A Second Order TV-type Approach for Inpainting and Denoising Higher Dimensional Combined Cyclic and Vector Space Data

Ronny Bergmann, Andreas Weinmann

January 13, 2015

Abstract

In this paper we consider denoising and inpainting problems for higher dimensional combined cyclic and linear space valued data. These kind of data appear when dealing with nonlinear color spaces such as HSV, and they can be obtained by changing the space domain of, e.g., an optical flow field to polar coordinates. For such nonlinear data spaces, we develop algorithms for the solution of the corresponding second order total variation (TV) type problems for denoising, inpainting as well as the combination of both. We provide a convergence analysis and we apply the algorithms to concrete problems.

Keywords: Higher order total variation minimization, vector-valued TV, cyclic data, combined denoising and inpainting, cyclic proximal point algorithm.

AMS classification: 65K05, 65K10, 68U10, 94A08

1 Introduction

One of the most well known methods for edge-preserving image denoising is the variational approach minimizing the Rudin-Osher-Fatemi (ROF) functional [66]. In its basic form, it deals with scalar data. Related variational approaches for vector space valued data have gained a lot of interest in the literature and are still topic of ongoing research; we exemplarily refer to [11, 63, 40, 59] and the references therein. In this paper, we consider TV-type functionals incorporating first and second order differences for the nonlinear data spaces which combine vector space valued data —in the following called linear space data to avoid confusion —and vectors of cyclic data. In these spaces, we deal with denoising and inpainting problems as well as simultaneous inpainting and denoising problems.
Image inpainting is a problem arising in many applications in image processing, image analysis and related fields. Examples are restoring scratches in photographs, removal of superimposed objects, dealing with an area removed by a user, digital zooming as well as edge decoding. Principally, any missing data situation —whatever the reason might be— results in an inpainting problem. This is not restricted to 2D images. Further examples are defects in audio and video recordings, or in seismic data processing. In this respect, also interpolation, approximation, and extrapolation problems may be viewed as inpainting problems. We recommend the survey [44] and Chapter 6 of [26] as well as [15, 17] for an overview on inpainting and for further applications. There are various conceptional different approaches to inpainting, cf. [26, 44] and [17] which also includes some comparison. Among these are methods based on linear transforms from harmonic analysis such as curvelets and shearlets which are combined with a sparsity approach based on $\ell_1$ minimization on the corresponding coefficients [16, 32, 31, 49]. Other approaches are based on (often nonlinear) PDE and variational models, cf., e.g., [12, 18, 22, 23, 33, 56, 55, 57, 58, 70, 73]. In general, exemplar-based and sparsity-based methods perform better for filling large texture areas, whereas diffusion-based and variational techniques yield better results for natural images. Among the variational techniques applied, total variation (TV) minimization is one of the prominent models. The minimizer of the corresponding TV functional yields the inpainted image.

The first TV regularized model was proposed in [66] for denoising. It was first applied to missing data situations/inpainting in [3, 25]. Further references for TV based image inpainting are [14, 27, 70]. In contrast to classical methods, the results are typically not over-smoothed; however, it is well known that these minimizers very often show ‘staircasing’ effects, i.e. the result is often piecewise constant, although the underlying signal varies smoothly in the corresponding regions. In order to avoid staircasing, higher order and, in particular, second order differences and derivatives (in a continuous domain setting), are often employed. References are the pioneering work [19] as well as [13, 20, 24, 29, 30, 46, 51, 53, 54, 67, 68, 69]. TV functionals for linear space valued data were considered in [11] in the context of linear color spaces. A total generalized variational model can be found, e.g., in [59]. The authors describe a method minimizing the rank of the Jacobian matrix computed in each pixel and obtain a model for denoising linear space valued color data. Second order total generalized variation was generalized for tensor fields in [76].

However, in many applications, data having values in nonlinear spaces appear. Examples are diffusion tensor images [4, 62], color images based on non-flat color models [21, 48, 50, 77] or motion group-valued data [65, 75]. Due to its importance, processing such manifold valued data has gained a lot of interest in recent years. To mention only some examples, wavelet-type multi scale transforms for manifold data have been considered in [43, 75, 80]. Statistical issues on Riemannian manifolds are the topic of [9, 10, 36, 60, 61] and circular data are, in particular, considered in [35, 47]. Furthermore, manifold-valued partial differential equations are studied in [28, 42, 74].

Concerning TV functionals for manifold-valued data, an analysis from a theoretical viewpoint has been carried out in [38, 39]. These papers extend previous work [37] on $S^1$-valued functions where, in particular, the existence of minimizers of certain TV-type energies is shown. An algorithm for TV minimization on Riemannian manifolds was proposed in [52]. This approach uses a reformulation as a multi label optimization problem with an infinite number of labels and a subsequent convex relaxation. An approach for linear spaces using a relaxation of the
label optimization problem as well was presented in [41]. In concrete applications, the number of labels grows rapidly with the dimension of the data space. First order TV minimization for $S^1$-valued data has been considered in [71, 72]. In particular, these authors consider inpainting for manifold-valued data via cyclic and parallel proximal point algorithms were proposed in [81]. Again, a first order setup is considered here. A second order TV setup for denoising $S^1$-valued data based on cyclic proximal point algorithms was established in [5].

In this paper, we consider inpainting, denoising as well as combined inpainting and denoising problems for combined cyclic and linear space valued images. For example, such data appear when dealing with nonlinear color spaces such as HSV, HSL, HSI or HCL. Another example appears in the context of optical flows. When considering the flow vectors between consecutive images in polar coordinates which means separating magnitude and direction, the resulting data takes its values in $\mathbb{R} \times S$. This approach is natural and interesting and seems promising to improve the results obtained with the usual $\mathbb{R}^2$-valued approach. We consider two variational models for the inpainting problem for higher dimensional combined cyclic and vector space valued data based on a second order TV-type formulation. The first model deals with the noise free situation whereas the second one also considers the noisy case combining denoising and inpainting. In particular, pure denoising is covered by specifying the inpainting area as the empty set. In contrast to first order TV methods, our higher order approaches avoid unwanted staircasing effects, also in our nonlinear setting.

Contributions. For combined cyclic and linear space data, we derive solvers for the inpainting and the combined denoising and inpainting problem which in particular includes the case of pure denoising. Our solvers are cyclic proximal point algorithms. The needed proximal mappings for the higher dimensional situation do not simply arise by component-wise application of the one dimensional situation. The reason for this is that the natural second differences defined couple the components of the range space. Furthermore, in contrast to pure denoising, additional/different proximal mappings are needed when dealing with the inpainting problem due to the additional constraints. We provide a convergence analysis for both the noisy and the noise free model based algorithms developed in this paper. For both algorithms, we show the convergence to a minimizer under certain restrictions which are typical when dealing with nonlinear data. We apply our algorithms to denoising, inpainting and combined denoising and inpainting in the nonlinear HSV color space. Furthermore, we apply our algorithms for denoising frames in volumetric phase-valued data – in our case, frames of a 2D film. Our approach is based on utilizing the neighboring $k$ frames to incorporate the temporal neighborhood. The idea generalizes to arbitrary data spaces and volumes consisting of layers of 2D data.

Outline of the paper. In Section 2 we introduce the variational models we consider for inpainting and denoising of combined cyclic and linear space data in this paper. We start with vector space data in Section 2.1; then we define absolute differences for combined cyclic and vector space data in Section 2.2 which allow us to derive the corresponding variational models for combined cyclic and vector space data. This is done in Section 2.3 for both inpainting noise free combined cyclic and vector space data as well as inpainting and denoising combined cyclic and vector space data.
In Section 3 we develop algorithms for minimizing the variational models introduced previously. These algorithms base on the cyclic proximal point algorithm we present in Section 3.1. We present explicit formulas for the proximal mappings needed for inpainting in Section 3.2. Using these explicit representations, we derive a cyclic proximal point algorithm for inpainting both the noisy and noise free combined cyclic and vector space data in Section 3.3. The convergence analysis of both algorithms is the topic of Section 3.4.

Finally, in Section 4 we apply the derived algorithms to various concrete situations. We consider denoising data living in the nonlinear HSV color space in Section 4.1. Then we consider inpainting in such color spaces: The noise free setup is the topic of Section 4.2, and combined inpainting and denoising is performed in Section 4.3. Finally, we apply our algorithms for denoising frames of a $S^1$-valued 2D film in Section 4.4.

2 Second order variational models for inpainting and denoising combined cyclic and linear space data

In this section we derive models for denoising, inpainting as well as simultaneous inpainting and denoising data having cyclic and linear space components. In Subsection 2.1, we first concentrate on introducing the considered models based on first and second order absolute finite differences restricting to the linear space setting. In Subsection 2.2 we obtain suitable definitions for absolute differences for combined cyclic and vector space data. In Subsection 2.3, we use these definitions to obtain inpainting and simultaneous inpainting and denoising models for combined cyclic and linear data. In particular, denoising is covered by considering the empty set as inpainting region.

2.1 Inpainting and denoising vector space data

The Rudin-Osher-Fatemi (ROF) functional [66]

$$\sum_{i,j} (f_{i,j} - x_{i,j})^2 + \lambda \sum_{i,j} |\nabla x_{i,j}|, \quad \lambda > 0,$$

is one of the most well known and most popular functionals in variational image processing. In its penalized form, it consists of two terms: the first term measures the distance to the data $f$ the second term is a TV regularizer (in a discrete anisotropic form), where $\nabla$ denotes the discrete gradient operator, usually implemented as first order forward differences in vertical and horizontal directions. In this form, the ROF Model is typically used for denoising purposes. To avoid the appearing staircasing effect, often higher order and, in particular, second order differences (respectively, derivatives, in a continuous domain setting) are employed [19, 13, 20, 24, 29, 30, 46, 51, 53, 54, 67, 68, 69].

**Denoising vector space data.** For pure denoising we consider the discrete second order TV-type functional

$$J(x) = F(x; f) + \alpha TV_1(x) + \beta TV_2(x) + \gamma TV_{1,1}(x). \quad (1)$$
Here the data term $F(x; f)$ for given data $f$ reads

$$F(x; f) = \frac{1}{2} \sum_{i,j=1}^{N,M} d(f_{i,j}, x_{i,j})^2,$$

where $d$ is a distance on the data space. For data living in a vector space, $d(f_{i,j}, x_{i,j}) = \|f_{i,j} - x_{i,j}\|$ is an appropriate choice. The first order difference component $\alpha \text{TV}_1(x)$ is given by

$$\alpha \text{TV}_1(x) = \alpha_1 \sum_{i,j=1}^{N,M-1} D_1(x_{i,j}, x_{i+1,j}) + \alpha_2 \sum_{i,j=1}^{N-1,M} D_1(x_{i,j}, x_{i,j+1}),$$

$$+ \frac{\alpha_3}{\sqrt{2}} \sum_{i,j=1}^{N-1,M-1} D_1(x_{i,j}, x_{i+1,j+1}) + \frac{\alpha_4}{\sqrt{2}} \sum_{i,j=1}^{N,M-1} D_1(x_{i,j+1}, x_{i,j+1}).$$

(2)

Again, if the data space is a vector space, any norm of the ordinary first order differences $D_1(x_{i,j}, x_{i+1,j}) = \|x_{i,j} - x_{i+1,j}\|$ is an appropriate choice. The first order TV term incorporates horizontal, vertical and both diagonal differences. The diagonals are incorporated to reduce unwanted anisotropy effects. We note that $J'(x) = F(x; f) + (\alpha_1, \alpha_2, 0, 0) \text{TV}_1(x)$ is just the vector version of the anisotropic discrete ROF functional above. Using the notation

$$D_2(x, y, z) = \|x - 2y + z\| \quad \text{and} \quad D_{1,1}(x, y, u, v) = \|x - y - u + v\|,$$

(3)

for a norm of the standard second order differences for vector space data, the second order difference component, consisting of a horizontal and vertical component $\beta \text{TV}_2(x)$ as well as a diagonal component $\gamma \text{TV}_{1,1}(x)$, is given by

$$\beta \text{TV}_2(x) = \beta_1 \sum_{i=1, j=2}^{N-1,M} D_2(x_{i-1,j}, x_{i,j}, x_{i,j+1}) + \beta_2 \sum_{i=2, j=1}^{N,M-1} D_2(x_{i,j-1}, x_{i,j}, x_{i,j+1}),$$

$$\gamma \text{TV}_{1,1}(x) = \gamma \sum_{i,j=1}^{N-1,M-1} D_{1,1}(x_{i,j}, x_{i+1,j}, x_{i,j+1}, x_{i+1,j+1}).$$

(4)

(5)

The model parameters $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \beta_1, \beta_2, \gamma$ regulate the influence of the different TV terms.

Next, we consider suitable modifications of the above functional to obtain models for the inpainting problem with noisy and noiseless data. We start by first formulating the inpainting problem.

**Inpainting problem.** Given an image domain $\Omega_0 = \{1, \ldots, N\} \times \{1, \ldots, M\}$, an inpainting region $\Omega \subset \Omega_0$ is a subset of the image domain $\Omega_0$, where the pixel values $f_{i,j}, (i, j) \in \Omega$, are lost. The noiseless or noisy inpainting problems now consist of finding a function $x$ defined on $\Omega_0$ from data $f$ given on the complement $\Omega_0^C$ of the inpainting region, such that $x$ is a suitable extension to $f$ onto $\Omega$ and for the second case additionally denoised.
Inpainting noiseless vector space data. To deal with the noiseless situation, we consider the following modification of the functional $J$ given by (1). Since the data is assumed to be noiseless, we add the constraint that the target variable agrees with the data on the complement of the inpainting region. Furthermore, the data term considers only those indices for which actually data are available. This eliminates the data term from the functional. More precisely, the second order variational inpainting problem considered in this paper reads for a vector space as

$$\arg \min_{x \in (\mathbb{S}^1)^{N \times M}} \alpha \text{TV}_1(x) + \beta \text{TV}_2(x) + \gamma \text{TV}_{1,1}(x),$$

subject to $x_{i,j} = f_{i,j}$ for all $(i,j) \in \Omega^C$. 

The TV terms $\text{TV}_1, \text{TV}_2, \text{TV}_{1,1}$ are defined by (2),(4) and (5) using the difference terms $D_1, D_2, D_{1,1}$ based on the Euclidean norm in the vector space. Due to the constraint they actually only act on those difference terms that affect an entry in the inpainting region.

Second Order TV formulation of the inpainting problem in the presence of noise. For the inpainting problem in presence of noise the requirement of equality on $\Omega^C$ is replaced by $x$ being a suitable, i.e., smooth approximation. In this case, we search for a minimizer of the following second order TV functional for cyclic data for inpainting:

$$J_{\Omega}(x) = F_{\Omega^C}(x;f) + \alpha \text{TV}_1(x) + \beta \text{TV}_2(x) + \gamma \text{TV}_{1,1}(x),$$

where for any subset $B \subset \Omega_0$ of the image domain, we define

$$F_B(x;f) := \sum_{(i,j) \in B} d(x_{i,j}, f_{i,j})^2.$$ 

This means we use a data term, that enforces similarity to the given data $f$ on $\Omega^C$ while applying a regularization based on (2),(4), and (5) for the whole image domain. Specifying the inpainting area as the empty set, we obtain the pure denoising problem (1).

