Technische Universität Kaiserslautern

Second Order Differences of Cyclic Data and Application to Variational Denoising

Ronny Bergmann*

Image Processing Group
University of Technology Kaiserslautern

May 14, 2014

Seminar of the Institute of Computational Biology
Helmholtz Center Munich

FELIX KLEIN
ZENTRUM FÜR
MATHEMATIK

*joint work with A. Weinmann, G. Steidl, F. Laus

Introduction

What do we want to do?

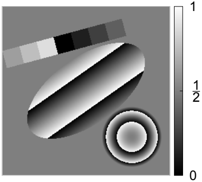


image $f \in [0,1]^{N\times M}$

Introduction

What do we want to do?

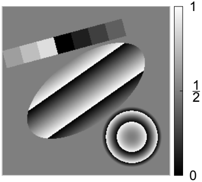
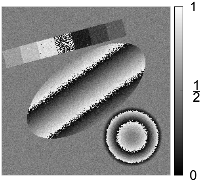


image $f \in [0,1]^{N\times M}$



disturbed by noise

Introduction

What do we want to do?

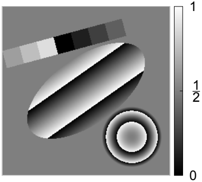
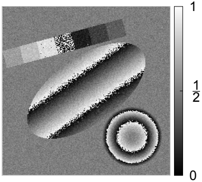


image $f \in [0,1]^{N\times M}$



disturbed by noise

Goal: reconstruct image f from noisy data, preserve edges.

Introduction

What do we want to do?

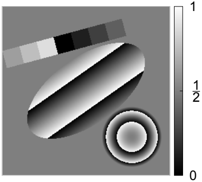
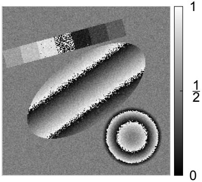
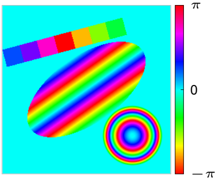


image $f \in [0,1]^{N \times M}$



disturbed by noise

Here: image f is phase valued, i.e. def. on the circle $\mathbb{S}^1 \cong [-\pi, \pi)$:



phase valued image f, $f_{i,j} \in \mathbb{S}^1$

Goal: reconstruct image f from noisy data, preserve edges.

Introduction

What do we want to do?

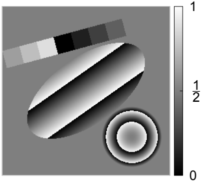
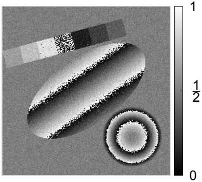
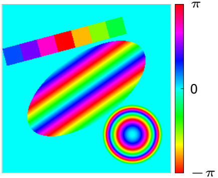


image $f \in [0,1]^{N\times M}$

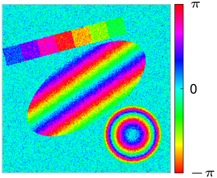


disturbed by noise

Here: image f is phase valued, i.e. def. on the circle $\mathbb{S}^1 \cong [-\pi, \pi)$:



phase valued image f, $f_{i,j} \in \mathbb{S}^1$



disturbed by noise

Goal: reconstruct phase valued image f from noisy data, preserve edges.

Introduction II

[Osher, Rudin, Fatemi, 1992]

- tool: minimizing the Rudin-Osher-Fatemi (ROF) functional

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

  - ∇ discrete gradient
  - $\sum_{i,j} |\nabla x_{i,j}|$ discrete total variation (TV)
  - regularization parameter $\lambda > 0$

$\Rightarrow$ edge-preserving

- stair caising-effect: reduced by adding higher order derivatives

Introduction II

[Osher, Rudin, Fatemi, 1992]

- tool: minimizing the Rudin-Osher-Fatemi (ROF) functional

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

  - ∇ discrete gradient
  - $\sum_{i,j} |\nabla x_{i,j}|$ discrete total variation (TV)
  - regularization parameter $\lambda > 0$

$\Rightarrow$ edge-preserving

- stair caising-effect: reduced by adding higher order derivatives

Recently [Cremers,Strekalovski, 2012], [Lellmann et al., 2013], [Weinmann et al., 2013]

- TV denoising generalized to Riemannian manifolds
- several algorithms to find the minimizer x^*

Outline

First & Second Order Differences on $\mathbb{R}$

A short reminder.

