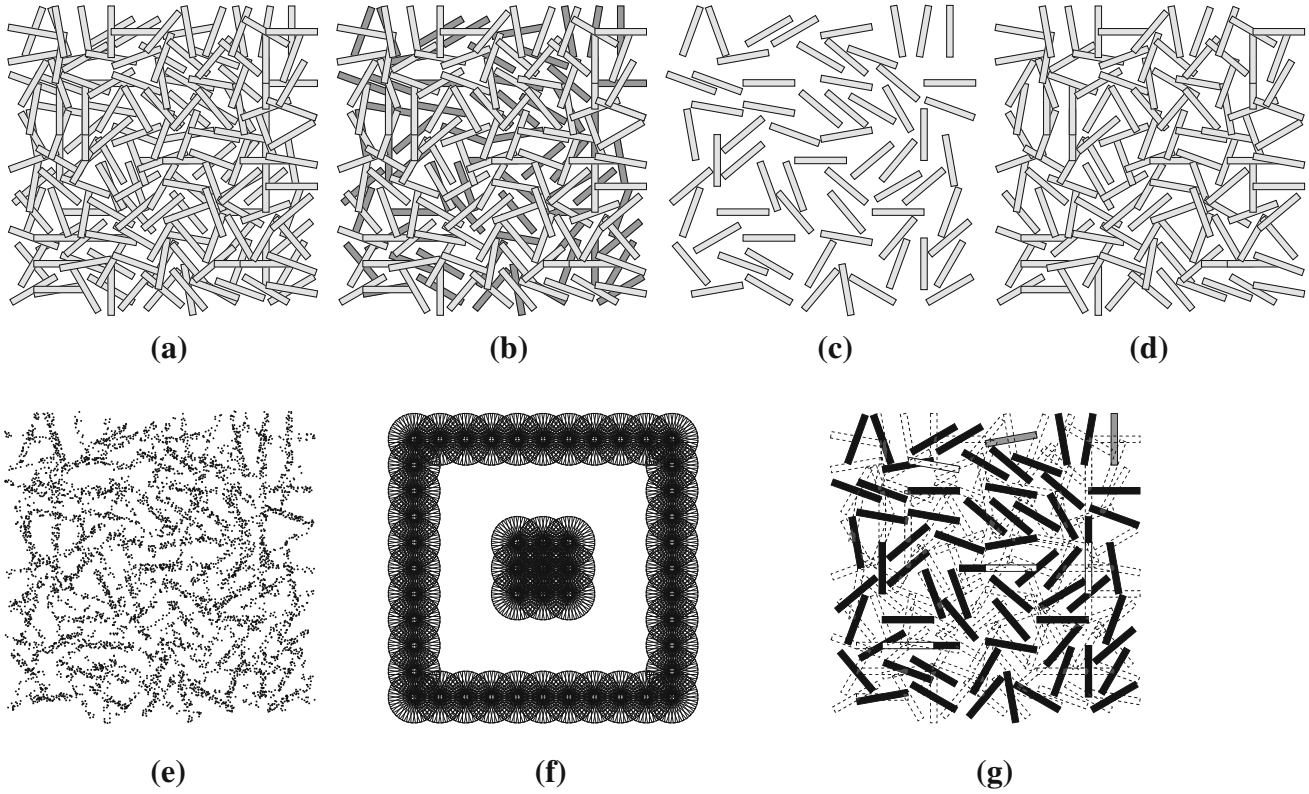


Fig. 12 Scenario for evaluating the approach of Sect. 4.5. **f** Illustrates the set of all rectangles and corresponding subsets (**c**, **d**). Unlike (**d**), the rectangles (**c**) do not intersect. Sampling the rectangles from both (**c**, **d**), shown together by (**a**, **b**), produced the input data (**e**). The task is to recognize among (**f**) all foreground objects (**c**) based on unary features (coverage of points) and disjunctive constraints (rectangles should not intersect). **g** Discusses the result. **a** Collection of rectangular areas that result in (**e**) after uniform point sampling. **b** Decomposition of the rectangles (**a**) into foreground (*dark*, cf. **c**), and background (*light*, cf. **d**). **c** Randomly oriented foreground rectangles that do not intersect. **d** Arbitrary sample of background rectangles from (**f**). **e** Input data: point pattern resulting from uniformly sampling the rectangles (**a**). **f** All possible rectangles densely cover the domain as indicated in the center region (not completely shown for better visibility). **g** Assignment (labeling) of the rectangles (**f**) based on the data (**e**): recognized foreground objects from (**c**) (*black*) and recognized background objects from (**d**) (*dashed*). Two foreground objects were erroneously labeled as background (*gray*). All remaining rectangles from (**f**) also belong to the background, four of which were erroneously labeled as foreground (*white*)

A detailed elaboration of unsupervised scenarios based on our approach, for both vector- and patch-valued data, will be studied in our follow-up work (Sect. 5).

4.5 Labeling with Adaptive Distances

In this section, we consider a simple instance of the more general class of scenarios where the distance matrix (3.6) $D = D(W)$ depends on the assignment matrix W, in addition to the likelihood matrix $L(W)$ and the similarity matrix $S(W)$.

Figure 12e displays a point pattern that was generated by sampling a foreground and background process of randomly oriented rectangles, as explained by the remaining panels in Fig. 12. The task is to recover the foreground process among all possible rectangles (Fig. 12f) based on (1) unary features given by the fraction of points covered by each rectangle, and on (2) the prior knowledge that unlike background rectangles, elements of the foreground process do *not* intersect. Rectangles of the background process were slightly less densely sampled than foreground rectangles so as to make the unary features indicative. Due to the overlap of many rectangles (Fig. 12a), however, these unary features are noisy ("weak").

As a consequence, exploiting the prior knowledge that foreground rectangles do not intersect becomes decisive. This is done by determining the intersection pattern of all rectangles (Fig. 12f) in terms of Boolean values that are arranged into matrices R_{ij}, for each edge ij of the grid graph whose vertices correspond to the centroids of the rectangles in Fig. 12f: $(R_{ij})_{k,l} = 1$ if rectangle k at position i intersects with rectangle l at position j, and $(R_{ij})_{k,l} = 0$ otherwise. Due to the geometry of the rectangles, a rectangle at position i may only intersect with $8 \times 18 = 144$ rectangles located within a 8-neighborhood $j \in \mathcal{N}_\varepsilon(i)$. Generalizations to other geometries are straightforward.