2.2 Absolute differences for combined cyclic and vector space data

In order to implement the above variational inpainting problem (6) and the simultaneous inpainting and denoising problem (7) for combined cyclic and vector space data, we have to find suitable difference operators $D_1, D_2, D_{1,1}$ for data consisting of combined cyclic and linear space components. In order to do so, we first find suitable definitions for vectors of cyclic data. Then, we combine these definitions with those for the linear space case in a way suitable to the space of interest in this paper. We use the symbols $D_\bullet$ already introduced in (2) for the linear space case, also for the case of vectors of cyclic and combined data. We further unify the notation to $D(\cdot; w)$ for different weights $w$. This overload is employed to avoid additional notation and should not cause confusion since the space under consideration will be clear from the context.

We let $w = (w_j)_{j=1}^d \in \mathbb{R}^d, w \neq 0$, be a vector with $\sum_{j=1}^d w_j = 0$, and call such a vector $w$ a weight. Special cases are the binomial coefficients with alternating signs

$$b_d = \left((-1)^{j+d-1} \binom{d}{j-1}\right)_{j=1}^{d+1}.$$
Figure 1. The three points $x, y, z \in \mathbb{R}^2$ illustrate the multivariate finite difference $\Delta_2(x, y, z) = 2\|p - y\|_2$, i.e. how “near” they are to lying equally distributed on a line segment in the right order.

For a set of vectors $x^{(j)} \in \mathbb{R}^n$, $j = 1, \ldots, d$, we denote the matrix containing these as columns by $x := (x^{(1)}, \ldots, x^{(d)}) \in \mathbb{R}^{n \times d}$. The \textit{absolute finite difference} $D$ for the vectors $x^{(j)} \in \mathbb{R}^n$, $j = 1, \ldots, d$, with respect to the weight $w \in \mathbb{R}^d$ is defined by

$$D(x; w) = \|xw\|_2 = \left\| \sum_{j=1}^d w_j x^{(j)} \right\|_2.$$  

For $w = b_d$, we obtain the forward differences of order $d$ for the vectors $x^{(1)}, \ldots, x^{(d+1)}$, i.e.

$$D_d(x) := D(x; b_d) = \left\| \sum_{j=1}^{d+1} (-1)^{j+d-1} \binom{d}{j-1} x^{(j)} \right\|_2.$$  

Another useful weight used in this paper is $w = b_{1,1} := (-1, 1, 1, -1)$. We denote the corresponding finite absolute difference by $D_{1,1}(x) := D(x; b_{1,1})$.

\textbf{Example 2.1.} For three points $x, y, z \in \mathbb{R}^n$ and $w = b_2 = (1, -2, 1)^T$ the second order absolute difference is given by $D_2(x, y, z) = \|x - 2y + z\|_2$, cf. (3). This can be interpreted as measuring the distance from $y$ to the midpoint of the line segment connecting $x$ and $z$; more precisely, we have $D_2(x, y, z) = 2\|\frac{1}{2}(x + z) - y\|_2$. The situation is shown in Figure 1 for $n = 2$, but also illustrates the situation for general $n > 2$ in the plane defined by $x, y, z$. For $n = 1$ the situation simplifies to $y$ always lying on the line —though not necessarily the segment— connecting $x$ and $z$.

We consider the cyclic case next. Let $S^1$ denote the unit circle $S^1 := \{p_1^2 + p_2^2 = 1 : p = (p_1, p_2)^T \in \mathbb{R}^2\}$ endowed with the \textit{geodesic or arc length distance}

$$d_{S^1}(p, q) = \arccos \langle p, q \rangle, \quad p, q \in S^1.$$  

Given a base point $q \in S^1$, the \textit{exponential map} $\exp_q : \mathbb{R} \to S^1$ from the tangent space $T_qS^1 \simeq \mathbb{R}$ of $S^1$ at $q$ onto $S^1$ is defined by

$$\exp_q(x) = R_x q, \quad R_x := \begin{pmatrix} \cos x & -\sin x \\ \sin x & \cos x \end{pmatrix}.$$  

7
This map is $2\pi$-periodic, i.e., $\exp_q(x) = \exp_q((x)_{2\pi})$ for any $x \in \mathbb{R}$, where $(x)_{2\pi}$ is the unique point such that

$$x = 2\pi k + (x)_{2\pi} \quad \text{with} \quad (x)_{2\pi} \in [-\pi, \pi), k \in \mathbb{Z}. \quad (9)$$

If we fix $q$, we obtain a representation system of $\mathbb{S}^1$, i.e., $\exp_q$ is a bijective map where $\exp_q(0) = q$ and there is a unique $x \in [-\pi, \pi)$ for each $p \in \mathbb{S}^1$ such that $\exp_q(x) = p$. A vector $x \in [-\pi, \pi)^m$ represents a point $\mathbb{S}^1|^m$ by component-wise application and for a point $q \in (\mathbb{S}^1)^m$, the map $\exp_q : [-\pi, \pi)^m \rightarrow (\mathbb{S}^1)^m$, $\exp_q(x) := (\exp_q(x_1), \ldots, \exp_q(x_m))^T$, is bijective where the properties from above hold component-wise. A distance measure on $(\mathbb{S}^1)^m$ is given by

$$d_{(\mathbb{S}^1)^m}(p, q) = \|\arccos(p\cdot q_i)^m\|_2.$$

In the following, we will introduce higher order differences on $(\mathbb{S}^1)^m$ by generalizing the one-dimensional approach from [5]. Let $x^{(j)} \in [-\pi, \pi)^m$, $j = 1, \ldots, d$. Using the notation $x := (x^{(1)}, \ldots, x^{(d)}) \in [-\pi, \pi)^{d \times m}$, the absolute cyclic difference of $x^{(1)}, \ldots, x^{(d)}$ with respect to a weight $w \in \mathbb{R}^d \setminus \{0\}$ is defined as

$$D(x; w) := \min_{\alpha \in \mathbb{R}^m} D([x]_2 + \alpha, \ldots, x^{(d)} + \alpha)_2; w),$$

where $[x]_2$ is multivalued and its $i$th component $([x]_2)_i$ is given by

$$([x]_2)_i = \begin{cases} (x_i)_2, & \text{if } x_i^{(j)} \neq (2z + 1)\pi \text{ for some } z \in \mathbb{Z}, \\ \pm \pi & \text{else}. \end{cases} \quad (10)$$

This definition may seem a bit technical at first glance. However, it allows for two points $x^{(i)}$, $x^{(j)}$, $i, j \in I_d := \{1, \ldots, d\}$, having the same value $x^{(i)}_l = x^{(j)}_l$ in one component $l \in \{1, \ldots, m\}$ to be treated separately. In fact, we may choose any $q \in (\mathbb{S}^1)^m$ as a base point for our representation system, which shifts any set of points given with respect to $\exp_{q'}$ by a fixed value of $\alpha := \exp^{-1}_{q'}(q')$. When the shift by $\alpha \in \mathbb{R}^m$ is small enough, such that no component of $x$ is affected by the component-wise application of $[\cdot]_2\pi$, both representation systems yield the same value. Hence the minimum simplifies to

$$D(x; w) = \min_{k \in \mathbb{R}^m} D([x]_2 - x_k + \pi, \ldots, x^{(d)} - x_k + \pi)_2; w),$$

where $x_k := (x^{(j)}_k)_{j=1}^m$. This is illustrated in Figure 2 for three points $x, y, z \in (\mathbb{S}^1)^2$.

Finally, we come to the space of interest in this paper which is $\mathcal{X} := (\mathbb{S}^1)^m \times \mathbb{R}^n$. A vector $x = (x_i)_{i=1}^m \in \mathcal{X}$ consists of two parts: the phase-valued $x_\mathcal{S} := (x_i)_{i=1}^m \in (\mathbb{S}^1)^m$ and the real valued components $x_\mathcal{R} = (x_i)_{i=m+1}^m \in \mathbb{R}^n$ of $x \in \mathcal{X}$. The distance of two points $x, y \in \mathcal{X}$ on this product space is given by

$$d_{\mathcal{X}}(x, y) = \sqrt{\|x_\mathcal{R} - y_\mathcal{R}\|_2^2 + d_{(\mathbb{S}^1)^m}(x_\mathcal{S}, y_\mathcal{S})^2}.$$

For a set of points $x^{(1)}, \ldots, x^{(d)} \in \mathcal{X}$, using the notation $x = (x^{(1)}, \ldots, x^{(d)})$ as before, the finite difference for cyclic and noncyclic data with respect to a weight $w \in \mathbb{R}^d \setminus \{0\}$ is defined by

$$D(x; w) := \sqrt{D(x_\mathcal{R}; w)^2 + D(x_\mathcal{S}; w)^2}.$$
Different shifts, where the first two, $F_s, F'_s$, yield the same constellation of $x, y, z$. Several shifts yield the same value $\Delta_2$; (b) cases yielding different values for the minimum occur for $x, y, z$ at the borders of the representation system.

Figure 2. For given points $x, y, z \in [-\pi, \pi]^2$ in a representation system $F_s$, i.e., with base point $s$, we have (a) shifts by arbitrary $\alpha \in \mathbb{R}^2$, e.g., to $s' = s + \alpha$. Several shifts yield the same value $\Delta_2$; (b) cases yielding different values for the minimum occur for $x, y, z$ at the borders of the representation system.

We further introduce the short hand notations

$$D_d(x) = D(x, b_d), \quad x \in \mathcal{X}^{d+1}, \quad d \in \mathbb{N},$$

(11)

to denote the corresponding absolute finite differences of order $d$. In particular, we have again $D_{1,1}(x) := D(x; b_{1,1})$ for $x \in \mathcal{X}^d$, with $b_{1,1} = (-1, 1, 1, -1)^T$. For the weights corresponding to first and second order differences, we have a particularly nice representation, which is given by the following Lemma.

Lemma 2.2. For $w \in \{b_1, b_2, b_{1,1}\}$ and $x \in \mathcal{X}^d$ where $d$ denotes the length of $w$, we have

$$D(x, w)^2 = \| (xS w)_{2\pi} \|_2^2 + \| (xR w) \|_2^2,$$

(12)

Proof. For the real-valued components there is nothing to show. Hence we may restrict to $n = 0$ (which corresponds to purely cyclic data) and thus have to show that $D(x, w) = \| (xw)_{2\pi} \|_2$ for $x \in (S^1)^{n \times d}$. To this end we apply Proposition 2.5 of [5] to each row $x$ and conclude the validity of (12). 

2.3 Inpainting combined cyclic and vector space data

We can now apply the definition of absolute differences for combined cyclic and vector space data we derived in Section 2.2 to obtain variational models for inpainting and simultaneous inpainting and denoising. This extends the models for the Euclidean differences in Subsection 2.1 using a first order TV term (2) as well as to the second order TV terms (4) and (5).
to a more general setting. The noiseless inpainting model now reads

$$\arg \min_{x \in \mathcal{X}^{N,M}} \alpha \text{TV}_1(x) + \beta \text{TV}_2(x) + \gamma \text{TV}_{1,1}(x),$$

subject to $x_{i,j} = f_{i,j}$ for all $(i,j) \in \Omega^C$.

Here, $\text{TV}_1$ is defined by (2) incorporating the first order absolute cyclic differences $D_1$ given by (11) and the second order TV terms $\text{TV}_2, \text{TV}_{1,1}$ are defined by (4) and (5) employing the second order absolute cyclic differences $D_2, D_{1,1}$ given by (11), respectively.

Proceeding similarly, we get a variational formulation of the inpainting problem for noisy combined cyclic and vector space data by computing a minimizer of

$$J_\Omega(x) = F_{\Omega^C}(x; f) + \alpha \text{TV}_1(x) + \beta \text{TV}_2(x) + \gamma \text{TV}_{1,1}(x).$$

Here the data term is given by (8) using the distance function on $\mathcal{X} = (\mathbb{S}^1)^m \times \mathbb{R}^n$. As in the noiseless situation, $\text{TV}_1$ is defined by (2) again incorporating the first order absolute cyclic differences $D_1$ and the second order TV terms $\text{TV}_2, \text{TV}_{1,1}$ are defined by (4) and (5) employing the second order absolute cyclic differences $D_2, D_{1,1}$ from (11), respectively.

3 Algorithms for inpainting and denoising combined cyclic and linear space data

In the following, we derive algorithms to solve the inpainting problem (6) and the combined inpainting and denoising problem (7) for combined cyclic and vector space data, cf. (13) and (14). Note that the latter includes the denoising of combined cyclic and vector space data for the case of $\Omega = \emptyset$, an empty inpainting set. These algorithms are based on a cyclic proximal point algorithm whose concept we recall in Section 3.1. We derive explicit formulas for the proximal mappings needed for inpainting and denoising of such combined data in Section 3.2. Using these explicit representations, we derive a cyclic proximal point algorithm for inpainting noiseless combined cyclic and vector space data and similarly for simultaneously inpainting and denoising data in Section 3.3. This also includes an efficient choice for the cycles involved. Finally in Section 3.4, we prove convergence of our algorithm to a minimizer under certain conditions, that reflect the space-inherent non-convexity of the involved functionals.

3.1 The cyclic proximal point algorithm

For closed, convex and proper functional $\varphi: \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ the proximal mapping is given by

$$\text{prox}_{\lambda \varphi}(f) := \arg \min_x \frac{1}{2} \|f - x\|^2 + \lambda \varphi(x),$$

where $\lambda > 0$ is a tradeoff or regularization parameter. The fixed points of $\text{prox}_{\lambda \varphi}(f)$ are minimizers of $\varphi$. Hence, if the proximal mapping $\text{prox}_{\lambda \varphi}(f)$ can be computed in closed form, an algorithm for finding a minimizer is given by iterating

$$x^{(k)} = \text{prox}_{\lambda \varphi}(x^{(k-1)}), \quad k = 1, 2, \ldots$$

10
for some starting value $x^{(0)}$. This algorithm is called *proximal point algorithm* (PPA) and was introduced by Rockafellar [64]. It was recently extended to Riemannian manifolds of non-positive sectional curvature [34] and also to Hadamard spaces [1]. If the function $\varphi$ can be split into simpler parts, i.e. $\varphi = \sum_{i=1}^{c} \varphi_i$, for which then individually the proximal mappings are known in closed form, a similar algorithm is given for a sequence $\{\lambda_k\}_k$ of regularization parameters by

$$x^{(k+\frac{1}{c})} = \text{prox}_{\lambda_k \varphi_i}(x^{(k+\frac{1}{c-1})}), \quad l = 1, \ldots, c, \quad k = 1, 2, \ldots,$$

and it is called *cyclic proximal point algorithm* (CPPA). Its formulation on Euclidean space is derived in [8], see also the survey [7]. It converges to a minimizer of $\varphi$ if

$$\sum_{k=0}^{\infty} \lambda_k = \infty, \quad \text{and} \quad \sum_{k=0}^{\infty} \lambda_k^2 < \infty. \quad (15)$$

The concept of CPPAs for Hadamard spaces has been treated in in [2]. A CPPA for TV minimization for manifolds and in Hadamard spaces has been derived in [81]. For second order TV type problems, a CPPA to denoise $\mathbb{S}^1$ data was derived in [5]. A preliminary model, different to the one appearing in this paper, was applied to inpainting of $\mathbb{S}^1$ data in [6]. For manifold data in general, the main challenge is to derive proximal mappings which are as explicit as possible.