Let $w = (w_j)_{j=1}^d \in \mathbb{R}^d \setminus \{0\}$ be a weight: $\langle w, 1_d \rangle := \sum_{j=1}^{d} w_j = 0$

The finite difference operator is given by

$$\Delta(x; w) := \langle x, w \rangle, \quad x \in \mathbb{R}^d$$

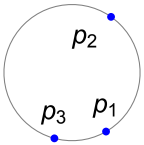
$\Delta(x; w)$ is shift invariant.

Examples

- $b_1 := (-1, 1)$: First order difference $\Delta(x; b_1) = x_2 - x_1$
- $b_2 := (1, -2, 1)$: Second order difference $\Delta(x; b_2) = x_1 - 2x_2 + x_3$
- $b_{1,1} := (-1, 1, 1, -1)$: 'mixed second order difference' $\Delta(x; b_{1,1}) = -x_1 + x_2 + x_3 - x_4$

First & Second Order Difference on $\mathbb{S}^1$

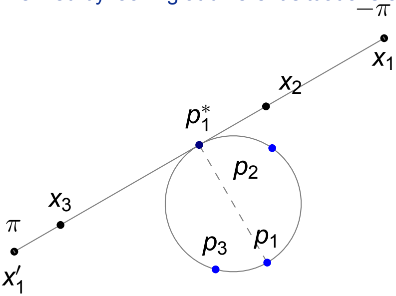
Defined by looking at different situations on $\mathbb{R}$ the points may take.



- $x_i \in [-\pi, \pi) \Leftrightarrow p_i := (\cos x_i, \sin x_i)$
- Idea: unwrap the circle onto any tangential line

First & Second Order Difference on $\mathbb{S}^1$

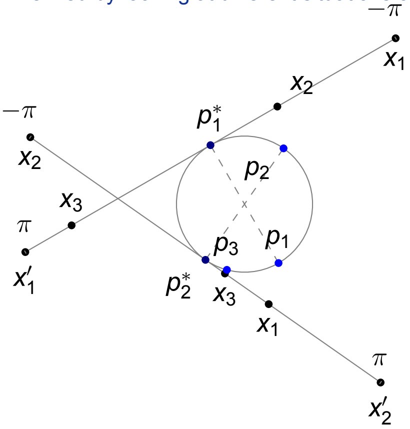
Defined by looking at different situations on $\mathbb{R}$ the points may take.



- $x_i \in [-\pi, \pi) \Leftrightarrow p_i := (\cos x_i, \sin x_i)$
- Idea: unwrap the circle onto any tangential line

First & Second Order Difference on $\mathbb{S}^1$

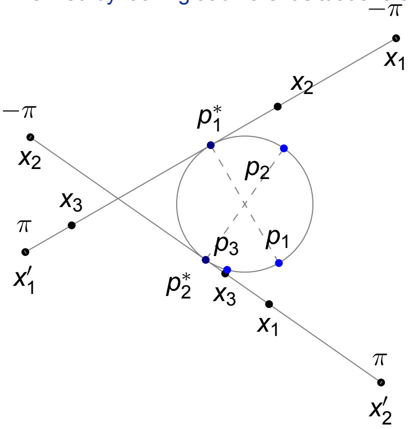
Defined by looking at different situations on $\mathbb{R}$ the points may take.



- $x_i \in [-\pi, \pi) \Leftrightarrow p_i := (\cos x_i, \sin x_i)$
- Idea: unwrap the circle onto any tangential line

First & Second Order Difference on $\mathbb{S}^1$

Defined by looking at different situations on $\mathbb{R}$ the points may take.



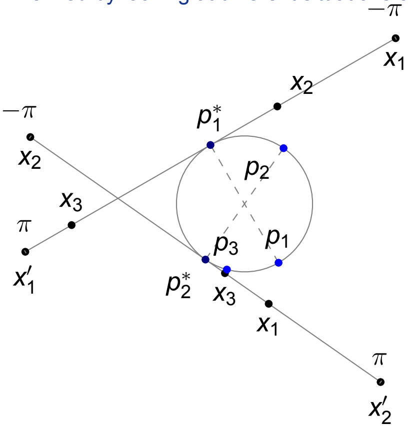
- $x_i \in [-\pi, \pi) \Leftrightarrow p_i := (\cos x_i, \sin x_i)$
- Idea: unwrap the circle onto any tangential line
- Absolute cyclic differences w.r.t w:

$$d(x; w) := \min_{\alpha \in \mathbb{R}} \left| \Delta\left([x + \alpha 1_d]_{2\pi}; w\right) \right|$$

- $[x]_{2\pi}$: element-wise mod 2π except $x_i = (2k+1)\pi$: Δ with $\pm\pi$
- shift invariant

First & Second Order Difference on $\mathbb{S}^1$

Defined by looking at different situations on $\mathbb{R}$ the points may take.