The inference task to recover the foreground rectangles (Fig. 12c) from the point pattern (Fig. 12e) may be seen as a multi-labeling problem based on an asymmetric Potts-like model: Labels correspond to equally oriented rectangles and have to be determined so as to maximize the coverage of points, subject to the pairwise constraints that selected rectangles do not intersect. Alternatively, we may think of binary "off–on" variables that are assigned to each rectangle in Fig. 12f, which have to be determined subject to *disjunctive* constraints: At each location, at most a single variable may become active, and pairwise active variables have to satisfy the intersection constraints. Note that in order to suppress intersecting rectangles, penalizing costs are only encountered if (a subset of) pairs of variables receive the *same* value 1 (=active and intersecting). This violates the submodularity constraint [29, Eq. (7)] and hence rules out global optimization using graph cuts.

Taking all ingredients into account, we define the distance vector field

$$D_i = D_i(W) = \frac{1}{\rho}\begin{pmatrix}\tilde{D}_i(W)\\ \sigma\end{pmatrix}, \tag{4.19a}$$

$$\tilde{D}_i(W) = -p^i + \frac{\lambda}{|\mathcal{N}_\varepsilon(i)|}\sum_{j\in\mathcal{N}_\varepsilon(i)} R_{ij}W_j, \quad \lambda,\sigma > 0, \tag{4.19b}$$

where $\rho > 0$ is the selectivity parameter from (3.6), $\sigma > 0$ represents the cost of the additional label: "none rectangle," vector p^i collects the fractions of points covered by the rectangles at position i, and $\lambda > 0$ weights the influence of the intersection prior. This latter term is defined by the matrices R_{ij} discussed above and given by the gradient with respect to W of the penalty $(\lambda/|\mathcal{N}_\varepsilon(i)|)\sum_{ij\in\mathscr{E}}\langle W_i, R_{ij}W_j\rangle$.

In [24], a continuous optimization approach using DC (difference of convex functions) programming was proposed to compute local minimizers of non-convex functionals similar to $\langle D(W), W\rangle$, with D given by (4.19). This "Euclidean approach"—in contrast to the geometric approach proposed here—entails to provide a DC decomposition of the intersection penalty just discussed and to *explicitly* take into account the affine constraints $W_i \in \Delta_{n-1}$. As a result, the DC approach computes a local minimizer by solving a *sequence* of convex quadratic programs.

In order to apply our present approach instead, we bypass the averaging step (3.13) because labels will most likely be different at adjacent vertices i in our random scenario, and we thus set $S(W) = L(W)$ with $L(W)$ given by (3.12) based on (4.19). Applying then algorithm (3.36) *implicitly* handles all constraints through the geometric flow and computes a local minimizer by multiplicative updates, within a small fraction of the runtime that the DC approach would need, and without compromising the quality of the solution (Fig. 12g).

4.6 Image Inpainting

Inpainting denotes the task to fill in a known region where no image data were observed or are known to be corrupted, based on the surrounding region and prior information.

Once the feature metric $d_{\mathscr{F}}$ is fixed, we assign to each pixel in the region to be inpainted *as datum* the *uninformativ feature vector* f which has the *same* distance $d_{\mathscr{F}}(f, f_j^*)$ to *every* prior feature vector $f_j^* \in \mathscr{P}_{\mathscr{F}}$. Note that there is not need to explicitly compute this data vector f. It merely represents the rule for evaluating the distance $d_{\mathscr{F}}$ if one of its arguments belongs to a region to be inpainted.

Figure 13 shows two basic examples that were used by the authors of [13,32], respectively, to examine numerically the tightness of convex relaxations of the image labeling problem. Unlike convex relaxations that constitute *outer* approximations of the combinatorically complex feasible set of assignments, our smooth non-convex approach may be considered as an *inner* approximation that yields results without the need of further rounding, i.e., the need of a post-processing step for projecting the solution of a convex relaxed problem onto the feasible set.

5 Conclusion and Further Work

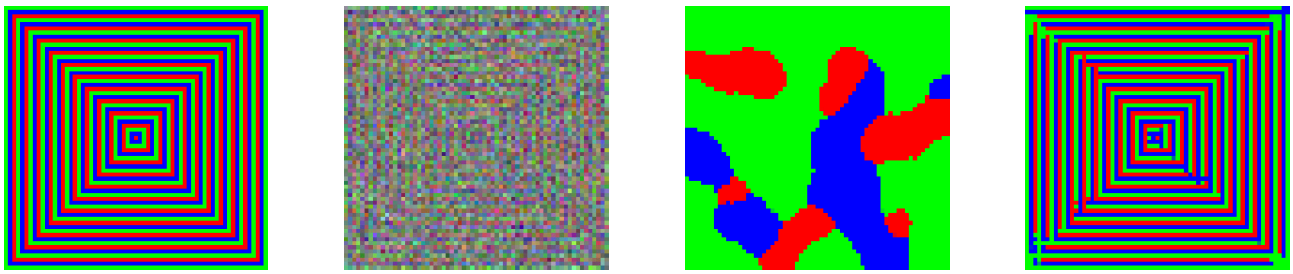
We presented a novel approach to image labeling, formulated in a smooth geometric setting. The approach contrasts with established convex and non-convex relaxations of the image labeling problem through smoothness and geometric averaging. The numerics boil down to parallel sparse updates that maximize the objective along an interior path in the feasible set of assignments and finally return a labeling. Although an elementary first-order approximation of the gradient flow was only used, the convergence rate seems competitive. In particular, a large number of labels, like in Sect. 4.4, does not slow down convergence as is the case of convex relaxations. All aspects specific to an application domain are represented by a single distance matrix D and a single user parameter ρ. This flexibility and the absence of ad hoc tuning parameters whose values do not have an intrinsic meaning should promote applications of the approach to various image labeling problems.

Aspects and open points to be addressed in future work include the following.