### 3.2 Proximal mappings for inpainting

In this part, we derive closed form expressions for the proximal mappings needed to make the cyclic proximal point algorithm from Section 3.1 work for the problems (13) and (14). To this end, we need a slight modification of Lemma 3.1 of [5] which deals with vector space data. The proof of this lemma can be given using the method employed in [5]. We present a slightly more geometric variant here. We first derive explicit expressions for the proximal mappings of functions of the form

$$\varphi(x) = \|xw - a\|_2, \quad a \in \mathbb{R}^n,$$

where the target variable $x$ is a matrix in $\mathbb{R}^{n \times d}$, where $d$ corresponds to the length of $w$. The vector $a$ introduces an offset. We employ the notation $\|y\|_F = \sqrt{\sum_{i,j=1}^{m,d} (y_{ij})^2}$ to denote the Frobenius norm of a matrix $y$.

**Lemma 3.1.** Let $f = (f^{(1)}, \ldots, f^{(d)}) \in \mathbb{R}^{n \times d}$ be a matrix whose columns $f^{(i)}$ represent the data items, let $0 \neq w \in \mathbb{R}^d$ ($w$ needs not necessarily to be a weight), and $\lambda > 0$ be given. For the functional

$$E(x; f, a, w) = \frac{1}{2} \|f - x\|_F^2 + \lambda \|xw - a\|_2, \quad (16)$$

with target variable $x \in \mathbb{R}^{n \times d}$, the minimizer $\hat{x}$ is given by

$$\hat{x} = f - smw^T, \quad (17)$$
where direction \( s := \begin{cases} \frac{f w - a}{\|fw - a\|_2} & \text{if } \|fw - a\|_2 \neq 0, \\ 0 & \text{else}, \end{cases} \) and step size \( m := \min\{\lambda, \frac{\|fw - a\|_2}{\|w\|_2} \} \). The minimum \( E(\hat{x}; f, a, w) \) is given by

\[
E(\hat{x}; f, a, w) = \begin{cases} \frac{1}{2} \|fw - a\|_2^2 + \frac{1}{2} \lambda^2 \|w\|_2^2 \quad & \text{if } m \leq \lambda, \\ \|w\|_2^2 \left( \frac{1}{2} \lambda^2 + \lambda(\|f w - a\|_2 - \lambda) \right) & \text{otherwise}. \end{cases}
\]  

(18)

Furthermore, given data \( f, \tilde{f} \in \mathbb{R}^{n \times d} \), and different offsets \( a, \tilde{a} \in \mathbb{R}^n \), the following implication holds:

\[
\|fw - a\|_2 < \|\tilde{f}w - \tilde{a}\|_2 \implies \min_{x \in \mathbb{R}^{n \times d}} E(x; f, a, w) < \min_{x \in \mathbb{R}^{n \times d}} E(x; \tilde{f}, \tilde{a}, w).
\]  

(19)

**Proof.** We first reduce the functional to be minimized to an equivalent problem without offset. By assumption there is an index \( j \) such that \( w_j \neq 0 \), which allows us to write (16) as

\[
E(x; f, a, w) = \frac{1}{2} \|f - x\|^2_2 \lambda |w_j| \left( x - \frac{a}{w_j} e_j^T \right) (\frac{w}{w_j})_2.
\]

Defining the new target matrix \( y := x - \frac{a}{w_j} e_j^T \), the new data matrix \( g = f - \frac{a}{w_j} e_j^T \), the new regularizing parameter \( \nu := \lambda |w_j| \), and the new (not necessarily weight) vector \( v := \frac{w}{w_j} \), we obtain the new problem

\[
F(y; g, v) = \frac{1}{2} \|g - y\|^2_2 + \nu \|yv\|_2,
\]  

(20)

where the second term is free of an offset. Their relation between minimizers \( \hat{x} \) of \( E \) and \( \hat{y} \) of \( F \) is given via \( \hat{y} = \hat{x} - \frac{a}{w_j} e_j^T \).

We now consider the problem (20) and first show (17) for \( F \). The corresponding statement for \( E \) follows by carrying out the resubstitution. If \( \|gv\|_2 = 0 \) then we have \( F(g; g, v) = 0 \) and hence \( \hat{y} = g \) is the minimizer of (20). So we may assume \( \|gv\|_2 \neq 0 \) in the following. We now distinguish whether \( \|yv\|_2 \neq 0 \) or \( \|yv\|_2 = 0 \). In the first case, we may differentiate \( F \) and setting the gradient of \( F \) to zero results in

\[
0 = y - g + \frac{\nu}{\|yv\|_2} (yv)v^T.
\]

We sum up the columns of this matrix valued equation weighted by \( v \) to further obtain

\[
yv - gv = \sum_{j=1}^d (y^{(j)} - g^{(j)}) v_j = -\nu \|yv\|_2 \sum_{j=1}^d v_j (yv) = -\nu \|yv\|_2 \|v\|_2^2.
\]

Rearranging yields

\[
(1 + \nu \|v\|_2^2) yv = gv,
\]

which implies that \( \frac{yv}{\|yv\|_2} = g \frac{gv}{\|gv\|_2} \), i.e., both vectors have the same direction. This leads to

\[
y = g - \nu \frac{gv}{\|gv\|_2} v^T.
\]
For \( \|yv\|_2 = 0 \) we look at the subgradient of \( F \). As condition for a minimizer \( \hat{y} \), we have that \( \hat{y} - g \) is in the subgradient of \( \nu \|yv\|_2 \). For \( y \) with \( \|yv\|_2 = 0 \), this subgradient is given as \( \{ zv^T : \|z\|_2 \leq \nu \} \). When considering the functional \( F \), the amplitude \( m \) from the assertion of the lemma reads \( m = \min(\nu, \|yv\|_2/\|v\|_2) \). If \( \|yv\|_2/\|v\|_2 < \nu \), then \( F \) is differentiable at \( y = g - smv^T \), \( \|y\| \neq 0 \), and we are in the previously considered case. Hence, we may assume that \( m = \nu \). Then, for \( \hat{y} = g - smv^T \), we have \( \hat{y} - g = smv^T \) with \( m = \nu \). This shows that \( \hat{y} \) fulfills the condition of a minimizer. In consequence, (17) is true for the functional \( F \). Then resubstituting shows (17) for \( E \), and plugging \( \hat{x} \) into \( E \) we get (18).

It remains to show the implication (19). To this end, let \( \mu = f_{\|w-a\|_2} \) and \( \tilde{\mu} := \frac{f_{\|w-a\|_2}}{\|w-a\|_2} \). By the assumption of (19), \( \|\tilde{\mu}\|_2 < \|\tilde{\mu}\|_2 \). We consider three cases. If \( \|\tilde{\mu}\|_2 < \lambda \), then \( \|\tilde{\mu}\|_2 < \lambda \). Hence, the minimizer of \( E(x; f, a, w) \) equals \( \frac{1}{2}\|w\|_2^2 \|\tilde{\mu}\|_2^2 \), and the one of \( E(x; \tilde{f}, \tilde{a}, w) \) equals \( \frac{1}{2}\|w\|_2^2 \|\tilde{\mu}\|_2^2 \geq \frac{1}{2}\|w\|_2^2 \|\tilde{\mu}\|_2^2 \) which shows (19). If \( \|\tilde{\mu}\|_2 > \lambda \) and \( \|\tilde{\mu}\|_2 > \lambda \), we have to consider the second line of (18) for \( \|\tilde{\mu}\|_2 \) which gives the minimal value of \( E(x; \tilde{f}, \tilde{a}, w) \). We have to show that
\[
\|w\|_2^2 \left( \frac{1}{2} \lambda + (\|\tilde{\mu}\|_2 - \lambda) \right) > \frac{1}{2}\|w\|_2^2 \|\tilde{\mu}\|_2^2;
\]
but this is a consequence of the second summand on the left hand side being positive and \( \lambda \geq \|\mu\| \). This shows (19) for this case. Finally, if both \( \|\tilde{\mu}\|_2 > \lambda \) and \( \|\tilde{\mu}\|_2 > \lambda \), we apply the second line of (18) and see that we need \( \|\tilde{\mu}\|_2 - \lambda > \|\tilde{\mu}\|_2 - \lambda \) for the statement to hold. This is true by assumption which completes the proof.

**Example 3.2.** We continue the situation from Example 2.1 and take three points \( f^{(j)} \in \mathbb{R}^2 \), \( j = 1, 2, 3 \) and denote \( f = (f^{(1)}, f^{(2)}, f^{(3)}) \). Depending on the chosen value for \( \lambda \) in the proximal mapping, there are two possibilities: If \( m = \frac{\|f_{\|w\|_2}\|_2}{\|w\|_2^2} \leq \lambda \), we obtain three points \( x = \text{prox}_{\lambda D_2}(f) = f - smb_2^T \) that lie on a line, cf. Figure 3(a). If \( m > \lambda \), then the result \( x \) of the proximal mapping does not yield \( \Delta_2(x) = 0 \), but the ‘movement’ of the points in direction \( s \) is restricted by \( \lambda b_2^T \), cf. Figure 3(b).

After these preparations, we now deal with the proximal mappings needed for the inpainting problems (13) and (14) for combined cyclic and vector space data. In particular, each data item now is an element of \( \mathcal{X} = (\mathbb{S}^1)^m \times \mathbb{R}^n \). As motivation, let us first have a look at the first order difference \( D_1 \) and the inpainting problem (13) for noiseless data. By the constraint \( x_{ij} = f_{ij} \) outside the inpainting region, it might happen that at the boundary of the inpainting region the member \( x_{ij} = f_{ij} \) is fixed but its neigbor, say \( x_{i,j+1} \), may vary. Then, we have to study the corresponding functional \( D_1(x_{ij}, x_{i,j+1}) \) for fixed \( x_{ij} \) and find its proximal mapping. The following theorem deals with this issue in a more general setup.

Before we state the theorem we introduce some notation: For \( x = (x_S, x_R)^T \in \mathcal{X}^d \) we define \((\cdot)^\mathcal{X} : \mathbb{R}^{(m+n)\times d} \to \mathcal{X}^d \) by
\[
(x)_{\mathcal{X}} := ((x_S)_{2\pi}, x_R)^T,
\]
where \((\cdot)_{2\pi}\) defined by (9) is applied to the \( m \) cyclic components of each column vector \( x^{(i)} \in \mathcal{X} \). Similarly we define
\[
[x]_{\mathcal{X}} := ([x_S]_{2\pi}, x_R)^T,
\]
where \([\cdot]_{2\pi}\) defined in (10) is applied to the phase-valued components of \( x \).
In the following we consider a weight vector \( w \in \mathbb{R}^d \), a data matrix \( x = (x_1, \ldots, x_d) \) with each member \( x^{(i)} \) having values in \( X = (S^1)^m \times \mathbb{R}^n \) and a subset \( A \subset \{1, \ldots, d\} \). We partition \( w \) into a variable part \( w^a \) and into a fixed part \( \tilde{w} \) according to whether the index \( i \) of \( w^{(i)} \) belongs to \( A \) or not. Accordingly, we partition \( x \) into a variable part \( x^a \) and into a fixed part \( \tilde{x} \) and consider the mappings

\[
\varphi_A: x^a \mapsto D(x, w)
\]

for the corresponding differences \( D(x, w) \), where only the \( x^a \) are considered as variable and the \( \tilde{x} \) are fixed values. We derive an explicit representation for the corresponding proximal mappings in the following theorem.

**Theorem 3.3.** Let \( w \) be one of the weights \( w = (-1, 1), w = (1, -2, 1), \) or \( w = (-1, 1, 1, -1) \) which corresponds to considering the first order difference \( D_1 \) and the second order differences \( D_2 \) and \( D_{1,1} \), respectively. Let \( d \) be the respective length of \( w \), \( A \subset \{1, \ldots, d\} \), and \( w \) be partitioned into the corresponding variable part \( w^a \) and into a fixed part \( \tilde{w} \). We partition \( x, f \in X^d \) accordingly and let \( \tilde{f} = \tilde{x} \). Then, the proximal mapping of \( \varphi_A \) defined in (21) is given by

\[
\text{prox}_{\lambda \varphi_A}(f^a) = (f^a - s m w^a)_X,
\]

with the parameter \( \lambda > 0 \); here, the direction(s) \( s \) and amplitude \( m \) are given by

\[
s = \begin{cases} 
\frac{|f^a|_X}{\|f^a\|_2} & \text{if } \|f^a\|_2 \neq 0, \\
0 & \text{else,}
\end{cases} \quad \text{and} \quad m = \min \left\{ \lambda, \frac{\|f^a\|_2}{\|w^a\|_2} \right\}. \tag{22}
\]

**Remark.** We note that the bracket \( [\cdot]_X \), and thus the proximal mapping (having an additional value for each additional instance of \( s \)), is multivalued if some components of \( (f^a w)_a \) are equal to \(-\pi\). More precisely, if there are \( l \in \{1, \ldots, n\} \) such components, we obtain \( 2^l \) solutions from
the different instances the vector $s$ might take. The reason for this is, that the mapping $\varphi_A$ is no longer convex for data in $X$, and that the minimizer defining the proximal mapping is not necessarily unique. Owing to this observation, we consider set valued proximal mappings gathering all minimizers. We notice that the above proximal mapping is single-valued if and only if $(f_{3\sigma}w, 2\pi) \in (-\pi, \pi)^d$. This is the generic case. The degenerate case involving antipodal points appears rather seldom in practice; at least, in a non-artificial noisy setup, it is very unlikely to encounter antipodal points. Furthermore, data with antipodal points or almost antipodal points may often be interpreted as not fine enough sampled data. This means, if the sampling rate is higher, the distance of nearby data items gets smaller and the situation just disappears.

Proof. In order to derive explicit formulae for the proximal mappings, we have to find the minimizer(s) of

$$E_X(x^a, f^a, w) := \frac{1}{2} \sum_{j \in A} d_X(f^{(j)}, x^{(j)})^2 + \lambda D(x; w).$$

By Lemma 2.2 we may rewrite $E_X(x^a, f^a, w)$ as

$$E_X(x^a, f^a, w) = \frac{1}{2} \sum_{j \in A} \|f_R^{(j)} - x_R^{(j)}\|_2^2 + \frac{1}{2} \sum_{j \in A} \min_{k, \sigma \in \mathbb{Z}^n} \|f_{3\sigma}^{(j)} - x_{3\sigma}^{(j)} - 2\pi k\|_2^2$$

$$+ \min_{\sigma \in \mathbb{Z}^n} \lambda \sqrt{\|x_{3\sigma} w\|_2^2 + \|x_{3\sigma} w - 2\pi \sigma\|_2^2}.$$

We now use that $\hat{x} = \tilde{f}$ and employ the notation $\|\cdot\|_F$ for the Frobenius norm. For the remaining part of the proof, let $p = |A|$. We obtain that

$$E_X(x^a, f^a, w) = \min_{k \in \mathbb{Z}^m \times \mathbb{R}^n} \left( \frac{1}{2} \|f_R^a - x_R^a\|_F^2 + \frac{1}{2} \|f_{3\sigma}^a - x_{3\sigma}^a - 2\pi k\|_F^2 + \right.$$

$$\left. \min_{\sigma \in \mathbb{Z}^n} \lambda \sqrt{\|x_{3\sigma}^a w^a + \tilde{f}_R \tilde{w}\|_2^2 + \|x_{3\sigma}^a w^a - (2\pi \sigma - \tilde{f}_R \tilde{w})\|_2^2} \right).$$

Using the notation $E_{k, \sigma}$ to denote the corresponding right hand side of (23), our minimization problem reads

$$\min_{x^a \in [-\pi, \pi)^{m \times \mathbb{R}^n}} E_X(x^a, f^a, w) = \min_{k \in \mathbb{Z}^m \times \mathbb{R}^n} \min_{x^a \in [-\pi, \pi)^{m \times \mathbb{R}^n}} E_{k, \sigma}(x^a, f, w)$$

$$= \min_{k \in \mathbb{Z}^m \times \mathbb{R}^n} \min_{\sigma \in \mathbb{Z}^n} E_{k, \sigma}(x^a, f, w),$$

where

$$E_{k, \sigma}(x^a, f, w) = \frac{1}{2} \|f_R^a - x_R^a\|_F^2 + \frac{1}{2} \|f_{3\sigma}^a - x_{3\sigma}^a - 2\pi k\|_F^2$$

$$+ \lambda \sqrt{\|x_{3\sigma}^a w^a + \tilde{f}_R \tilde{w}\|_2^2 + \|x_{3\sigma}^a w^a - (2\pi \sigma - \tilde{f}_R \tilde{w})\|_2^2}.$$

The last identity in (24) is a consequence of the fact that the $E_{k, \sigma}$-values of the additional candidates are already considered. More precisely, if the minimizer $\hat{x}$ of $E_{k, \sigma}$ has $S^4$ components
\( \hat{x}^{(j)} \), i.e. \( i \leq m \) with \( \hat{x}^{(j)} = \pi \), we define \( \hat{x} \) simply by letting \( \hat{x}^{(j)} = -\pi \), for all such \( i, j \). Having a look at (25), there are \( \hat{k}, \hat{\sigma} \) such that \( E_{k, \sigma}(\hat{x}) = E_{k, \hat{\sigma}}(\hat{x}) \). Summing up, the problem reduces to finding the minimizers of all \( E_{k, \sigma} \) in \([-\pi, \pi]^{m \times p}\) and comparing their value. For the remaining part of the proof, let \( 0_n \) be a zero (column) vector of length \( n \) and \( 0_{n,p} \) be a zero-matrix of dimension \( n \times p \).