- $x_i \in [-\pi, \pi) \Leftrightarrow p_i := (\cos x_i, \sin x_i)$
- Idea: unwrap the circle onto any tangential line
- Absolute cyclic differences w.r.t w:

$$d(x; w) := \min_{\alpha \in \mathbb{R}} \left| \Delta\left([x + \alpha 1_d]_{2\pi}; w\right) \right|$$

- $[x]_{2\pi}$: element-wise mod 2π except $x_i = (2k+1)\pi$: Δ with $\pm\pi$
- shift invariant

- b_1: arc length distance $d(x; b_1) = d_1(x_1, x_2)$
- b_2: $d(x; b_2) = d_2(x_1, x_2, x_3) = \left| \left(\Delta(x; b_2)\right)_{2\pi} \right|$ (the same holds for $b_{1,1}$)

Second Order Total Variation on the Circle

Transfer the ROF functional to the circle.

Let $f = (f_i)_{i=1}^N$ be given data on $\mathbb{S}^1$, $\alpha, \beta \geq 0$.
We are interested in the minimizers x^* of

$$J(x) := F(x; f) + \alpha\,\mathrm{TV}_1(x) + \beta\,\mathrm{TV}_2(x),$$

where

- data fidelity term $F(x; f) = \frac{1}{2}\sum_{i=1}^{N} d_1(f_i, x_i)^2$
- first order differences $\mathrm{TV}_1(x) = \sum_{i=1}^{N-1} d_1(x_i, x_{i+1})$
- second order differences $\mathrm{TV}_2(x) = \sum_{i=2}^{N-1} d_2(x_{i-1}, x_i, x_{i+1})$

Proximal Point Algorithm

For a proper, closed, convex function $\varphi\colon \mathbb{R}^N \to (-\infty, +\infty]$ and $\lambda > 0$ the proximal mapping $\operatorname{prox}_{\lambda\varphi}\colon \mathbb{R}^N \to \mathbb{R}^N$ is defined by

$$\operatorname{prox}_{\lambda\varphi}(f) := \operatorname*{arg\,min}_{x\in\mathbb{R}^N} \frac{1}{2}\|f - x\|_2^2 + \lambda\varphi(x),$$

- trade-off: minimizing φ vs. "staying near" f
- λ: weight or trade-off parameter
- fixpoints of $\operatorname{prox}_{\lambda\varphi}$: minima of φ.
- often: closed form of $\operatorname{prox}_{\lambda\varphi}$ known.

Proximal Point Algorithm (PPA)

[Moreau, 1965; Rockafellar, 1976]

$$x^{(k+1)} = \operatorname{prox}_{\lambda\varphi}(x^{(k)}), \quad k \in \mathbb{N}$$

Cyclic Proximal Point Algorithm

Split into smaller proximal mappings and iterate.

- $\varphi = \sum_{i=1}^{c} \varphi_i,$ c is called the cycle length,
- proximal mappings of summands φ_i "easier"

$\Rightarrow$ iteratively apply "small" proximal mappings $\operatorname{prox}_{\lambda\varphi_i}$

Cyclic Proximal Point Algorithm (CPPA)

$$x^{(k+\frac{i+1}{c})} = \operatorname{prox}_{\lambda_k\varphi_i}(x^{(k+\frac{i}{c})}), \quad i = 0, \dots, c-1, k \in \mathbb{N}.$$

Lemma (Convergence of the CPPA on $\mathbb{R}$ [Bertsekas, 2011])

Let φ have a minimizer x^* and $\{\lambda_k\}_k$ be a sequence, such that

- $\sum_{k=1}^{\infty} \lambda_k = \infty$
- $\sum_{k=1}^{\infty} \lambda_k^2 < \infty$

Then the CPPA converges to a minimizer.