Numerics Many alternatives exist to the simple algorithm detailed in Sect. 3.3.3. An alternative first-order example is exponential multiplicative update [11] that results from

(a) **(b)**

Fig. 13 Two instances shown on the left in (**a**, **b**), adopted from [13,32] to study the tightness of convex *outer* relaxations of the image labeling problem. The task is both to inpaint and to label the gray regions. Our smooth non-convex approach constitutes an *inner* approximation that yields the labeling results shown on the right in (**a**, **b**, without the need of a separate rounding post-processing step that projects the solution of convex relaxations onto the feasible set of label assignments (parameters: $\rho = 1$, $|\mathcal{N}_{\mathcal{E}}(i)| = 3 \times 3$). **a** Inpainting of the regions marked by *gray* through assignment leads to the result on the right. **b** Inpainting of the regions marked by *gray* through assignment leads to the result on the *right*



(a) **(b)** **(c)** **(d)**

Fig. 14 Illustration of the influence of using *non*uniform weights for geometric averaging (2.8) based on the approximation (5.2). **a** Image structure where only patch similarity enables to recognize pixel similarity. **b** Noisy input image to which the three prior vectors *red*, *green*, and *blue* are assigned. The ℓ_1 distance between data and prior vectors was used as distance function $d_{\mathcal{F}}$. **c** Uniform labeling with weights $w_j = \frac{1}{|\mathcal{N}_{\mathcal{E}}|}$ completely fails to recover the fine image structure (**a**). **d** Using non-uniform weights based on the comparison of 7×7 patches of the noise input data (**b**) considerably enhances the labeling. Errors naturally occur in the center image region and along the diagonals where patch similarity is not sufficiently supported by other pixels locations. Parameters for (**c, d**): $\rho = 0.1$, $|\mathcal{N}_{\mathcal{E}}| = 7 \times 7$

an explicit Euler discretization of the flow (3.21) rewritten in the form

$$\frac{\mathrm{d}}{\mathrm{d}t} \log\big(W_i(t)\big) = \nabla_i J(W) - \langle W_i, \nabla_i J(W)\rangle \mathbb{1}, \qquad i \in [m]. \tag{5.1}$$

Of course, higher-order schemes respecting the geometry are conceivable as well. We point out that the inherent *smoothness* of our problem formulation paves the way for *systematic* progress.

Non-uniform geometric averaging So far, we did not exploit the degrees of freedom offered by the weights w_i, $i \in [N]$ that define the Riemannian means by the objective (2.8). By doing so, the approximation of these means due to formula (3.33) generalizes in that the geometric mean has to be replaced by the weighted geometric mean

$$\mathrm{mean}_{g,w}(\mathscr{P}) = \prod_{j\in[N]} (p^j)^{w_j}, \quad w = \Delta_{N-1} \tag{5.2}$$

that is applied componentwise to the vectors $p^j \in \mathscr{P}$. Figure 14 illustrates the influence of these weights w_j that were computed in a preprocessing step for each pixel i within the neighborhood $\mathcal{N}_{\mathcal{E}}$ by computing the distance $d_p(p_i, p_j)$ (defined as mean of the ℓ_2-distance of the respective color vectors) between 7×7 noisy data patches p_i, p_j centered at i and j, respectively, to obtain the normalized weights $w_j = \frac{\tilde{w}_j}{\langle \mathbb{1}, \tilde{w}\rangle}$, $\tilde{w}_j = \exp\big(-d_p(p_i, p_j)/\rho\big)$.

Turning this *data*-driven adaptivity of the assignment process through non-uniform weights into a *solution*-driven adaptivity, by replacing the data f by $u(W)$ due to (3.19) that evolves with W, enables an even more general way for further enhancing the assignment process.

Connection to nonlinear diffusion Referring to the discussion of neighborhood filters and nonlinear diffusion in Sect. 1.3, research making these connections explicit is attractive because, apparently, our approach is not covered by existing work.

Unsupervised scenarios. The nonexistence of a prior data set $\mathscr{P}_{\mathscr{F}}$ in applications was only briefly addressed in Sect. 4.4. In particular, the emergence of labels along with assignments and a corresponding generalization of our approach deserves attention.

Learning and updating prior information. This fundamental problem ties in with the preceding point: How can we learn and evolve prior information from many assignments over time?

We hope for a better mathematical understanding of corresponding models and that our work will stimulate corresponding research.

Acknowledgements Support by the German Research Foundation (DFG) was gratefully acknowledged, Grant GRK 1653.

6 Appendix 1: Basic Notation

For $n \in \mathbb{N}$, we set $[n] = \{1, 2, \dots, n\}$. $\mathbb{1} = (1, 1, \dots, 1)^\top$ denotes the vector with all components equal to 1, whose dimension can either be inferred from the context or is indicated by a subscript, e.g., $\mathbb{1}_n$. Vectors $v^1, v^2, \dots$ are indexed by lowercase letters and superscripts, whereas subscripts v_i, $i \in [n]$, index vector components. $e^1, \dots, e^n$ denotes the canonical orthonormal basis of $\mathbb{R}^n$.

We assume data to be indexed by a graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ with nodes $i \in \mathscr{V} = [m]$ and associated locations $x^i \in \mathbb{R}^d$, and with edges $\mathscr{E}$. A regular grid graph and $d = 2$ is the canonical example. But $\mathscr{G}$ may also be irregular due to some preprocessing like forming super-pixels, for instance, or correspond to 3D images or videos ($d = 3$). For simplicity, we call i *location* although this actually is x^i.

If $A \in \mathbb{R}^{m \times n}$, then the row and column vectors are denoted by $A_i \in \mathbb{R}^n$, $i \in [m]$ and $A^j \in \mathbb{R}^m$, $j \in [n]$, respectively, and the entries by A_{ij}. This notation of row vectors A_i is the only exception from our rule of indexing vectors stated above.

The *componentwise application of functions* $f : \mathbb{R} \to \mathbb{R}$ *to a vector* is simply denoted by $f(v)$, e.g.,

$$\forall v \in \mathbb{R}^n, \qquad \sqrt{v} := (\sqrt{v_1}, \dots, \sqrt{v_n})^\top, \tag{6.1a}$$

$$\exp(v) := \left(e^{v_1}, \dots, e^{v_n}\right)^\top \quad \text{etc.} \tag{6.1b}$$

Likewise, binary relations between vectors apply componentwise, e.g., $u \geq v \;\Leftrightarrow\; u_i \geq v_i$, $i \in [n]$, and binary componentwise operations are simply written in terms of the vectors. For example,

$$pq := (\dots, p_i q_i, \dots)^\top, \qquad \frac{p}{q} := \left(\dots, \frac{p_i}{q_i}, \dots\right)^\top, \tag{6.2}$$

where the latter operation is only applied to strictly positive vectors $q > 0$. The *support* $\operatorname{supp}(p) = \{p_i \neq 0 \colon i \in \operatorname{supp}(p)\} \subset [n]$ of a vector $p \in \mathbb{R}^n$ is the index set of all non-nonvanishing components of p.