For any \( k, \sigma \), the functional \( E_{k, \sigma} \) has a unique minimizer given by Lemma 3.1 as

\[
\hat{x}_{k, \sigma} = f^a - 2\pi \left( \begin{array}{c} k \\ 0_{n,p} \end{array} \right) - sk, \sigma m_{k, \sigma}(w^a)^T.
\]  

(26)

We first derive \( s_{k, \sigma} \) using Lemma 3.1, where we use the notation,

\[
s_{k, \sigma} = \nu_{k, \sigma}/\|\nu_{k, \sigma}\|_2.
\]

We notice that the data for Lemma 3.1 is given by \( f^a - 2\pi \left( \begin{array}{c} k \\ 0_{n,p} \end{array} \right) \) and the offset \( a \) in the same lemma is \( a = 2\pi \left( \begin{array}{c} \hat{\sigma} \\ 0_n \end{array} \right) - \hat{\tilde{w}} \). We get

\[
\nu_{k, \sigma} = \left( f^a - 2\pi \left( \begin{array}{c} k \\ 0_{n,p} \end{array} \right) \right) w^a + \hat{\tilde{w}} - 2\pi \left( \begin{array}{c} \hat{\sigma} \\ 0_n \end{array} \right)
\]

\[
= f w - 2\pi \left( \begin{array}{c} k \\ 0_{n,p} \end{array} \right) w^a + \left( \begin{array}{c} \hat{\sigma} \\ 0_n \end{array} \right),
\]

and

\[
m_{k, \sigma} = \min \left\{ \lambda \nu_{k, \sigma}/\|w^a\|_2 \right\}.
\]

By (19), we have to find the minimum of the norms \( \|\nu_{k, \sigma}\|_2 \) with respect to \( k, \sigma \) in order to find the minimum or minima and their corresponding minimizers of the \( E_{k, \sigma} \). We first consider the following special case, where

\[
(fw)_i \notin 2\pi\mathbb{Z} - \pi \text{ for all } i \in \{1, \ldots, m\}.
\]

(28)

More precisely, in none of the cyclic data dimensions \( i = 1, \ldots, m \) of \( f \in \mathcal{X}^d \), the scalar product with the weight \( (fw)_i \) is an odd multiple of \( \pi \). There exist \( r_1, \ldots, r_m \) such that \( (fw)_i - 2\pi r_i \in (-\pi, \pi) \) for all \( i \in \{1, \ldots, m\} \). Then \( \|\nu_{k, \sigma}\|_2 \) is minimal with respect to \( k, \sigma \) if and only if \( kw^a + \sigma = r \) where \( r = (r_1, \ldots, r_m)^T \), cf (27). For each fixed matrix \( \hat{k} \in \mathbb{Z}^{m \times p} \) there is a uniquely determined vector \( \hat{\sigma} = \hat{\sigma}(\hat{k}) \) solving this system of linear equations. Such a pair minimizes \( \|\nu_{k, \sigma}\|_2 \) w.r.t. \( k, \sigma \) and

\[
\nu_{k, \hat{\sigma}} = (fw)^k.
\]

Using (26) we obtain the corresponding minimizer w.r.t. \( x \) as

\[
\hat{x}^{a}_{k, \hat{\sigma}} = f^a - 2\pi \left( \begin{array}{c} k \\ 0_{n,p} \end{array} \right) - sm(w^a)^T,
\]

with \( s, m \) as given in (22). Now there is precisely one \( k^* \) with its corresponding \( \hat{\sigma} = \hat{\sigma}(k^*) \) such that \( \hat{x}^{a}_{k^*, \hat{\sigma}} \in [-\pi, \pi]^{m \times p} \times \mathbb{R}^{n \times p} \) which implies that

\[
x^* = \hat{x}^{a}_{k^*, \hat{\sigma}} = (f^a - sm(w^a)^T)^k.
\]

16
Proposition 3.5. For $f, g \in X^N$ let

$$\mathcal{E}(x; g, f) = \sum_{i=1}^{N} d_X(g^{(i)}, x^{(i)})^2 + \lambda d_X(f^{(i)}, x^{(i)})^2.$$
Figure 4. Minimizing a second order difference in $\mathbb{R}^2$, where $f^{(1)}$ is fixed, i.e. $x^{(1)} = f^{(1)}$ and $A = \{2, 3\}$ are the active data points. Again, to different values of $\lambda$ are shown:

(a) $\lambda > D_2(f; w)$, i.e. the corresponding proximal mapping reaches the minimum,
(b) $\lambda < D_2(f; w)$, i.e. the corresponding proximal mapping is just a step reducing the value $D_2(x; w) < D_2(f; w)$.

Then, the minimizer(s) $\hat{x}$ of $E$ are given by

$$\hat{x} = \left( g + \frac{\lambda f}{1 + \lambda} + \frac{\lambda}{1 + \lambda} 2\pi v \right) \chi,$$

where $v = (v^{(i)}_j)_{i,j=1}^{N,n+m} \in \mathcal{X}^N$ is defined by

$$v^{(i)}_j = \begin{cases} 0 & \text{if } j > m, \\ 0 & \text{if } |g^{(i)}_j - f^{(i)}_j| < \pi, j \leq m, \\ \text{sgn}(g^{(i)}_j - f^{(i)}_j) & \text{if } |g^{(i)}_j - f^{(i)}_j| > \pi, j \leq m. \end{cases}$$

Proof. We observe that the functional $E$ under consideration (which only involves squared distances) does neither couple the cyclic and vectorial components nor within the components. Hence, the proposition follows from considering both the $\mathbb{R}$-valued data and the $S^1$-valued data case separately. This has been done in [5] in Lemma 3.3 and Proposition 3.7, respectively.

3.3 Cyclic proximal point algorithms

The proximal mappings from Theorem 3.3 can be efficiently applied in parallel to compute minimizers of (13) and (14) using a cycle length in the proximal point algorithm from Section 3.1 by splitting the functionals accordingly.

A splitting for noiseless inpainting. Each summand in the first and second order differences in (13) can be incorporated into a proximal mapping using Theorem 3.3 by setting all values affected by the subjection as fixed and only keep the remaining ones as active.
If two summands act on distinct data, their proximal mappings can be computed in parallel. This reduces the cycle length $c$ of the CPPA tremendously and provides an efficient, parallel implementation. In the following, we will split each of the summands

$$J(x) = \alpha \text{TV}_1(x) + \beta \text{TV}_2(x) + \gamma \text{TV}_{1,1}(x)$$

into

$$J = \sum_{i=1}^{18} J_i$$

with summands $J_i$ given by the subsequent explanation. We start with the $\alpha \text{TV}_1^\Omega(x)$ term and first consider the horizontal summand $\alpha_1 \sum_{(i,j)} d(x_{i,j}, x_{i+1,j})$. We split this sum into even and odd part $J_1$ and $J_2$, more precisely

$$\alpha_1 \sum_{(i,j)} D_1(x_{i,j}, x_{i+1,j}) = J_1 + J_2,$$

where

$$J_1 + J_2 := \alpha_1 \sum_{(i,j)} D_1(x_{2i,j}, x_{2i+1,j}) + \alpha_1 \sum_{(i,j)} D_1(x_{2i+1,j}, x_{2i+2,j}),$$

where for each summand individually the indices are set as active w.r.t. Theorem 3.3 if they are not fixed by the constraint, i.e. if $(i, j) \in \Omega^C$. Furthermore, the summation can be restricted to those differences containing active data items, i.e. first order differences that completely lie in $\Omega^C$ can be excluded. For the vertical as well as for the diagonal summands in $\alpha \text{TV}_1(x)$ we proceed analogously to obtain the splitting functionals $J_3, \ldots, J_8$.

Next, we consider the $\beta \text{TV}_2(x)$ term. We first look at its first (horizontal) summand, which is given by $\beta_1 \sum_{(i,j)} D_2(x_{i-1,j}, x_{i,j}, x_{i+1,j})$. We decompose this summand into three sums $J_9$, $J_{10}$, $J_{11}$, given by

$$J_9 = \beta_1 \sum_{(i,j)} D_2(x_{3i-1,j}, x_{3i,j}, x_{3i+1,j}),$$

$$J_{10} = \beta_1 \sum_{(i,j)} D_2(x_{3i,j}, x_{3i+1,j}, x_{3i+2,j}),$$

$$J_{11} = \beta_1 \sum_{(i,j)} D_2(x_{3i+1,j}, x_{3i+2,j}, x_{3i+3,j}),$$

again, with the partition of the values in $x$ with respect to $\Omega$ and $\Omega^C$. For the vertical summand in $\beta \text{TV}_2(x)$ we proceed analogously to obtain $J_{12}, \ldots, J_{14}$.
It remains to split the term $\gamma\TV_{1,1}(x)$ into four functionals $J_{15}, \ldots, J_{18}$ as follows:

$$J_{15} = \gamma \sum_{(i,j)} D_{1,1}(x_{2i,2j}, x_{2i+1,2j}, x_{2i,2j+1}, x_{2i+1,2j+1}),$$

$$J_{16} = \gamma \sum_{(i,j)} D_{1,1}(x_{2i+1,2j}, x_{2i+2,2j}, x_{2i+1,2j+1}, x_{2i+2,2j+1}),$$

$$J_{17} = \gamma \sum_{(i,j)} D_{1,1}(x_{2i,2j+1}, x_{2i+1,2j+1}, x_{2i,2j+2}, x_{2i+1,2j+2}),$$

$$J_{18} = \gamma \sum_{(i,j)} D_{1,1}(x_{2i+1,2j+1}, x_{2i+2,2j+1}, x_{2i+1,2j+2}, x_{2i+2,2j+2}).$$

The splitting consists of all summands from (13), where inside one function $J_l$, $l \in \{1, \ldots, 18\}$, any data point $x_{i,j} \in \mathcal{X}$ is occurring at most once and such a functional can be evaluated in parallel. This leads to a cycle length of $c = 18$.

**A splitting for combined inpainting and denoising.** In order to derive a cyclic proximal point algorithm for the combined inpainting and denoising model (13), we encounter two differences compared to the previous derivation: All data is always marked as active because no index $(i, j) \in \Omega_0$ is restricted by a constraint, i.e., the set $A$ from Theorem 3.3 is always $A = \{1, \ldots, d\}$, where $d$ depends on the summand being a first, second or second order mixed absolute difference. For these summands, the same splitting into functionals $J_{15}, \ldots, J_{18}$ as in the noiseless model (13) is used and the same explanation for unknown data applies as for the noiseless case. In the additional functional $F_{\Omega^C}(x; f)$, similar to (7), each of the summands acts on distinct data with respect to $x$. Hence the $|\Omega^C|$ many summands can be combined into

$$J_{19} = \sum_{(i,j) \in \Omega^C} d_{\mathcal{X}}(x_{i,j}, f_{i,j})^2$$

and evaluated in parallel using the proximal mapping given by Proposition 3.5.

**Initialization.** In order to initialize the algorithm, we also employ the CPPA to derive initial values for the inpainting area. All data $x_{i,j}$, $(i, j) \in \Omega$ are initialized as *unknown* and the values $x_{i,j} = f_{i,j}$, $(i, j) \in \Omega^C$ is set to be *inactive* for the inpainting, where no noise is present. During the proximal mappings, some might act on data involving both active and inactive items, which can be computed using Theorem 3.3. Furthermore a finite difference term might contain one or more unknown values. If it consists of only one unknown data item $\hat{x} \in \mathcal{X}$, the value can be obtained by solving a system of linear equations when setting the corresponding difference $D(x, b) = 0, x \in \mathcal{X}^d$. Afterwards, this data item is set to be known. Otherwise, i.e. if more than one involved data item is unknown, they are kept unchanged. Using this procedure, at least all data items being adjacent to known pixels are initialized. Hence after at most $k = \max\{N, M\}$ iterations, all pixels are known.

The complete procedure for both models of noiseless and noisy inpainting is summarized in Algorithm 1.
Algorithm 1  CPPA for minimizing (1) for \((S^1)^n \times R^m\) valued data

**Input** a sequence \(\{\lambda_k\}_k\) of positive values, cf. (15)
parameters \(\alpha = (\alpha_1, \alpha_2, \alpha_3, \alpha_4)\), \(\beta = (\beta_1, \beta_2)\), \(\gamma\),
a set \(\Omega \subset \Omega_0\), and data \(f \in X^N \times M\)

**function** CPPA\((\alpha, \beta, \gamma, \{\lambda_k\}_k, f)\)
- Initialize \(x^{(0)}_{ij} = f_{ij}, (i,j) \in \Omega^C\), \(x^{(0)}_{ij}\) as active, as unknown for \((i,j) \in \Omega\) and \(k = 0\)
- Initialize the cycle length as \(c = 18\) (noiseless case) or \(c = 19\) (noisy case)
**repeat**
  - for \(l \leftarrow 1\) to \(c\) do
    - \(x^{(k+\frac{l}{c})}_{ij} \leftarrow \text{prox}_{\lambda \phi^l}(x^{(k+\frac{l-1}{c})}_{ij})\) 
      employing the proximal mappings from Theorem 3.3 and Proposition 3.5
  - \(k \leftarrow k + 1\)
**until** a convergence criterion are reached
**return** \(x^{(k)}\)

### 3.4 Convergence Analysis

In the following, we are concerned with the convergence analysis of the proposed inpainting algorithms. As typical when dealing with nonlinear geometries, we show convergence under certain conditions. Our method is based on an unwrapping procedure which allows to apply results of [2, 7] subsequently. A related method was used by the authors in the context of denoising \(S^1\) data in [5]. Compared to the pure denoising approach, there are several issues we have to deal with in the inpainting situation: first, the proof in [5] relies on the uniqueness of the minimizers in the unwrapped situation which is not given for inpainting; second, a main step in the aforementioned proof is based on bounding all distances \(d_X(f_{ij}, x_{ij}), (i,j) \in \Omega_0\), to get information on \(x\), while for inpainting the values \(f_{ij}, (i,j) \in \Omega\), are missing.

We first briefly discuss the conditions we impose for our convergence analysis. Then we derive the necessary information to prove our main result formulated as Theorem 3.14. It states that both the algorithm proposed for inpainting and the algorithm proposed for simultaneous inpainting and denoising converge to a minimizer.

We employ the following notation to denote the distance on the first \(m\) components of two data items \(x, y \in X\). We notice that those are the \(S^1\)-valued components. We let

\[
d_{X,m}(x, y) := d_{(S^1)^m}(x_S, y_S).
\]

Our first condition is that the data \(f: \Omega^C \to X\) given on the complement of the inpainting region \(\Omega\) is *dense enough* in the sense that the distance between pixels and their neighbors in \(\Omega^C\) is sufficiently small. More precisely, we consider

\[
d_{\infty}^G(f) = \max_{(i,j) \in \Omega_0} \max_{(k,l) \in \Omega_1} \frac{d_{X,m}(f_{ij}, f_{kl})}{d((i,j), (k,l))},
\]

where, on the grid, \(d((i,j), (k,l))\) is the length of the shortest path with respect to the eight-
Lemma 3.6. Let \( d_{\mathcal{N}_{i,j}} = \{ (k,l) \in \Omega^n : \text{ for each } m < \gcd(k-i,l-j), (i,j) + \frac{m}{\gcd(k-i,l-j)}(k-i,l-j) \in \Omega \} \)

denotes the set of nearest neighbors belonging to \( \Omega^n \) with respect to the taxicab metric extended by diagonal steps. Here \( \gcd(r,s) \) denotes the greatest common divisor of \( r \) and \( s \). We note that the above definition takes the inpainting region into account and only restricts the spherical components. It turns out that for the non-spherical linear space components, no restrictions are necessary, and that large distances in these components do not influence the behavior in the spherical components negatively.

We use the notation \( d_{\infty}(f) = d_{\mathcal{N}_i}(f) \) when there is no inpainting region, i.e., the pure denoising situation. Then, due to the triangle inequality, our definition reduces to \( \mathcal{N}_{i,j} \) being the eight-neighborhood. Using this notation, we assume that \( d_{\infty}(f) \) is sufficiently small. A precise quantification of “sufficiently small” is given in the lemmas and theorems later on. We notice that assuming data being dense enough is quite common when dealing with manifold-valued data, see e.g. [78, 79]. Our second requirement is that the parameter sequence \( \alpha, \beta, \gamma \) are sufficiently small. For large parameters, solutions become almost constant which is often undesired and causes an interpretation problem, e.g. when the original data is equally distributed around the circle. Finally, we require that the parameter sequence \( \{ \lambda_k \} \) of the CPPA fulfills (15) with a small \( \ell^2 \) norm. The latter can be achieved by rescaling the parameter sequence.

Our analysis is based on an unwrapping procedure which means that we ‘lift the whole setup’ to the universal covering of \( \mathcal{X} \) which we denote by \( \mathcal{Y} = \mathbb{R}^{n+m} \).

Similar to the notation \( d_{\mathcal{X},m} \), we use

\[
|x-y|_{\mathcal{Y},m} := \| (x_i)_{i=1}^m - (y_i)_{i=1}^m \|_2
\]

to denote the distance on the first \( m \) components of the data \( x, y \in \mathcal{Y} \).

Universal coverings stem from algebraic topology. We refer to [45] for an introduction. A covering consists of a covering space and a canonical projection (inducing discrete fibers). We here explicitly consider the canonical projections \( \pi_x \) which are for \( x \in \mathcal{X} \) given by

\[
\pi_x(y) = \pi_x(y_5, y_R) = (\exp_{x5}(y_5), y_R), \quad y \in \mathcal{X},
\]

i.e. the linear space components remain unchanged while the cyclic components undergo the exponential mapping component-wise. It is well known that continuous mappings to the base space have a lifting to the covering space. The lifting is uniquely determined by specifying \( \pi^{-1}(x) \) for only one \( x \). This lifting construction also applies to discrete mappings \( g: \Omega_0 \to \mathcal{X} \) whenever \( d_{\infty}(g) < \pi \). We record this observation for further use.

Lemma 3.6. Let \( g: \Omega_0 \to \mathcal{X} \) be an image with \( d_{\infty}(g) < \pi \), and consider \( q \in \mathcal{X} \) fulfilling \( d_{\mathcal{N}_i}(g_i, (g_{1,1})) < \pi \), \( i = 1, \ldots, m \), i.e. no pair of cyclic data components is antipodal. We choose \( \tilde{g}_{1,1} \in \mathcal{Y} \) such that \( \pi_5(\tilde{g}_{1,1}) = g_{1,1} \). Then there exists a unique lifted image \( \tilde{g}: \Omega_0 \to \mathcal{Y} \) such that \( \pi_5 \tilde{g} = g \) holds component-wise and \( d_{\infty}(g) < \pi \).

Next, we lift the inpainting functionals and derive relations between the lifted and not lifted functionals and lifted and non lifted discrete functions. To precisely formulate these relations we need some preparation.
For $d > 0$ and data $f : \Omega^C \to X$ given on the complement of the inpainting region $\Omega$, we consider the class $S^\Omega(f, \delta)$ of grid functions $x$ defined on the whole domain $\Omega_0$, which we define by

$$S^\Omega(f, \delta) = \{ x : \Omega_0 \to X : e_\infty(x, f) \leq \delta \},$$

where

$$e_\infty(x, f) := \max_{(i,j) \in \Omega_0} d_{X,m}(x_{i,j}, f_{\nu(i,j)}),$$

and the mapping

$$\nu : \Omega_0 \to \Omega \text{ assigns } (i, j) \in \Omega_0 \text{ a nearest neighbor in } \Omega.$$  

Here we measure the nearness with respect to the distance on the grid induced by taking shortest paths with respect to the eight-neighborhood. The $x$ specified this way are ‘near’ to the images $f$ on $\Omega^C$ and do not vary too much in $\Omega$. We also need an extension operator $E$ extending a function $f$ defined on $\Omega^C$ to a function $E(f)$ defined on $\Omega_0$. A particularly simple extension operator is the nearest neighbor operator $E_\nu$, defined, for all $(i, j)$, by

$$E_\nu f_{i,j} = f_{\nu(i,j)}$$

with $\nu$ as in (33). Then there is a constant $D_\nu$, independent of $f$ but dependent on $\Omega$, such that

$$d_{\infty}(E_\nu(f)) \leq D_\nu(\Omega) d_{\infty}^*(f).$$

We come to the ‘lifted’ inpainting functionals. We first notice that we may write (13) in the form (14) modifying the distance term to be infinite if $x \neq f$ on $\Omega^C$. Then, on the universal covering space $Y$ of $X$, the inpainting functional $J_\Omega$ read

$$J_\Omega(x) = \tilde{F}_{\Omega^C}(x; \tilde{f}) + \alpha \tilde{TV}_1(x) + \beta \tilde{TV}_2(x) + \gamma \tilde{TV}_{1,1}(x),$$

where $\tilde{f} : \Omega^C \to Y$ is an lifted image of $f$. We get the following relations:

**Lemma 3.7.** Let $f : \Omega^C \to X$ with $d_{\infty}^*(f) < \frac{\pi}{8D_\nu(\Omega)}$ be given and let $q \in X$ be a point not antipodal to $f_{\nu(1,1)}$ in any sphere component. Choose a point $\tilde{f}_{\nu(1,1)}$ with $\pi(q; \tilde{f}_{\nu(1,1)}) = f_{\nu(1,1)}$ and let $\tilde{f}$ be the corresponding lifting of $f$ obtained from the extension using $E_\nu$.

Then every $x \in S^\Omega(f, \frac{\pi}{8})$ has a unique lifting $\tilde{x}$ w.r.t. the base point $q$ fulfilling $|\tilde{x}_{\nu(1,1)} - \tilde{f}_{\nu(1,1)}|_Y \leq \frac{\pi}{8}$. Furthermore,

$$J_\Omega(x) = \tilde{J}_\Omega(\tilde{x}) \text{ for all } x \in S^\Omega(f, \delta),$$

where $J_\Omega$ either denotes the functional in (13) or (14) and $\tilde{J}_\Omega$ is its analogue in $Y$ by (36).

**Proof.** Let us consider $x \in S^\Omega(f, \delta)$. For $(k,l)$ in the eight-neighborhood of $(i,j)$, we have

$$d_{X,m}(x_{i,j}, x_{k,l}) \leq d_{X,m}(x_{i,j}, f_{\nu(i,j)}) + d_{X,m}(f_{\nu(i,j)}, f_{\nu(k,l)}) + d_{X,m}(f_{\nu(k,l)}, x_{k,l})$$

$$< \frac{2\pi}{8} + D_\nu(\Omega) d_{\infty}^*(f) \leq \frac{3\pi}{8}.$$
For the second inequality we applied the definition of $S^\Omega(f,\delta)$, and for the third inequality, we applied (35). As a consequence, $d_{\infty}(x) < \frac{3\varepsilon}{\pi}$. By assumption, we have $d_{X,m}(x_{n(1)},f_{n(1)}) < \frac{\varepsilon}{8}$. Therefore, every $x \in S(f,\frac{\varepsilon}{8})$ has a unique lifting $\tilde{x}$ by Lemma 3.6 w.r.t. to the base point $q$ fulfilling $|\tilde{x}_{1,1} - \tilde{f}_{1,1}|_{\infty} \leq \frac{\varepsilon}{8}$.

In order to show (37) we show equality for each of the involved summands. First, we consider TV$_1$. By Lemma 3.6 we have $d_{X,m}(x_{i,j},x_{k}) = |x_{i,j} - x_{k}|_{y,m}$, $k \in \{(i,j+1),(i+1,j),(i+1,j+1)\}$. Hence the definitions of TV$_1$ and TV$_2$ imply TV$_1(x) = TV_1(\tilde{x})$. Concerning second order differences, we first consider the expressions $D_2(x_{i-1,j},x_{i,j},x_{i+1,j})$. Similar to (38), $d_{X,m}(x_{i-1,j},x_{i+1,j}) < \frac{\varepsilon}{2}$ which implies that the distance between any two members of the triple is smaller than $\frac{\varepsilon}{2}$. Due to the properties of the lifting $\tilde{x}$ this implies $D(\tilde{x}_{i-1,j},\tilde{x}_{i,j},\tilde{x}_{i+1,j};b_2) < \pi$. The same argument applies to $D_2(x_{i,j-1},x_{i,j},x_{i,j+1})$ which yields the equality for the TV$_2$ terms. A similar argument shows that TV$_{1,1}(x) = TV_{1,1}(\tilde{x})$. Concerning the data term $F(x;f)$ we need $r_{i,j} = d_{X,m}(x_{i,j},f_{\nu(i,j)})$ and $\tilde{r}_{i,j} = |\tilde{x}_{i,j} - \tilde{f}_{\nu(i,j)}|_{y,m}$ to agree for any $(i,j) \in \Omega^C$. By definition of $S^\Omega(f,\delta)$, we have $r_{i,j} \leq \frac{\varepsilon}{2}$ for all $(i,j) \in \Omega_0$. Furthermore, by the construction of $\tilde{f}$ and $\tilde{x}$ it holds $\tilde{r}_{i,j} = r_{i,j} + 2r_k$, with $k \in \mathbb{N}$ and $k_{\nu(1,1)} = 0$. We estimate $|\tilde{r}_{i,j+1} - \tilde{r}_{i,j}| = |\tilde{x}_{i,j+1} - \tilde{f}_{\nu(i,j+1)}|_{y,m} - |\tilde{x}_{i,j} - \tilde{f}_{\nu(i,j)}|_{y,m} \leq \frac{\varepsilon}{4}$. If $k_{i,j} \neq k_{i,j+1}$, then there exists $k \in \mathbb{Z}\{0\}$ such that

$$|\tilde{r}_{i,j+1} - \tilde{r}_{i,j}| = |\tilde{r}_{i,j+1} - \tilde{r}_{i,j} + 2\pi k| \geq 2\pi - \frac{\varepsilon}{4} > \frac{\varepsilon}{4}.$$  

This is a contradiction and therefore $k_{i,j} = k_{i,j+1}$. Similarly we conclude $k_{i,j} = k_{i+1,j}$. Hence, $k_{i,j} = k_{\nu(1,1)} = 0$ for all $(i,j) \in \Omega_0$ which implies $r_{i,j} = \tilde{r}_{i,j}$ for all $(i,j) \in \Omega^C$ and completes the proof. 

To formulate the next lemma we need the quantity $d_{1}^\Omega$ for functions defined on the complement of the inpainting region. We consider $f : \Omega^C \to \mathcal{X}$, and define $d_{1}^\Omega(f)$ in analogy to (30) by

$$d_{1}^\Omega(f) = \sum_{(i,j) \in \Omega_0} \max_{(k,l) \in N_{i,j}} d_{X,m}(f_{i,j},f_{k,l})$$

with $N_{i,j}$ as in (31). For the nearest neighbor extension operator $E_{\nu}$ defined in (34) we have the following estimate: there is a constant $C_{\nu}$, independent of $f$ but dependent on $\Omega$, such that

$$d_{1}(E_{\nu}(f)) := d_{1}^\Omega(E_{\nu}(f)) \leq C_{\nu}(\Omega)d_{1}^\Omega(f).$$

**Lemma 3.8.** Let $f : \Omega^C \to \mathcal{X}$ be given and set $p = \max\{\alpha_1,\ldots,\alpha_4,\beta_1,\beta_2,\gamma\}$. Let $\varepsilon > 0$ and choose $p$ so small that

$$d_{1}^\Omega(f) \leq \frac{1}{20pC_{\nu}(\Omega)} \min\left(\left(\varepsilon^2,\frac{\varepsilon}{2}\right)\right),$$

where $C_{\nu}(\Omega)$ is given by (35). Then any minimizer $x^*$ of the inpainting functional $J_{\Omega}$ given in (13) or (14) fulfills

$$e_{\infty}(x^*, f) \leq \varepsilon.$$
Proof. Applying the definition of $e_\infty$ in (32) we consider a minimizer $x^*$ of $J_\Omega$ given by (13) or (14) and estimate

$$d_{X,m}(x_{i,j}, f_{\nu(i,j)}) \leq d_{X,m}(x_{i,j}, x_{\nu(i,j)}) + d_{X,m}(x_{\nu(i,j)}, f_{\nu(i,j)}).$$

(40)

If $J_\Omega$ is given by (13), then the second term in (40) equals 0. If $J_\Omega$ is given by (14), we extend the data $f$ given on $\Omega^C$ to a grid function $g$ defined on $\Omega_0$ by setting $g_{i,j} = E_\nu(f_{i,j}) = f_{\nu(i,j)}$ for all $(i, j) \in \Omega$. We get the estimate

$$J_\Omega(x^*) \leq J_\Omega(g) = \alpha TV_1(g) + \beta TV_2(g) + \gamma TV_{1,1}(g).$$

(41)

We further estimate the right hand side of (41): since the second order differences may be estimated by two times the first order differences, we get $\beta TV_2(g) \leq 2(\max \beta_i)(1, 1, 0, 0) TV_1(g)$, and an analogous inequality for $\gamma TV_{1,1}(g)$. Hence

$$J_\Omega(g) \leq 5 \max \{\alpha_1, \ldots, \alpha_4, \beta_1, \beta_2, \gamma\} (1, 1, 1, 1) TV_1(g).$$

Next, we estimate each summand appearing in $TV_1(g)$ by the corresponding summand in $d_1(g)$ to conclude that $(1, 1, 1, 1) TV_1(g) \leq 4d_1(g)$. We use (35) to get

$$J_\Omega(g) \leq 20pd_1(g) \leq 20pC_\nu(\Omega)d_1^\Omega(f).$$

As a consequence we obtain for $(i, j) \in \Omega^C$

$$d_{X,m}(x_{i,j}, f_{\nu(i,j)})^2 \leq J_\Omega(x^*) \leq J_\Omega(g) \leq 20pC_\nu(\Omega)d_1^\Omega(f) \leq \left(\frac{\varepsilon}{2}\right)^2.$$

(42)

Looking at the first summand in (40), we notice that

$$d_{X,m}(x_{i,j}, x_{\nu(i,j)}) \leq (1, 1, 1, 1) TV_1(x^*) \leq J_\Omega(g) \leq 20pC_\nu(\Omega)d_1^\Omega(f) \leq \frac{\varepsilon}{2}.$$

(43)

Combining (43) and (42), we get

$$d_{X,m}(x_{i,j}, f_{\nu(i,j)}) \leq d_{X,m}(x_{i,j}, x_{\nu(i,j)}) + d_{X,m}(x_{\nu(i,j)}, f_{\nu(i,j)}) \leq \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon$$

which implies that $e_\infty(x, f) = \max_{(i,j)\in\Omega_0}\{d_{X,m}(x_{i,j}, f_{\nu(i,j)})\} \leq \varepsilon$ which finishes the proof. \hfill $\Box$

**Lemma 3.9.** Lemma 3.8 holds also true for data $f: \Omega^C \to \mathcal{V}$ and the inpainting functionals $\tilde{J}_\Omega$ given by (36).