$\langle x, y \rangle$ denotes the standard Euclidean inner product and $\|x\| = \langle x, x \rangle^{1/2}$ the corresponding norm. Other ℓ_p-norms, $1 \leq p \neq 2 \leq \infty$, are indicated by a corresponding subscript, $\|x\|_p = \left(\sum_{i \in [d]} |x_i|^p\right)^{1/p}$, except for the case $\|x\| = \|x\|_2$. For matrices $A, B \in \mathbb{R}^{m \times n}$, the canonical inner product is $\langle A, B \rangle = \operatorname{tr}(A^\top B)$ with the corresponding Frobenius norm $\|A\| = \langle A, A \rangle^{1/2}$. $\operatorname{Diag}(v) \in \mathbb{R}^{n \times n}$, $v \in \mathbb{R}^n$, is the diagonal matrix with the vector v on its diagonal.

Other basic sets and their notation are

– the positive orthant

$$\mathbb{R}^n_+ = \{p \in \mathbb{R}^n \colon p \geq 0\}, \tag{6.3}$$

– the set of strictly positive vectors

$$\mathbb{R}^n_{++} = \{p \in \mathbb{R}^n \colon p > 0\}, \tag{6.4}$$

– the ball of radius r centered at p

$$\mathbb{B}_r(p) = \{p \in \mathbb{R}^n \colon \|p\| \leq r\}, \tag{6.5}$$

– the unit sphere

$$\mathbb{S}^{n-1} = \{p \in \mathbb{R}^n \colon \|p\| = 1\}, \tag{6.6}$$

– the probability simplex

$$\Delta_{n-1} = \{p \in \mathbb{R}^n_+ \colon \langle \mathbb{1}, p \rangle = 1\} \tag{6.7}$$

– and its relative interior

$$\mathscr{S} = \mathring{\Delta}_{n-1} = \Delta_{n-1} \cap \mathbb{R}^n_{++}, \tag{6.8a}$$

$$\mathscr{S}_n = \mathscr{S} \text{ with concrete value of} n (e.g., \mathscr{S}_3), \tag{6.8b}$$

– closure (not regarded as manifold)

$$\overline{\mathscr{S}} = \Delta_{n-1}, \tag{6.9}$$

– the sphere with radius 2

$$\mathscr{N} = 2\mathbb{S}^{n-1}, \tag{6.10}$$

– the assignment manifold

$$\mathscr{W} = \mathscr{S} \times \cdots \times \mathscr{S}, \quad \text{(m times)} \tag{6.11}$$

– and its closure (not regarded as manifold)

$$\overline{\mathscr{W}} = \overline{\mathscr{S}} \times \cdots \times \overline{\mathscr{S}}, \quad \text{(m times)}. \tag{6.12}$$

For a discrete distribution $p \in \Delta_{n-1}$ and a finite set $S = \{s^1, \ldots, s^n\}$ vectors, we denote by

$$\mathbb{E}_p[S] := \sum_{i \in [n]} p_i s^i \tag{6.13}$$

the mean of S with respect to p.

Let $\mathscr{M}$ be a any differentiable manifold. Then $T_p\mathscr{M}$ denotes the tangent space at base point $p \in \mathscr{M}$ and $T\mathscr{M}$ the total space of the tangent bundle of $\mathscr{M}$. If $F\colon \mathscr{M} \to \mathscr{N}$ is a smooth mapping between differentiable manifold $\mathscr{M}$ and $\mathscr{N}$, then the differential of F at $p \in \mathscr{M}$ is denoted by

$$DF(p)\colon T_p\mathscr{M} \to T_{F(p)}\mathscr{N}, \qquad DF(p)\colon v \mapsto DF(p)[v]. \tag{6.14}$$

If $F\colon \mathbb{R}^m \to \mathbb{R}^n$, then $DF(p) \in \mathbb{R}^{n\times m}$ is the Jacobian matrix at p, and the application $DF(p)[v]$ to a vector $v \in \mathbb{R}^m$ means matrix-vector multiplication. We then also write $DF(p)v$. If $F = F(p,q)$, then $D_pF(p,q)$ and $D_qF(p,q)$ are the Jacobians of the functions $F(\cdot, q)$ and $F(p,\cdot)$, respectively.

The gradient of a differentiable function $f\colon \mathbb{R}^n \to \mathbb{R}$ is denoted by $\nabla f(x) = \big(\partial_1 f(x), \ldots, \partial_n f(x)\big)^\top$, whereas the Riemannian gradient of a function $f\colon \mathscr{M} \to \mathbb{R}$ defined on Riemannian manifold $\mathscr{M}$ is denoted by $\nabla_{\mathscr{M}} f$. Eq. (2.5) recalls the formal definition.

The *exponential mapping* [21, Def. 1.4.3]

$$\mathrm{Exp}_p\colon T_p\mathscr{M} \to \mathscr{M}, \quad v \mapsto \mathrm{Exp}_p(v) = \gamma_v(1), \tag{6.15a}$$

$$\gamma_v(0) = p, \ \dot\gamma_v(0) = \frac{\mathrm{d}}{\mathrm{d}t}\gamma_v(t)\big|_{t=0} = v, \tag{6.15b}$$

maps the tangent vector v to the point $\gamma_v(1) \in \mathscr{M}$, uniquely defined by the *geodesic curve* $\gamma_v(t)$ emanating at p in direction v. $\gamma_v(t)$ is the shortest path on $\mathscr{M}$ between the points $p, q \in \mathscr{M}$ that γ_v connects. This minimal length equals the *Riemannian distance* $d_{\mathscr{M}}(p,q)$ induced by the *Riemannian metric*, denoted by

$$\langle u, v\rangle_p, \tag{6.16}$$

i.e., the inner product on the tangent spaces $T_p\mathscr{M}$, $p \in \mathscr{M}$, that smoothly varies with p. Existence and uniqueness of geodesics will not be an issue for the manifolds $\mathscr{M}$ considered in this paper.