**Proof.** The statement is obtained following the lines of the proof of Lemma 3.8. \hfill $\Box$

Now we combine Lemma 3.7 and 3.8 to locate the minimizers of $J$ and $\tilde{J}$. This part of the proof is rather similar to [5] which is the reason for streamlining it.

**Lemma 3.10.** Let $f: \Omega^C \to \mathcal{X}$ with $d_{X}^\Omega(f) < \frac{\varepsilon}{8\|\nu\|_{\infty}}$ and $\delta \leq \frac{\varepsilon}{8}$ be given. Let further $\varepsilon$ be given, such that $0 < \varepsilon < \delta \leq \frac{\varepsilon}{8}$. Choose the parameters $\alpha, \beta, \gamma$ of $J_\Omega$ from (13) or (14) such that (39) is fulfilled w.r.t. $\varepsilon$.

Then any minimizer $x^*$ of $J_\Omega$ lies in $S^\Omega(f, \delta)$. Furthermore, denote by $\tilde{f}$ the unique lifting of $f$ w.r.t. a base point $q$ and fixed $\tilde{f}_{\nu(1,1)}$ with $\pi_q(\tilde{f}_{\nu(1,1)}) = f_{\nu(1,1)}$. Then each minimizer $y^*$ of $\tilde{J}_\Omega$ defines a minimizer $x^* := \pi_q(y^*)$ of $J_\Omega$. Conversely, the uniquely defined lifting $\tilde{x}^*$ of a minimizer $x^*$ of $J_\Omega$ is a minimizer of $\tilde{J}_\Omega$.  

25
Proof. By Lemma 3.8, any minimizer $x^*$ of the inpainting functional $J_\Omega$ fulfills $e_\infty(x^*,f) \leq \varepsilon$. Since $\varepsilon < \delta$, we get $x^* \in S^\Omega(\bar{f},\delta)$. For the second statement we notice that the mapping $x \mapsto \tilde{x}$ is a bijection from $S^\Omega(\bar{f},\delta)$ to the set $S^\Omega(\bar{f},\delta) := \{ y : \Omega_0 \rightarrow \mathcal{Y} : \| y_{i,j} - f_{\nu(i,j)} \|_2 < \delta \}$. If $y^* \in \mathcal{Y}$ is a minimizer of $J_{\bar{f}}$, it lies in $S^\Omega(\bar{f},\delta)$ by Lemma 3.9. By (37) and the minimizing property of $y^*$ we obtain for any $x \in S^\Omega(\bar{f},\delta)$ that $J_{\bar{f}}(\pi_q(y^*)) = \bar{J}_\Omega(y^*) \leq \bar{J}_\Omega(\tilde{x}) = \bar{J}_\Omega(x)$. As a consequence, $\pi_q(y^*)$ is a minimizer of $J_{\bar{f}}$ on $S^\Omega(\bar{f},\delta)$. By Lemma 3.8 all the minimizers of $J_{\bar{f}}$ are contained in $S^\Omega(\bar{f},\delta)$ and hence $\pi_q(y^*)$ is a minimizer of $J_{\bar{f}}$. For the last statement let $x^*$ be a minimizer of $J_{\bar{f}}$. For its lifting $\tilde{x}^*$ and any $\tilde{y} \in S^\Omega(\bar{f},\delta)$, we get $\tilde{J}_{\bar{f}}(\tilde{x}^*) = J_{\bar{f}}(x^*) \leq J_{\bar{f}}(\pi_q(\tilde{y})) = \tilde{J}_{\bar{f}}(\tilde{y})$. Thus, $\tilde{x}^*$ is a minimizer of $\tilde{J}_{\bar{f}}$ on $S^\Omega(\bar{f},\delta)$. Since by Lemma 3.9 all minimizers of $\tilde{J}$ lie in $S^\Omega(\bar{f},\delta)$, the last assertion follows.

After establishing relations between these functionals and their lifted versions, grid functions and data, we next formulate a convergence result for vector space data in $\mathcal{Y}$. It is a reformulation of a convergence result which can be found in [2] for the more general class of Hadamard spaces or which can be derived from [7].

**Theorem 3.11.** Let $J = \sum_{l=1}^c J_l$, with each $J_l$ being a proper, closed, convex functional on $\mathcal{Y}_{\Omega_0}$ and assume that $J$ has a global minimizer. Assume further that there is $L > 0$ such that the iterates $\{x^{(k+\frac{1}{l})}\}$ of the CPPA, cf. Algorithm 1, fulfill

$$J_l(x^{(k)}) - J_l(x^{(k+\frac{1}{l})}) \leq L\|x^{(k)} - x^{(k+\frac{1}{l})}\|_2, \quad l = 1, \ldots, c,$$

for all $k \in \mathbb{N}_0$. Then the sequence $\{x^{(k)}\}_k$ converges to a minimizer of $J$. In particular, the iterates fulfill

$$\|x^{(k+\frac{1}{l})} - x^{(k+\frac{1}{l})}\|_2 \leq 2\lambda_k L,$$

(44)

and, for all $x \in \mathcal{Y}_{\Omega_0}$,

$$\|x^{(k+1)} - x\|_2^2 \leq \|x^{(k)} - x\|_2^2 - 2\lambda_k (J(x^{(k)}) - J(x)) + 2\lambda_k^2 L^2 c (c + 1).$$

(45)

Next we locate the iterates of the CPPA for vector space data in $\mathcal{Y}$ on a ball whose radius can be controlled. Since the data $f : \Omega^C \rightarrow \mathcal{Y}$ is not defined on the whole grid $\Omega_0$, we incorporate an extension operator $E$, e.g. $E_\nu$. An extension is needed as an initialization of the CPPA. We note that there is a positive number $L'$ such that the iterates $\{x^{(k+\frac{1}{l})}\}$ produced by Algorithm 1 fulfill

$$\|E(f) - x^{(k+\frac{1}{l})}\|_\infty \leq L'.$$

(46)

This can be seen by taking $m, M \in \mathbb{R}$ as the minimum and maximum of all components and pixels of $E(f)$, respectively, and then noticing that these minima and maxima only become greater or smaller, respectively, during the iterations.

**Lemma 3.12.** Let $f : \Omega^C \rightarrow \mathcal{Y}$ and a parameter sequence $\lambda = \{\lambda_k\}_k$ of the CPPA with property (15) be given. Further let $\{x^{(k+\frac{1}{l})}\}$ be the sequence produced by Algorithm 1 for the inpainting functionals $\bar{J}_{\Omega}$ given by (36). Let $x^* : \Omega_0 \rightarrow \mathcal{Y}$ be a minimizer of $\bar{J}_{\Omega}$. Then, for all $k \in \mathbb{N}_0$ and all $l \in \{1, \ldots, c\}$, we have

$$\|x^{(k+\frac{1}{l})} - x^*\|_2 \leq R := \sqrt{\|E(f) - x^*\|_2^2 + 2\lambda_2^2 L^2 c (c + 1) + 2\|\lambda\|_\infty c L},$$

(47)
where \( L = \max(4, L') \) using \( L' \) from (46) and \( c \) denotes the number of inner iterations, i.e. \( c = 18 \) in case of (13) and \( c = 19 \) in case of (14), respectively, and \( E \) is an operator extending \( \mathcal{Y} \) valued functions defined on \( \Omega^C \) to \( \Omega_0 \) used for initializing the algorithm.

**Proof.** Equation (45) in Theorem 3.11 tells us that

\[
\|x^{(k+1)} - x^*\|_2^2 \leq \|x^{(k)} - x^*\|_2^2 + 2\lambda_c[\tilde{J}(x^{(k)}) - \tilde{J}(x)] + 2\lambda_c^2 L^2 c(c + 1). \tag{48}
\]

We choose \( L \) as the maximum of the Lipschitz constants for the terms originating from \( \overline{TV}_1, \overline{TV}_2 \) and \( \overline{TV}_{1,1} \) which are all bounded by 4. For the quadratic data term, we may differentiate \( x_{i,j} \to \frac{1}{2}|f_{i,j} - x_{i,j}|^2 \) and notice that the \( x_{i,j} \) are confined to an \( L' \) ball around \( E(f)_{i,j} \) which bounds \( L \) by \( L' \) in this case. Therefore, we can set \( L = \max(4, L') \). We apply (48) with a minimizer \( x = x^* \) to obtain

\[
\|x^{(k+1)} - x^*\|_2^2 \leq \|x^{(k)} - x^*\|_2^2 + 2\lambda_c^2 L^2 c(c + 1) \leq \|x^{(0)} - x^*\|_2^2 + 2\sum_{j=0}^k \lambda_j^2 L^2 c(c + 1).
\]

Using \( x^{(0)} = E(f) \) yields \( \|x^{(k+1)} - x^*\|_2^2 \leq \|E(f) - x^*\|_2^2 + 2\|\lambda\|_2^2 L^2 c(c + 1) \). Now we use (44) and the triangle inequality to estimate

\[
\|x^{(k+\frac{1}{2})} - x^*\|_2^2 \leq 2\lambda_c L + \|E(f) - x^*\|_2^2 \leq 2\lambda_c L + \sqrt{|E(f) - x^*|_2^2 + 2\|\lambda\|_2^2 L^2 c(c + 1)},
\]

which completes the proof. \( \square \)

The following Lemma shows that lifting commutes with applying the proximal mappings for the previous assumptions.

**Lemma 3.13.** Let \( f : \Omega^C \to \mathcal{X} \) with \( \delta_{\Xi}(f) < \frac{\pi}{8D_\nu(\Omega)} \) and its lifting \( \tilde{f} \) of \( f \) w.r.t. a base point \( q \) as before. For each summand \( J_l \) in the splitting \( J_{\Omega} = \sum_{l} J_l \) from Section 3.3 for both inpainting functionals \( J_{\Omega} \) defined via (13) and (14), their corresponding functionals \( \tilde{J}_{\Omega} \) from (36), any \( x \in \mathcal{S}(f, \delta) \), \( \delta \in (0, \frac{\pi}{8}] \), and its lifting \( \tilde{x} \) w.r.t. \( q \), we have

\[
\text{prox}_{\lambda J_l}(x) = \pi_q(\text{prox}_{\lambda \tilde{J}_l}(\tilde{x})), \tag{49}
\]

for all \( l \in \{1, \ldots, 18\} \) in case of (13), and for all \( l \in \{1, \ldots, 19\} \) in case of (14).

**Proof.** The functional \( J_{19} \) appearing in the splitting of (14) is based on the distance to the data \( f \) for items \((i,j) \in \Omega^C \). Since \( x \in \mathcal{S}(f, \delta) \), it holds \( d_{\chi,m}(x_{i,j}, f_{i,j}) \leq \frac{\delta}{2} \) for all \((i,j) \in \Omega^C \). The components of the proximal mapping \( \text{prox}_{\lambda J_{19}} \) are given by Proposition 3.5 from which we conclude (49) for \( l = 19 \). The other proximal mappings of \( J_1, \ldots, J_{18} \), are given via proximal mappings of the first and second order cyclic differences from Theorem 3.3. We first consider cyclic components the first order differences \( D_1 \). By the triangle inequality we have

\[
d_{\chi,m}(x_{i,j}, x_{i,j+1}) \leq d_{\chi,m}(x_{i,j}, f_{\nu(i,j)}) + d_{\chi,m}(f_{\nu(i,j)}, f_{\nu(i,j+1)}) + d_{\chi,m}(x_{i,j+1}, x_{\nu(i,j+1)}) \leq \frac{2\pi}{8} + D_{\nu}(\Omega) \delta_{\Xi}(f) \leq \frac{3\pi}{8}.
\]

27
Analogously we get \( d(\mathcal{G})_m(x_{i,j}, x_{i+1,j}) \leq \frac{3\pi}{8} \). By the explicit form of the proximal mapping given in Theorem 3.3 we obtain (49) for the \( J_l, l = 1, \ldots, 8 \) which involve first order differences. Next we consider the second order differences \( D_2 \) with respect to the cyclic components. Let us exemplarily consider the vertical second order difference \( D_2(x_{i,j-1}, x_{i,j}, x_{i,j+1}) \). Analogously as above, we see \( d_{\chi,m}(x_{i,j-1}, x_{i,j}) < \frac{3\pi}{16}, d_{\chi,m}(x_{i,j}, x_{i,j+1}) < \frac{3\pi}{16} \) as well as \( d_{\chi,m}(x_{i,j-1}, x_{i,j}) < \frac{\pi}{2} \). Hence all contributing values of \( x \) lie in a common ball of radius \( \pi/2 \). Applying the proximal mapping in Theorem 3.3 the resulting points lie in a common open ball of radius \( \pi \).

An analogous statement holds true for the horizontal part. Hence the proximal mappings of these second differences agree with the cyclic version under identification via \( \pi_q \). This implies (49) for \( J_9, \ldots, J_{14} \). It remains to deal with the cyclic components of the mixed second order differences \( D_{1,1}(x_{i,j}, x_{i+1,j}, x_{i,j+1}, x_{i+1,j+1}) \). As above, we have for neighboring data items that the distance is smaller than \( \frac{3\pi}{8} \). For all four contributing values of \( x \) we have that the pairwise distance is smaller by \( \frac{\pi}{2} \). So they again lie in a common \( \pi \) ball and the proximal mappings agree under identification. This completes the proof.

In the following main theorem we combine the preceding lemmas to show that the output of the applied proximal mappings remains in \( \mathcal{S}^\Omega(f, \delta) \). This then allows for an iterated application of Lemma 3.13.