Remark 8 The exponential mapping Exp_p should not be confused with

– the exponential function e^v used, e.g., in (6.1);
– the mapping $\exp_p\colon T_p\mathscr{S} \to \mathscr{S}$ defined by Eq. (3.8a).

The abbreviations "l.h.s." and "r.h.s." mean *left-hand side* and *right-hand side* of some equation, respectively. We abbreviate *with respect to* by "wrt."

7 Appendix 2: Proofs and Further Details

7.1 Proofs of Section 2

Proof (of Lemma 1) Let $p \in \mathscr{S}$ and $v \in T_p\mathscr{S}$. We have

$$D\psi(p) = \mathrm{Diag}(p)^{-1/2} \tag{7.1}$$

and $\big\langle \psi(p), D\psi(p)[v]\big\rangle = \langle 2\sqrt{p}, \frac{v}{\sqrt{p}}\rangle = 2\langle \mathbb{1}, v\rangle = 0$, that is, $D\psi(p)[v] \in T_{\psi(p)}\mathscr{N}$. Furthermore,

$$\big\langle D\psi(p)[u], D\psi(p)[v]\big\rangle = \big\langle u/\sqrt{p}, v/\sqrt{p}\big\rangle \overset{(2.1)}{=} \langle u, v\rangle_p, \tag{7.2}$$

i.e., the Riemannian metric is preserved and hence also the length $L(s)$ of curves $s(t) \in \mathscr{N}$, $t \in [a,b]$: Put $\gamma(t) = \psi^{-1}\big(s(t)\big) = \frac{1}{4}s^2(t) \in \mathscr{S}$, $t \in [a,b]$. Then $\dot\gamma(t) = \frac{1}{2}s(t)\dot s(t) = \frac{1}{2}\psi\big(\gamma(t)\big)\dot s(t) = \sqrt{\gamma(t)}\dot s(t)$ and

$$L(s) = \int_a^b \|\dot s(t)\|\mathrm{d}t = \int_a^b \left\langle \frac{\dot\gamma(t)}{\sqrt{\gamma(t)}}, \frac{\dot\gamma(t)}{\sqrt{\gamma(t)}}\right\rangle^{1/2} \mathrm{d}t \tag{7.3a}$$

$$\overset{(2.1)}{=} \int_a^b \|\dot\gamma(t)\|_{\gamma(t)}\mathrm{d}t = L(\gamma). \tag{7.3b}$$

□

Proof (of Prop. 1) Setting $g\colon \mathscr{N} \to \mathbb{R}$, $q \mapsto g(s) := f\big(\psi^{-1}(s)\big)$ with $s = \psi(p) = 2\sqrt{p}$ from (2.3), we have

$$\nabla_{\mathscr{N}} g(s) = \left(I - \frac{s}{\|s\|}\frac{s^\top}{\|s\|}\right)\nabla g(s), \tag{7.4}$$

because the 2-sphere $\mathscr{N} = 2\mathbb{S}^{n-1}$ is an embedded submanifold, and hence the Riemannian gradient equals the orthogonal projection of the Euclidean gradient onto the tangent space. Pulling back the vector field $\nabla_{\mathscr{N}} g$ by ψ using

$$\nabla g(s) = \nabla f\big(\psi^{-1}(s)\big) = \nabla f\Big(\frac{1}{4}s^2\Big) = \frac{1}{2}s\big(\nabla f(p)\big), \tag{7.5}$$

we get with (7.1), (7.4) and $\|s\| = 2$ and hence $s/\|s\| = \frac{1}{2}\psi(p) = \sqrt{p}$

$$\nabla f_{\mathscr{S}}(p) = \big(D\psi(p)\big)^{-1}\big(\nabla_{\mathscr{N}} g(\psi(p))\big) \tag{7.6a}$$

$$= \mathrm{Diag}(\sqrt{p})\Big(\big(I - \sqrt{p}\sqrt{p}^{\top}\big)\sqrt{p}\big(\nabla f(p)\big)\Big) \tag{7.6b}$$

$$= p\big(\nabla f(p)\big) - \langle p, \nabla f(p)\rangle p, \tag{7.6c}$$

which equals (2.6). We finally check that $\nabla f_{\mathscr{S}}(p)$ satisfies (2.5) (with $\mathscr{S}$ in place of $\mathscr{M}$). Using (2.1), we have

$$\langle \nabla f_{\mathscr{S}}(p), v\rangle_p = \Big\langle \sqrt{p}\big(\nabla f(p)\big) - \langle p, \nabla f(p)\rangle\sqrt{p}, \frac{v}{\sqrt{p}}\Big\rangle \tag{7.7a}$$

$$= \langle \nabla f(p), v\rangle - \langle p, \nabla f(p)\rangle\langle \mathbb{1}, v\rangle \tag{7.7b}$$

$$\overset{(2.2)}{=} \langle \nabla f(p), v\rangle, \quad \forall v \in T_p\mathscr{S}. \tag{7.7c}$$

□

Proof (of Prop. 2) The geodesic on the 2-sphere emanating at $s(0) \in \mathscr{N}$ in direction $w = \dot{s}(0) \in T_{s(0)}\mathscr{N}$ is given by

$$s(t) = s(0)\cos\Big(\frac{\|w\|}{2}t\Big) + 2\frac{w}{\|w\|}\sin\Big(\frac{\|w\|}{2}t\Big). \tag{7.8}$$

Setting $s(0) = \psi(p)$ and $w = D\psi(p)[v] = v/\sqrt{p}$, the geodesic emanating at $p = \gamma_v(0)$ in direction v is given by $\psi^{-1}\big(s(t)\big)$ due to Lemma 1, which results in (2.7a) after elementary computations. □

7.2 Proofs of Section 3 and Further Details

Proof (of Prop. 3) We have $p = \exp_p(0)$ and

$$\frac{\mathrm{d}}{\mathrm{d}t}\exp_p(ut) = \frac{\langle p, e^{ut}\rangle p e^{ut}u - p e^{ut}\langle p, e^{ut}u\rangle}{\langle p, e^{ut}\rangle^2} \tag{7.9a}$$

$$= p(t)u - \langle p(t), u\rangle p(t), \tag{7.9b}$$

which confirms (3.10), is equal to (3.9) at $t = 0$ and hence yields the first expression of (3.11). The second expression of (3.11) follows from a Taylor expansion of (2.7a)

$$\gamma_v(t) \approx p + vt + \frac{1}{4}\big(v_p^2 - \|v_p\|^2 p\big)t^2, \qquad v_p = \frac{v}{\sqrt{p}}. \tag{7.10}$$