**Theorem 3.14.** We consider data \( f : \Omega^C \rightarrow X \) with \( d^C_\infty(f) < \frac{\pi}{2D_\min(1)} \). We choose the parameter sequence \( \lambda = \{\lambda_k\}_k \) fulfilling property (15) and \( \varepsilon > 0 \) such that

\[
\sqrt{4\varepsilon^2 + 2\|\lambda\|^2_2L^2(c+1) + 2\|\lambda\|_\infty cL} < \frac{\pi}{16},
\]

where \( c = 18 \) or \( c = 19 \) and \( L = \max(4, L') \) with \( L' \) as in (46). We assume further that the parameter vectors \( \alpha, \beta, \gamma \) of the inpainting functionals \( J_{1,2} \) given by either (13) or (14) satisfy (39) and that the initialization of the inpainting region \( Ef \) is close to the nearest neighbor extension \( E_\nu(f) \) in the sense that \( d^\Omega_\chi(E_\nu(f), E_\nu(f)) \leq \varepsilon \). Then the sequence \( \{x^{(k)}\}_k \) generated by the CPPA given by Algorithm 1 converges to a global minimizer of \( J \).

**Proof.** Let \( \tilde{f} \) be the lifting of \( E_\nu(f) \) with respect to a base point \( q \) not antipodal to \( f_{n(1,1)} \) and fixed \( \tilde{f}_{n(1,1)} \) with \( \pi_q(\tilde{f}_{n(1,1)}) = f_{n(1,1)} \). Furthermore, let \( \tilde{J}_1 \) denotes the analogue of \( J_\Omega \) for \( \mathcal{Y} \)-valued data given by (36). Since \( d^\Omega_\chi(E(\tilde{f}), E_\nu(f)) \leq \varepsilon \leq \frac{\pi}{32} \), the function \( E(f) \) is in \( \mathcal{S}^\Omega(f, \frac{\pi}{8}) \) and we may apply Lemma 3.7 to conclude that, for any \( x \) with \( d^\Omega_{\chi,m}(x, E(\tilde{f})) < \frac{3\pi}{8} \), its unique lifting \( \tilde{x} \) fulfills \( \tilde{J}_1(\tilde{x}) = J_\Omega(x) \). From Lemma 3.8 we conclude that the minimizer \( y^* \) of \( \tilde{J}_1 \) fulfills \( \|\tilde{y}^* - \tilde{f}\|_2 \leq \varepsilon < \frac{\pi}{32} \). By (47) we obtain

\[
R = \sqrt{\|\tilde{y}^* - \tilde{E}(\tilde{f})\|^2_2 + 2\|\lambda\|^2_2L^2(c+1) + 2\|\lambda\|_\infty cL} \\
\leq \sqrt{2\|\tilde{y}^* - \tilde{f}\|^2_2 + 2\|\lambda\|^2_2L^2(c+1) + 2\|\lambda\|_\infty cL} \\
\leq \sqrt{4\varepsilon^2 + 2\|\lambda\|^2_2L^2(c+1) + 2\|\lambda\|_\infty cL} < \frac{\pi}{16},
\]

where \( \tilde{E}(\tilde{f}) \) denotes the lifting of \( E(f) \). By Lemma 3.12 the iterates \( y^{(k+\frac{1}{2})} \) of the CPPA for \( \mathcal{Y} \)-valued data fulfill

\[
\|y^{(k+\frac{1}{2})} - y^*\| \leq R < \frac{\pi}{16}.
\]
Hence $\|y^{(k+\frac{1}{c})} - \tilde{f}\|_\infty < \frac{\pi}{8}$ which means that all iterates $y^{(k+\frac{1}{c})}$ stay within $\tilde{S}_\Omega(\tilde{f}, \frac{\pi}{8})$ and $\|(y^{(k+\frac{1}{c})} - E(f))\| < \frac{3\pi}{32}$.

After these preparations we now consider the sequence $\{x^{(l+k+c)}\}$ of the CPPA for $X$-valued data $f$ with initialization $E(f)$. We show that $x^{(k+\frac{1}{c})} = \pi_q(y^{(k+\frac{1}{c})})$. By definition, $x^{(0)} = E(f) = \pi_q(E(f)) = \pi_q(y^{(0)})$. We assume that $x^{(k+\frac{1}{c})} = \exp_q(y^{(k+\frac{1}{c})})$. By the local bijectivity of the lifting shown in Lemma 3.7, and since $y^{(k+\frac{1}{c})} \in \tilde{S}_\Omega(\tilde{f}, \delta)$, we conclude $x^{(k+\frac{1}{c})} \in S^\Omega(f, \delta)$. By Lemma 3.13, $\pi_q(y^{(k+\frac{1}{c})}) = \pi_q(\text{prox}_{\lambda_k J_l}(y^{(k+\frac{1}{c})})) = \text{prox}_{\lambda_k J_l}(x^{(k+\frac{1}{c})}) = x^{(k+\frac{1}{c})}$.

By the same argument as above we have again $x^{(k+\frac{1}{c})} \in S(f, \delta)$. Finally, Theorem 3.11 tells us that $x^{(k)} = \pi_q(y^{(k)}) \to \pi_q(y^*)$ as $k \to \infty$ and by Lemma 3.10 $x^* := \pi_q(y^*)$ is a global minimizer of $J$. This completes the proof.

### 4 Applications

In this section we apply our algorithms to various image processing tasks. As a first example, we consider denoising in nonlinear color spaces in Section 4.1. Our second application is inpainting in such color spaces: in Section 4.2, we consider a noise-free setup; in Section 4.3 we combine inpainting and denoising. Finally, we apply our algorithms for denoising frames in volumetric phase-valued data – in our case, frames of a 2D film. Our approach is based on utilizing the neighboring $k$ frames to incorporate the temporal neighborhood. The idea generalizes to arbitrary data spaces and volumes consisting of layers of 2D data. A detailed description and applications are given in Section 4.4.

The algorithms were implemented in MATLAB. The computations for the following examples were performed on a MacBook Pro with an Intel Core i5, 2.6 Ghz and Mac OS 10.10.1.

#### 4.1 Denoising nonlinear color space data

As a first application, we consider denoising color space data. Various nonlinear color spaces have been considered in the literature; examples are luma plus chroma/chrominance based spaces such as YIQ, YUV and YDbDr and HSL type color space such as HSL, HSI or HSD. We here consider the HSV (hue-saturation-value) color space: the hue component is cyclic, the saturation and the value component are real-valued.

We apply our algorithm for denoising combined cyclic and linear data to these $S^1 \times \mathbb{R}^2$ valued data. We compare the results with the usual approach using the linear RGB color space. For both spaces, we compare our approach on the product space with a model that denoises each channel separately. Finally, we compare the results of all these approaches under different noise models: we impose Gaussian noise on each component in RGB space and we impose Gaussian and wrapped Gaussian noise on the saturation and value component and the hue component, respectively, in HSV space.
In Figure 5, a colorful drawing of a sailboat\(^1\) of size 512 \(\times\) 512 pixel is obstructed by noise on all three channels of the HSV color space: for the hue, which is given on \([0, 1]\) we applied wrapped Gaussian noise \((\text{mod} \ 1)\) with \(\sigma = \frac{1}{15}\), for saturation and value, which are also given on the same range but are not cyclic, we applied Gaussian noise, also with \(\sigma = \frac{1}{15}\). The resulting image is shown in Figure 5 (a). We then apply four different first and second order differences based approaches, where for each, the best result among a range of parameter choices is shown, where \(\alpha := \alpha_1 = \alpha_2 \in \frac{1}{32} \mathbb{N}_0^d\) and the same range applies for \(\beta := \beta_1 = \beta_2 = \gamma\). We measure the quality using a peak signal to noise ratio (PSNR) on RGB: first, we apply a real valued approach to each of the RGB channels separately. For \(\alpha = 0, \beta = \frac{1}{16}\) we obtain a PSNR of 21.05, which is shown in Figure 5 (b). Applying a vector valued approach on RGB, i.e. setting \(m = 0, n = 3\) in Algorithm 1 with \(k = 400\) iterations, we obtain for \(\alpha = 0, \beta = \frac{1}{8}\) the result shown in Figure 5 (c) having a PSNR of 21.19, which outperforms the component-wise denoising. This stems from the fact, that a vector valued approach takes edges into account, that occur in several channels together and keeps them aligned. On the other hand, we apply Algorithm 1 to each of the channels of HSV, i.e., setting \(m = 1, n = 0\) for the first channel and taking the real valued case from above for the second and third. In order to keep the channels unscaled, the algorithm presented in this paper is rescaled to run w.r.t. to mod 1. The result is shown in Figure 5 (d) yielding a PSNR of 22.52. Finally, applying a vector valued approach on HSV, i.e. setting \(m = 1\) and \(n = 2\) —again having the cyclic channel w.r.t. mod 1— yields an image shown in Figure 5 (e), having a PSNR of 22.78, the best result of all four compared algorithms. Note that especially the colors are much better reconstructed than in the RGB based denoising approaches, which both suffer from reduced saturation. Furthermore, edges can be much better recognized than in both channel wise approaches, see especially the magnified region of the sail. As a challenge for our approach we now impose color noise, i.e., independent Gaussian noise, \(\sigma = \frac{1}{5}\), on all three channels of the RGB color space; see Figure 6 (a). For this kind of noise, the RGB approach seems particularly suited. The experiments in Figure 6 show results comparable with those for HSV noise. Even more, the parameters obtained by performing the parameter search on the same range of parameters yields for all cases the same optimal parameters as for the HSV noise case. So despite being completely different noise, the results are again better for the cyclic and non-cyclic data case. We conclude that also in case of color noise, the combined cyclic and non-cyclic approach on the HSV color space is favorable over the vector space approach, i.e. in RGB space.

### 4.2 Inpainting noise-free data

Here we consider the situation where some data items are missing, are lost or have been removed by a user. As example space we again consider the HSV color space. As in Section 4.1, we compare the results with the usual approach using the linear RGB color space and with the HSV approach working component-wise.

We consider a synthetic image in the HSV color space given by the function

\[
(\text{atan2} \frac{x}{y}, 1 - x^2, 1 - |x + y|), \quad x, y \in [-\frac{1}{2}, \frac{1}{2}],
\]

where the first component is the arctangent function with two arguments. This extends a

\(^1\)Taken from the USC-SIPI Image Database, see \url{http://sipi.usc.edu/database/database.php?volume=misc&image=14}
Figure 5. Denoising an image with independent wrapped Gaussian and Gaussian noise $\sigma = \frac{1}{3}$ on each of the HSV channels. The RGB-based approaches (b),(c) produce less colorful results than the HSV-based approaches (d),(e). In contrast to channel-wise denoising (d), the combined approach proposed in this paper (e) gets the object boundaries more properly. Moreover, the approach in (e) yields the best PSNR.

A synthetic example already used by the authors in [6]. The original image is shown in Figure 7(f). The initial data is obtained by removing a disc with radius $r = \frac{1}{4}$ as shown in Figure 7(a). The goal is to “recover” the image in Figure 7(f). For the inpainting we again apply Algorithm 1 using $k = 800$ iterations and performing a parameter search on $\alpha := \alpha_1 = \alpha_2, \beta := \beta_1 = \beta = 2 = \gamma \in \frac{1}{5}\mathbb{N}_0$. Then, the real valued approaches in RGB color space, both channel wise (b) and vectorial (c), do not reconstruct the original colors correctly. In contrast, the channel wise approach (d) and the vectorial approach (e) on the HSV space keep the colors and both produce a very satisfactory results. The result of the vectorial approach in Figures 7 (e) is a nuance “sharper”, its reconstruction is a little bit better in PSNR at least.
Figure 6. Denoising an image with independent Gaussian noise $\sigma = \frac{1}{8}$ on each RGB channel. The results of the RGB-based approaches (b),(c) are less colorful than those of the HSV-based approaches (d),(e). Compared with channel-wise denoising (d), the object boundaries are reconstructed more properly by the approach of this paper (e) which employs a $S^1 \times \mathbb{R}^2$ model. The approach in (e) yields the best PSNR.

4.3 Inpainting and denoising data

In many situations data is noisy and parts are lost or invalid. This results in an combined inpainting and denoising problem for which we apply the proposed methods next. As in Sections 4.1 and 4.2 before, we consider the HSV color space and compare the results with the usual approach using the linear RGB color space and with the HSV approach working component-wise.

As a test scenario, we add wrapped Gaussian and Gaussian noise with with $\sigma = \frac{1}{8}$ to the cyclic and non-cyclic components, respectively, similar to Section 4.1. Furthermore, we remove a ball in the center as also done in Section 4.2; see Figure 8. This initial data is shown in Figure 8 (a). Again, we would like to get back the image shown in Figure 8 (f). The parameters for the algorithm are obtained using the same setup for Algorithm 1 as in Section 4.1. Then, the real valued approaches in RGB color space, both channel wise (b) and vectorial (c), do not reconstruct the significant features. In contrast, the channel wise approach (d) and the vectorial
Figure 7. Reconstruction of a synthetic image with the black inner circle missing in a noiseless setup. The RGB-based reconstructions (a),(b) both yield a degrading of the colors, while the HSV-based approaches (c),(d) reconstruct the colors. The proposed approach (d) based on a $S^1 \times \mathbb{R}^2$ model yields the best PSNR.

approach (e) on the HSV space reconstruct all significant features and produce almost perfect results. The result of the vectorial approach in Figures 8 (e) is a nuance “sharper” yielding a slightly higher PSNR.

4.4 Denoising sections in volumetric cyclic data

Finally, we apply our algorithms for denoising frames in volumetric data. Examples of volumetric data are frames of a 2D film or a stack of slices, each slice being a 2D image as appearing, e.g., in computed tomography. We want to denoise such slices incorporating the temporal/spatial information stemming from the third dimension, which is, e.g., the temporal neighborhood information in a film.

To be more precise, we consider volumetric data $I_i(x,y)$, where $I_i(x,y)$ is an data item at the bivariate pixel location $x, y$ in the $i$th frame/slice. The setup is rather general, and we can assume $I_i(x,y)$ being data from some rather general space $M$ – say a manifold; here we
Figure 8. Reconstruction of a synthetic image with the black inner circle indicating missing data and the measurement itself being noisy. The RGB based reconstructions (b),(c) miss the main smooth features. The HSV model based reconstructions (d),(e) reconstructs these features yielding a satisfactory result. The proposed vectorial approach (e) yields the best PSNR.

exemplarily consider $M = S^1$. We take a look at the $k$ neighboring (left and right) frames $I_l, l = i - k, \ldots, i, \ldots, i + k$ around a center frame $I = I_i$ at position $i$. With the position $i$, we now associate bivariate data living in $M^{2k+1}$ being the vector of data points at the same position $(x, y)$ in the neighboring frames. To be precise, we consider the bivariate data $J_i$, with $J_i(x, y) \in M^{2k+1}$ given by

$$J_i(x, y) = (I_{i-k}(x, y), I_{i-k+1}(x, y), \ldots, I_i(x, y), \ldots, I_{i+k-1}(x, y), I_{i+k}(x, y)).$$

We apply our algorithms to the derived data $J_i$ and compare the result to the usual denoising of the single frame $I_i$.

The video underlying Figure 9 is constructed as follows. As basis, we use the image given by the first component of (50) on $[-\frac{1}{2}, \frac{1}{2}]^2$; outside the disc of radius $\frac{1}{2}$, we add $\frac{\pi}{4}$ which is the same as rotating the input $(x, y)$ of each pixel in $\Omega_0$ by the same amount clockwise. The video consists of 13 frames rotation the disc clockwise by $\pi$ and the outer region by $\frac{\pi}{2}$ counterclockwise, i.e. from $-\frac{\pi}{2}$ before the center frame to $\frac{\pi}{2}$ at the end of the sequence, which
is a rotation of $\frac{\pi}{12}$ per frame. We finally sample each of these frames with 256 × 256 pixel on $[-\frac{1}{2}, \frac{1}{2}]^2$.