□

Proof (of Lemma 4) By construction, $S(W) \in \mathscr{W}$, that is, $S_i(W) \in \mathscr{S}$, $i \in [m]$. Consequently,

$$0 \le J(W) = \sum_{i\in[m]} \langle S_i(W), W_i\rangle \le \sum_{i\in[m]} \|S_i(W)\|\|W_i\| < m. \tag{7.11}$$

The upper bound corresponds to matrices $\overline{W}^* \in \overline{\mathscr{W}}$ and $S(\overline{W}^*)$ where for each $i \in [m]$, *both* $\overline{W}_i^*$ and $S_i(\overline{W}^*)$ equal the *same* unit vector e^{k_i} for some $k_i \in [m]$. □

Proof (Explicit form of (3.27)) The matrices $T^{ij}(W) = \frac{\partial}{\partial W_{ij}}S(W)$ are implicitly given through the optimality condition (2.9) that each vector $S_k(W)$, $k \in [m]$, defined by (3.13) has to satisfy

$$S_k(W) = \mathrm{mean}_{\mathscr{S}}\{L_r(W_r)\}_{r\in\tilde{\mathscr{N}}_{\mathscr{E}}(k)} \tag{7.12a}$$

$$\Leftrightarrow \quad 0 = \sum_{r\in\tilde{\mathscr{N}}_{\mathscr{E}}(k)} \mathrm{Exp}^{-1}_{S_k(W)}\big(L_r(W_r)\big). \tag{7.12b}$$

Writing

$$\phi\big(S_k(W), L_r(W_r)\big) := \mathrm{Exp}^{-1}_{S_k(W)}\big(L_r(W_r)\big), \tag{7.13}$$

while temporarily dropping below W as argument to simplify the notation, and using the indicator function $\delta_{\mathrm{P}} = 1$ if the predicate P = true and $\delta_{\mathrm{P}} = 0$ otherwise, we differentiate the optimality condition on the r.h.s. of (7.12),

$$0 = \frac{\partial}{\partial W_{ij}} \sum_{r\in\tilde{\mathscr{N}}_{\mathscr{E}}(k)} \phi\big(S_k(W), L_r(W_r)\big) \tag{7.14a}$$

$$= \sum_{r\in\tilde{\mathscr{N}}_{\mathscr{E}}(k)} \Big(D_{S_k}\phi(S_k, L_r)\Big[\frac{\partial}{\partial W_{ij}}S_k(W)\Big] \tag{7.14b}$$

$$+ \delta_{i=r} D_{L_r}\phi(S_k, L_r)\Big[\frac{\partial}{\partial W_{rj}}L_r(W_r)\Big]\Big) \tag{7.14c}$$

$$= \Big(\sum_{r\in\tilde{\mathscr{N}}_{\mathscr{E}}(k)} D_{S_k}\phi(S_k, L_r)\Big)\Big(\frac{\partial}{\partial W_{ij}}S_k(W)\Big) \tag{7.14d}$$

$$+ \delta_{i\in\tilde{\mathscr{N}}_{\mathscr{E}}(k)} D_{L_i}\phi(S_k, L_i)\Big(\frac{\partial}{\partial W_{ij}}L_i(W_i)\Big) \tag{7.14e}$$

$$=: H^k(W)\Big(\frac{\partial}{\partial W_{ij}}S_k(W)\Big) + h^{k,ij}(W). \tag{7.14f}$$

Since the vectors $\phi(S_k, L_r)$ given by (7.13) are the negative Riemannian gradients of the (locally) strictly convex objectives (2.8) defining the means S_k [21, Thm. 4.6.1], the regularity of the matrices $H^k(W)$ follows. Thus, using (7.14f) and defining the matrices

$$T^{ij}(W) \in \mathbb{R}^{m\times n}, \quad T^{ij}_{kl}(W) := \frac{\partial}{\partial S_{kl}(W)}W_{ij}, \quad i,k \in [m],\ j,l \in [n], \tag{7.15}$$

results in (3.27). The explicit form of this expression results from computing and inserting into (7.14f) the corresponding Jacobians $D_p\phi(p,q)$ and $D_q\phi(p,q)$ of

$$\phi(p,q) = \mathrm{Exp}_p^{-1}(q) \tag{7.16a}$$

$$= \frac{d_{\mathscr{S}}(p,q)}{\sqrt{1-\langle\sqrt{p},\sqrt{q}\rangle^2}}\big(\sqrt{pq} - \langle\sqrt{p},\sqrt{q}\rangle p\big), \tag{7.16b}$$

and

$$\frac{\partial}{\partial W_{ij}} L_i(W_i) = \frac{e^{-U_{ij}}}{\langle W_i, e^{-U_i}\rangle}\big(e^j - L_i(W_i)\big). \tag{7.16c}$$

The term (7.16b) results from mapping back the corresponding vector from the 2-sphere $\mathscr{N}$,

$$\operatorname{Exp}_p^{-1}(q) = -\big(D\psi(p)\big)^{-1}\Big(\frac{1}{2}\nabla_{\mathscr{N}} d^2_{\mathscr{N}}\big(\psi(p),\psi(q)\big)\Big), \tag{7.17}$$

where ψ is the sphere map (2.3) and $d_{\mathscr{N}}$ is the geodesic distance on $\mathscr{N}$. The term (7.16c) results from directly evaluating (3.12). □

Proof (of Lemma 5) We first compute $\exp_p^{-1}$. Suppose

$$q = \exp_p(u) = \frac{pe^u}{\langle p, e^u\rangle}, \qquad p,q \in \mathscr{S}, \quad u \in \mathbb{R}^n. \tag{7.18}$$

Then

$$\log(q) = \log(p) + u - \log(\langle p, e^u\rangle)\mathbb{1}, \tag{7.19a}$$

$$\log(\langle p, e^u\rangle) = \frac{1}{n}\langle \mathbb{1}, \log(p) - \log(q)\rangle, \tag{7.19b}$$

and

$$u = \exp_p^{-1}(q) = \Big(I - \frac{1}{n}\mathbb{1}\mathbb{1}^\top\Big)\big(\log(q) - \log(p)\big). \tag{7.20}$$