In our example, Figure 9, we show in (d) the seventh frame of the constructed video from the previous the paragraph. On each frame, we impose wrapped Gaussian noise with standard deviation $\sigma = \frac{2}{5}$, see Figure 9(a) for frame 7 of the video. In (b), we perform denoising just on the frame $i = 7$. Performing a first and second order denoising yields staircasing and/or reduction of the sharp edge at the disc border. Choosing the parameters $\alpha, \beta \in \frac{1}{64} N_0$ and $k = 400$ iterations, we obtain the optimal value $\alpha = \frac{1}{32}, \beta = 0$, which indicates, that staircasing still resembles a better result than unsharpening the edge. In (c), we perform a combined vectorial denoising in $(S^1)^{13}$ as proposed for a total of 13 frames ($k = 6, i = 7$) and show the central seventh frame. At the cost of being computationally more expensive due to the increased data set, this approach outperforms the first, single frame based approach.

5 Conclusion and Future Research

In this article we dealt with denoising, inpainting and combined denoising and inpainting for combined cyclic and linear space data. We derived cyclic proximal point solvers for the corresponding second order variational models in these nonlinear data spaces. Especially for all occurring summands of first and second order differences in combined vector valued data spaces, the corresponding proximal mappings are derived in explicit formulas, even for the case, where some data items are fixed by constraints. We develop a cyclic proximal point algorithm including an efficient splitting of the variational models in our focus. We further provide a convergence analysis. We applied our algorithms to denoising, inpainting and combined denoising and inpainting problems in the nonlinear HSV color space and for denoising frames in volumetric phase-valued data.

A topic of future research are algorithms for higher order TV-type functionals for data living in more general manifolds.
References

[1] M. Bačák. The proximal point algorithm in metric spaces. *Isr. J. Math.*, 194(2):689–701, 2013.

[2] M. Bačák. Computing medians and means in Hadamard spaces. *SIAM J. Optim.*, 24(3):1542–1566, 2014.

[3] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera. Filling in by joint interpolation of vector fields and gray levels. *IEEE Trans. Image Process.*, 10(8):1200–1211, 2001.

[4] P. Basser, J. Mattiello, and D. LeBihan. MR diffusion tensor spectroscopy and imaging. *Biophys. J.*, 66:259–267, 1994.

[5] R. Bergmann, F. Laus, G. Steidl, and A. Weinmann. Second order differences of cyclic data and applications in variational denoising. *SIAM J. Imaging Sci.*, 2015. to appear.

[6] R. Bergmann and A. Weinmann. Inpainting of cyclic data using first and second order differences. In *EMMCVPR 2015*, pages 155–168. Springer, 2015.

[7] D. P. Bertsekas. Incremental gradient, subgradient, and proximal methods for convex optimization: a survey. Technical Report LIDS-P-2848, Laboratory for Information and Decision Systems, MIT, Cambridge, MA, 2010.

[8] D. P. Bertsekas. Incremental proximal methods for large scale convex optimization. *Math. Program.*, Ser. B, 129(2):163–195, 2011.

[9] R. Bhattacharya and V. Patrangenaru. Large sample theory of intrinsic and extrinsic sample means on manifolds I. *Ann. Statist.*, 31(1):1–29, 2003.

[10] R. Bhattacharya and V. Patrangenaru. Large sample theory of intrinsic and extrinsic sample means on manifolds II. *Ann. Statist.*, 33(3):1225–1259, 2005.

[11] P. Blomgren and T. Chan. Color TV: total variation methods for restoration of vector-valued images. *IEEE Trans. Image Process.*, 7(3):304–309, 1998.

[12] F. Bornemann and T. März. Fast image inpainting based on coherence transport. *J. Math. Imaging Vision*, 28(3):259–278, 2007.

[13] K. Bredies, K. Kunisch, and T. Pock. Total generalized variation. *SIAM J. Imaging Sci.*, 3(3):1–42, 2009.

[14] X. Bresson and T. F. Chan. Fast dual minimization of the vectorial total variation norm and applications to color image processing. *Inverse Probl. Imaging*, 2(4):455–484, 2008.

[15] A. Bugeau, M. Bertalmio, V. Caselles, and G. Sapiro. A comprehensive framework for image inpainting. *IEEE Trans. Signal Process.*, 19:2634–2645, 2010.

[16] J.-F. Cai, R. H. Chan, and Z. Shen. Simultaneous cartoon and texture inpainting. *Inverse Probl. Imaging*, 4(3):379–395, 2010.

[17] J.-F. Cai, B. Dong, S. Osher, and Z. Shen. Image restoration: Total variation, wavelet frames, and beyond. *J. Amer. Math. Soc.*, 25(4):1033–1089, 2012.
[18] V. Caselles, J.-M. Morel, and C. Sbert. An axiomatic approach to image interpolation. *IEEE Trans. Image Process.*, 7(3):376–386, 1998.

[19] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numer. Math.*, 76(2):167–188, 1997.

[20] T. F. Chan, S. Esedoglu, and F. E. Park. Image decomposition combining staircase reduction and texture extraction. *J. Vis. Commun. Image R.*, 18(6):464–486, 2007.

[21] T. F. Chan, S. Kang, and J. Shen. Total variation denoising and enhancement of color images based on the CB and HSV color models. *J. Vis. Commun. Image R.*, 12:422–435, 2001.

[22] T. F. Chan and S. H. Kang. Error analysis for image inpainting. *J. Math. Imaging Vis.*, 26(1-2):85–103, 2006.

[23] T. F. Chan, S. H. Kang, and J. Shen. Euler’s elastica and curvature-based inpainting. *SIAM J. Appl. Math.*, pages 564–592, 2002.

[24] T. F. Chan, A. Marquina, and P. Mulet. High-order total variation-based image restoration. *SIAM J. Sci. Comput.*, 22(2):503–516, 2000.

[25] T. F. Chan and J. Shen. Local inpainting models and TV inpainting. *SIAM J. Appl. Math.*, 62(3):1019–1043, 2001.

[26] T. F. Chan and J. J. Shen. *Image processing and analysis: variational, PDE, wavelet, and stochastic methods*. SIAM, 2005.

[27] T. F. Chan, A. M. Yip, and F. E. Park. Simultaneous total variation image inpainting and blind deconvolution. *Int. J. Imag. Syst. Tech.*, 15(1):92–102, 2005.

[28] C. Chefd’Hotel, D. Tschumperlé, R. Deriche, and O. Faugeras. Regularizing flows for constrained matrix-valued images. *J. Math. Imaging Vis.*, 20:147–162, 2004.

[29] S. Didas, G. Steidl, and S. Setzer. Combined $\ell_2$ data and gradient fitting in conjunction with $\ell_1$ regularization. *Adv. Comput. Math.*, 30(1):79–99, 2009.

[30] S. Didas, J. Weickert, and B. Burgeth. Properties of higher order nonlinear diffusion filtering. *J. Math. Imaging Vis.*, 35:208–226, 2009.

[31] B. Dong, H. Ji, J. Li, Z. Shen, and Y. Xu. Wavelet frame based blind image inpainting. *Appl. Comput. Harmon. Anal.*, 32(2):268–279, 2012.

[32] M. Elad, J.-L. Starck, P. Querre, and D. L. Donoho. Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA). *Appl. Comput. Harmon. Anal.*, 19(3):340–358, 2005.

[33] S. Esedoglu and J. Shen. Digital inpainting based on the mumford–shah–euler image model. *European J. of Appl. Math.*, 13(04):353–370, 2002.

[34] O. P. Ferreira and P. R. Oliveira. Proximal point algorithm on Riemannian manifolds. *Optimization*, 51(2):257–270, 2002.

[35] N. I. Fisher. Statistical Analysis of Circular Data. Cambridge University Press, 1995.

[36] P. Fletcher. Geodesic regression and the theory of least squares on Riemannian manifolds.
[37] M. Giaquinta, G. Modica, and J. Sourček. Variational problems for maps of bounded variation with values in $S^1$. *Calc. Var.*, 1(1):87–121, 1993.

[38] M. Giaquinta and D. Mucci. The BV-energy of maps into a manifold: relaxation and density results. *Ann. Sc. Norm. Super. Pisa Cl. Sci.*, 5(4):483–548, 2006.

[39] M. Giaquinta and D. Mucci. Maps of bounded variation with values into a manifold: total variation and relaxed energy. *Pure Appl. Math. Q.*, 3(2):513–538, 2007.

[40] B. Goldluecke, E. Strekalovskiy, and D. Cremers. The natural vectorial total variation which arises from geometric measure theory. *SIAM J. Imaging Sci.*, 5(2):537–563, 2012.

[41] B. Goldluecke, E. Strekalovskiy, and D. Cremers. Tight convex relaxations for vector-valued labeling. *SIAM J. Imaging Sci.*, 6(3):1626–1664, 2013.

[42] P. Grohs, H. Hardering, and O. Sander. Optimal a priori discretization error bounds for geodesic finite elements. *Found. Comput. Math.*, 2015. to appear.

[43] P. Grohs and J. Wallner. Interpolatory wavelets for manifold-valued data. *Appl. Comput. Harmon. Anal.*, 27(3):325–333, 2009.

[44] C. Guillemot and O. Le Meur. Image inpainting: Overview and recent advances. *IEEE Signal Processing Mag.*, 31(1):127–144, 2014.

[45] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.

[46] W. Hinterberger and O. Scherzer. Variational methods on the space of functions of bounded Hessian for convexification and denoising. *Computing*, 76(1):109–133, 2006.

[47] S. R. Jammalamadaka and A. SenGupta. *Topics in Circular Statistics*. World Scientific Publishing Company, 2001.

[48] R. Kimmel and N. Sochen. Orientation diffusion or how to comb a porcupine. *J. Vis. Commun. Image R.*, 13:238–248, 2002.

[49] E. J. King, G. Kutyniok, and X. Zhuang. Analysis of inpainting via clustered sparsity and microlocal analysis. *J. Math. Imaging Vis.*, 48(2):205–234, 2014.

[50] R. Lai and S. Osher. A splitting method for orthogonality constrained problems. *J. Sci. Comput.*, 58(2):431–449, 2014.

[51] S. Lefkimmiatis, A. Bourquard, and M. Unser. Hessian-based norm regularization for image restoration with biomedical applications. *IEEE Trans. Image Process.*, 21(3):983–995, 2012.

[52] J. Lellmann, E. Strekalovskiy, S. Koetter, and D. Cremers. Total variation regularization for functions with values in a manifold. In *IEEE ICCV 2013*, pages 2944–2951, 2013.

[53] M. Lysaker, A. Lundervold, and X.-C. Tai. Noise removal using fourth-order partial differential equations with applications to medical magnetic resonance images in space and time. *IEEE Trans. Image Process.*, 12(12):1579–1590, 2003.

[54] M. Lysaker and X.-C. Tai. Iterative image restoration combining total variation minimization and a second-order functional. *Int. J. Comput. Vis.*, 66(1):5–18, 2006.
[55] T. M"arz. Image inpainting based on coherence transport with adapted distance functions. *SIAM J. Imaging Sci.*, 4(4):981–1000, 2011.

[56] T. M"arz. A well-posedness framework for inpainting based on coherence transport. *Found. Comput. Math.*, 2013.

[57] S. Masnou. Disocclusion: a variational approach using level lines. *IEEE Trans. Image Process.*, 11(2):68–76, 2002.

[58] S. Masnou and J.-M. Morel. Level lines based disocclusion. In *IEEE ICIP 1998*, pages 259–263, 1998.

[59] T. Möllenhoff, E. Strekalovskiy, M. Möller, and D. Cremers. Low rank priors for color image regularization. *Preprint*, 2014.

[60] J. Oller and J. Corcuera. Intrinsic analysis of statistical estimation. *Ann. Statist.*, 23(5):1562–1581, 1995.

[61] X. Pennec. Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements. *J. Math. Imaging Vis.*, 25(1):127–154, 2006.

[62] X. Pennec, P. Fillard, and N. Ayache. A Riemannian framework for tensor computing. *Int. J. Comput. Vis.*, 66:41–66, 2006.

[63] L. L. Rakét, L. Roholm, M. Nielsen, and F. Lauze. TV-$L^1$ optical flow for vector valued images. In *EMMCVPR 2011*, pages 329–343. Springer, 2011.

[64] R. T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM J. Control Optim.*, 14(5):877–898, 1976.

[65] G. Rosman, M. Bronstein, A. Bronstein, A. Wolf, and R. Kimmel. Group-valued regularization framework for motion segmentation of dynamic non-rigid shapes. In *SSVM 2011, LNCS 6667*, pages 725–736. Springer, 2012.

[66] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D.*, 60(1):259–268, 1992.

[67] O. Scherzer. Denoising with higher order derivatives of bounded variation and an application to parameter estimation. *Computing*, 60:1–27, 1998.

[68] S. Setzer and G. Steidl. Variational methods with higher order derivatives in image processing. In M. N. and Larry Schumaker, editor, *Approximation XII: San Antonio 2007*, pages 360–385, 2008.

[69] S. Setzer, G. Steidl, and T. Teuber. Infimal convolution regularizations with discrete L1-type functionals. *Commun. Math. Sci.*, 9(3):797–872, 2011.

[70] J. Shen and T. F. Chan. Mathematical models for local nontexture inpaintings. *SIAM J. Appl. Math.*, 62(3):1019–1043, 2002.

[71] E. Strekalovskiy and D. Cremers. Total variation for cyclic structures: Convex relaxation and efficient minimization. In *IEEE CVPR 2011*, pages 1905–1911, 2011.

[72] E. Strekalovskiy and D. Cremers. Total cyclic variation and generalizations. *J. Math. Imaging Vis.*, 47(3):258–277, 2013.
[73] D. Tschumperlé. Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE’s. *Int. J. Comput. Vis.*, 68(1):65–82, 2006.

[74] D. Tschumperlé and R. Deriche. Diffusion tensor regularization with constraints preservation. In *IEEE CVPR 2001*, volume 1, pages 948–953, 2001.

[75] I. Ur Rahman, I. Drori, V. Stodden, D. Donoho, and P. Schröder. Multiscale representations for manifold-valued data. *Multiscale Model. and Sim.*, 4:1201–1232, 2005.

[76] T. Valkonen, K. Bredies, and F. Knoll. Total generalized variation in diffusion tensor imaging. *SIAM J. Imaging Sci.*, 6(1):487–525, 2013.

[77] L. Vese and S. Osher. Numerical methods for p-harmonic flows and applications to image processing. *SIAM J. Numer. Anal.*, 40:2085–2104, 2002.

[78] J. Wallner and N. Dyn. Convergence and $C^1$ analysis of subdivision schemes on manifolds by proximity. *Comput. Aided Geom. D.*, 22:593–622, 2005.

[79] A. Weinmann. Nonlinear subdivision schemes on irregular meshes. *Constr. Approx.*, 31(3):395–415, 2010.

[80] A. Weinmann. Interpolatory multiscale representation for functions between manifolds. *SIAM J. Math. Anal.*, 44:162–191, 2012.

[81] A. Weinmann, L. Demaret, and M. Storath. Total variation regularization for manifold-valued data. *SIAM J. Imaging Sci.*, 7(4):2226–2257, 2014.