Thus, in view of (3.9), we approximate

$$\begin{aligned}
\operatorname{Exp}_p^{-1}(q) &\approx v \\
&= \big(\operatorname{Diag}(p) - pp^\top\big)u && (7.21a)\\
&= \Big(\operatorname{Diag}(p) - \frac{1}{n}p\mathbb{1}^\top - pp^\top + \frac{1}{n}p\mathbb{1}^\top\Big)\log\Big(\frac{q}{p}\Big) && (7.21b)\\
&= \big(\operatorname{Diag}(p) - pp^\top\big)\log\Big(\frac{q}{p}\Big). && (7.21c)
\end{aligned}$$

Applying this to the point set $\mathscr{P}$, i.e., setting

$$v^i = \big(\operatorname{Diag}(p) - pp^\top\big)\log\frac{p^i}{p}, \qquad i \in [N], \tag{7.22}$$

step (3) of (3.31) yields

$$v := \frac{1}{N}\sum_{i\in[N]} v^i = \frac{1}{N}\big(\operatorname{Diag}(p) - pp^\top\big)$$

$$\begin{aligned}
&\Big(\sum_{i\in[N]}\log(p^i) - N\log(p)\Big) && (7.23a)\\
&= \big(\operatorname{Diag}(p) - pp^\top\big)\log\Big(\frac{1}{p}\Big(\prod_{i\in[N]} p^i\Big)^{\frac{1}{N}}\Big) && (7.23b)\\
&= \big(\operatorname{Diag}(p) - pp^\top\big)\log\Big(\frac{\operatorname{mean}_g(\mathscr{P})}{p}\Big) && (7.23c)\\
&=: \big(\operatorname{Diag}(p) - pp^\top\big)u. && (7.23d)
\end{aligned}$$

Finally, approximating step (4) of (3.31) results in view of Prop. 3 in the update of p

$$\exp_p(u) = \frac{pe^u}{\langle p, e^u\rangle} = \frac{\operatorname{mean}_g(\mathscr{P})}{\langle \mathbb{1}, \operatorname{mean}_g(\mathscr{P})\rangle}. \tag{7.24}$$

□

References

1. Amari, S.I., Nagaoka, H.: Methods of Information Geometry. American Mathematical Society, Oxford University Press, Oxford (2000)
2. Aujol, J.F., Gilboa, G., Chan, T., Osher, S.: Structure-texture image decomposition-modeling, algorithms, and parameter selection. Int. J. Comput. Vis. **67**(1), 111–136 (2006)
3. Ball, K.: An elementary introduction to modern convex geometry. In: Levy, S. (ed.) Flavors of Geometry, MSRI Publ., vol. 31, pp. 1–58. Cambridge University Press (1997)
4. Bayer, D., Lagarias, J.: The nonlinear geometry of linear programming. I. Affine and projective scaling trajectories. Trans. Am. Math. Soc. **314**(2), 499–526 (1989)
5. Bayer, D., Lagarias, J.: The nonlinear geometry of linear programming. II. Legendre transform coordinates and central trajectories. Trans. Am. Math. Soc. **314**(2), 527–581 (1989)
6. Bishop, C.: Pattern Recognition and Machine Learning. Springer, Berlin (2006)
7. Bomze, I., Budinich, M., Pelillo, M., Rossi, C.: Annealed replication: a new heuristic for the maximum clique problem. Discr. Appl. Math. **121**, 27–49 (2002)
8. Bomze, I.M.: Regularity versus degeneracy in dynamics, games, and optimization: a unified approach to different aspects. SIAM Rev. **44**(3), 394–414 (2002)
9. Buades, A., Coll, B., Morel, J.: A review of image denoising algorithms, with a new one. SIAM Multiscale Model. Simul. **4**(2), 490–530 (2005)
10. Buades, A., Coll, B., Morel, J.M.: Neighborhood filters and PDEs. Numer. Math. **105**, 1–34 (2006)
11. Cabrales, A., Sobel, J.: On the limit points of discrete selection dynamics. J. Econ. Theory **57**, 407–419 (1992)
12. Čencov, N.: Statistical Decision Rules and Optimal Inference. American Mathematical Society, Providence (1982)
13. Chambolle, A., Cremers, D., Pock, T.: A convex approach to minimal partitions. SIAM J. Imaging Sci. **5**(4), 1113–1158 (2012)
14. Chan, T., Esedoglu, S., Nikolova, M.: Algorithms for Finding Global Minimizers of Image Segmentation and Denoising Models. SIAM J. Appl. Math. **66**(5), 1632–1648 (2006)
15. Hérault, L., Horaud, R.: Figure-ground discrimination: a combinatorial optimization approach. IEEE Trans. Pattern Anal. Mach. Intell. **15**(9), 899–914 (1993)
16. Heskes, T.: Convexity arguments for efficient minimization of the Bethe and Kikuchi free energies. J. Artif. Intell. Res. **26**, 153–190 (2006)

17. Hofbauer, J., Siegmund, K.: Evolutionary game dynamics. Bull. Am. Math. Soc. **40**(4), 479–519 (2003)
18. Hofman, T., Buhmann, J.: Pairwise data clustering by deterministic annealing. IEEE Trans. Pattern Anal. Mach. Intell. **19**(1), 1–14 (1997)
19. Horst, R., Tuy, H.: Global Optimization: Deterministic Approaches, 3rd edn. Springer, Berlin (1996)
20. Hummel, R., Zucker, S.: On the foundations of the relaxation labeling processes. IEEE Trans. Pattern Anal. Mach. Intell. **5**(3), 267–287 (1983)
21. Jost, J.: Riemannian Geometry and Geometric Analysis, 4th edn. Springer, Berlin (2005)
22. Kappes, J., Andres, B., Hamprecht, F., Schnörr, C., Nowozin, S., Batra, D., Kim, S., Kausler, B., Kröger, T., Lellmann, J., Komodakis, N., Savchynskyy, B., Rother, C.: A comparative study of modern inference techniques for structured discrete energy minimization problems. Int. J. Comput. Vis. **115**(2), 155–184 (2015)
23. Kappes, J., Savchynskyy, B., Schnörr, C.: A bundle approach to efficient MAP-inference by Lagrangian relaxation. In: Proc. CVPR (2012)
24. Kappes, J., Schnörr, C.: MAP-inference for highly-connected graphs with DC-programming. In: Pattern Recognition—30th DAGM Symposium, LNCS, vol. 5096, pp. 1–10. Springer (2008)
25. Karcher, H.: Riemannian center of mass and mollifier smoothing. Commun. Pure Appl. Math. **30**, 509–541 (1977)
26. Karcher, H.: Riemannian center of mass and so called karcher mean. arxiv:1407.2087 (2014)
27. Kass, R.: The geometry of asymptotic inference. Stat. Sci. **4**(3), 188–234 (1989)
28. Kolmogorov, V.: Convergent tree-reweighted message passing for energy minimization. IEEE Trans. Pattern Anal. Mach. Intell. **28**(10), 1568–1583 (2006)
29. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts? IEEE Trans. Pattern Anal. Mach. Intell. **26**(2), 147–159 (2004)
30. Ledoux, M.: The Concentration of Measure Phenomenon. American Mathematical Society, Providence (2001)
31. Lellmann, J., Lenzen, F., Schnörr, C.: Optimality bounds for a variational relaxation of the image partitioning problem. J. Math. Imaging Vis. **47**(3), 239–257 (2013)
32. Lellmann, J., Schnörr, C.: Continuous multiclass labeling approaches and algorithms. SIAM J. Imaging Sci. **4**(4), 1049–1096 (2011)
33. Losert, V., Alin, E.: Dynamics of games and genes: discrete versus continuous time. J. Math. Biol. **17**(2), 241–251 (1983)
34. Luce, R.: Individual Choice Behavior: A Theoretical Analysis. Wiley, New York (1959)
35. Milanfar, P.: A tour of modern image filtering. IEEE Signal Process. Mag. **30**(1), 106–128 (2013)
36. Milanfar, P.: Symmetrizing smoothing filters. SIAM J. Imaging Sci. **6**(1), 263–284 (2013)
37. Montúfar, G., Rauh, J., Ay, N.: On the fisher metric of conditional probability polytopes. Entropy **16**(6), 3207–3233 (2014)
38. Nesterov, Y., Todd, M.: On the riemannian geometry defined by self-concordant barriers and interior-point methods. Found. Comput. Math. **2**, 333–361 (2002)
39. Orland, H.: Mean-field theory for optimization problems. J. Phys. Lett. **46**(17), 763–770 (1985)
40. Pavan, M., Pelillo, M.: Dominant sets and pairwise clustering. IEEE Trans. Pattern Anal. Mach. Intell. **29**(1), 167–172 (2007)
41. Pelillo, M.: The dynamics of nonlinear relaxation labeling processes. J. Math. Imaging Vis. **7**, 309–323 (1997)
42. Pelillo, M.: Replicator equations, maximal cliques, and graph isomorphism. Neural Comput. **11**(8), 1933–1955 (1999)
43. Rosenfeld, A., Hummel, R., Zucker, S.: Scene labeling by relaxation operations. IEEE Trans. Syst. Man Cybern. **6**, 420–433 (1976)
44. Singer, A., Shkolnisky, Y., Nadler, B.: Diffusion interpretation of non-local neighborhood filters for signal denoising. SIAM J. Imaging Sci. **2**(1), 118–139 (2009)
45. Sutton, R., Barto, A.: Reinforcement Learning, 2nd edn. MIT Press, Cambridge (1999)
46. Swoboda, P., Shekhovtsov, A., Kappes, J., Schnörr, C., Savchynskyy, B.: Partial optimality by pruning for MAP-inference with general graphical models. IEEE Trans. Patt. Anal. Mach. Intell. **38**(7), 1370–1382 (2016)
47. Wainwright, M., Jordan, M.: Graphical models, exponential families, and variational inference. Found. Trends Mach. Learn. **1**(1–2), 1–305 (2008)
48. Weickert, J.: Anisotropic Diffusion in Image Processing. B.G Teubner, Leipzig (1998)
49. Werner, T.: A linear programming approach to max-sum problem: a review. IEEE Trans. Pattern Anal. Mach. Intell. **29**(7), 1165–1179 (2007)
50. Yedidia, J., Freeman, W., Weiss, Y.: Constructing free-energy approximations and generalized belief propagation algorithms. IEEE Trans. Inf. Theory **51**(7), 2282–2312 (2005)

Freddie Åström received his MSc degree in Biomedical Engineering from Linköping University, Sweden, in 2009 and his PhD degree from the Computer Vision Laboratory at Linköping University 2015. He is currently a postdoctoral researcher in the Research Training Group (RTG) 1653 at Heidelberg University, Germany. His research interests include variational methods, differential geometry, optimization, and their applications for color image processing and vision.

Stefania Petra received her BSc degree in Mathematics and Computer Science in 2001 and her MSc in Mathematics in 2003 from the Babeş-Bolyai University of Cluj-Napoca. In 2006, she received her PhD degree from the University of Würzburg in the field of numerical optimization. She continued working as a research fellow in the University of Mannheim and Heidelberg in the field of mathematical image processing. During 2013–2015, she was a Margarete von Wrangel-Fellow within the postdoctoral lecture qualification program of the Ministry of Science, Research and Arts of the state of Baden-Württemberg in Germany. Since 2015 she is an Assistant Professor at the Heidelberg University where she is leading the Mathematical Imaging Group at the Institute of Applied Mathematics. Her research interests include compressed sensing, mathematical models of image analysis with an emphasis on tomography and numerical optimization.

Bernhard Schmitzer obtained an MSc in theoretical physics at Imperial College, London (2010), and a doctorate in Applied Mathematics from Heidelberg University (2014) where he was a member of the Image and Pattern Analysis group of Christoph Schnörr. From 2014 to 2016, he was a postdoctoral researcher at the CEREMADE at Paris-Dauphine University. Currently, he is working at the University of Münster. His research interests are optimal transport and convex optimization for applications in imaging and vision.

Christoph Schnörr received his degrees from the Technical University of Karlsruhe (1991) and the University of Hamburg (1998), respectively. He became Full Professor at the University of Mannheim in 1998. In 2008, he joined Heidelberg University where he is heading the Image and Pattern Analysis Group at the Institute of Applied Mathematics. Together with colleagues, he has set up and is co-directing the Heidelberg Collaboratory for Image Processing (HCI). He also acts as coordinator of a Research Training Group on Probabilistic Graphical Models with Applications to Image Analysis, funded by the German Research Foundation (DFG). His research interests include mathematical models of image analysis and numerical optimization.