Proximal Mapping I

for each data fidelity term of data on $\mathbb{S}^1$.

- data fidelity term $\varphi(x) = d_1(f,x)^2, f \in [-\pi,\pi)$
- $\operatorname{prox}_{\lambda d_1(f,\cdot)^2}(g) = \arg\min_x \frac{1}{2} d_1(g,x)^2 + \lambda d_1(f,x)^2$
- idea again: "near g" vs. minimizing $d_1(f,x)^2$

Theorem (B., Laus, Steidl, Weinmann)

The unique minimizer x^* of $\operatorname{prox}_{\lambda d_1(f,\cdot)^2}(g)$ is

$$x^* = \left(\frac{g+\lambda f}{1+\lambda} + \frac{\lambda}{1+\lambda} 2\pi v\right)_{2\pi}, \quad v = \begin{cases} 0 & \text{for } |g-f| \leq \pi, \\ \operatorname{sgn}(g-f) & \text{for } |g-f| > \pi \end{cases}$$

Sketch of proof

- first term is the minimizer on $\mathbb{R}$
- second term the minimial value, taking $g + 2\pi k$, $f + 2\pi l$ into account

Proximal Mapping I

for each data fidelity term of data on $\mathbb{S}^1$.

- data fidelity term $\varphi(x) = d_1(f,x)^2, f \in [-\pi,\pi)$
- $\operatorname{prox}_{\lambda d_1(f,\cdot)^2}(g) = \arg\min_x \frac{1}{2} d_1(g,x)^2 + \lambda d_1(f,x)^2$
- idea again: "near g" vs. minimizing $d_1(f,x)^2$

Theorem (B., Laus, Steidl, Weinmann)

The unique minimizer x^* of $\operatorname{prox}_{\lambda d_1(f,\cdot)^2}(g)$ is

$$x^* = \left(\frac{g+\lambda f}{1+\lambda} + \frac{\lambda}{1+\lambda} 2\pi v\right)_{2\pi}, \quad v = \begin{cases} 0 & \text{for } |g-f| \leq \pi, \\ \operatorname{sgn}(g-f) & \text{for } |g-f| > \pi \end{cases}$$

Sketch of proof

- first term is the minimizer on $\mathbb{R}$
- second term the minimial value, taking $g + 2\pi k$, $f + 2\pi l$ into account

Proximal Mapping II

for the finite difference terms on $\mathbb{S}^1$.

- finite difference term $\varphi(x) = d(x; w), w \in \{b_1, b_2, b_{1,1}\}$
- x, g same length as w
- $\operatorname{prox}_{\lambda d(\cdot;w)}(g) = \arg\min_x d_1(g,x)^2 + \lambda d(x;w)$

Theorem (B., Laus, Steidl, Weinmann)

Set $s := \operatorname{sgn}(\langle g, w\rangle)_{2\pi}$ *and* $m := \min\left\{\lambda, \frac{|(\langle g,w\rangle)_{2\pi}|}{\|w\|_2^2}\right\}$.

1. *If* $|(\langle g, w\rangle)_{2\pi}| < \pi$, *the unique minimizer is given by*

$$x^* = (g - s\,m\,w)_{2\pi}$$

2. *If* $|(\langle g, w\rangle)_{2\pi}| = \pi$, *the two minimizers are*

$$x^* = (g \mp s\,m\,w)_{2\pi}$$

Idea of the proof: Minimizing over “possible constellations” on $\mathbb{R}$.

CPPA with Second Order TV for 1D Data on $\mathbb{S}^1$

How to split the higher order TV functional J?

- $F(x;f) = \frac{1}{2}\sum_{i=1}^{N} d_1(f_i,x_i)^2 =: J_1(x)$

proximal mapping I (applied element-wise)

- first order differences

$$\alpha\,\mathrm{TV}_1(x) = \alpha\sum_{i=1}^{N-1} d_1(x_i,x_{i+1})$$

- second order differences

$$\beta\,\mathrm{TV}_2(x) = \beta\sum_{i=2}^{N-1} d_2(x_{i-1},x_i,x_{i+1})$$

CPPA with Second Order TV for 1D Data on $\mathbb{S}^1$

How to split the higher order TV functional J?

- $F(x;f) = \frac{1}{2}\sum_{i=1}^{N} d_1(f_i,x_i)^2 =: J_1(x)$

proximal mapping I (applied element-wise)

- first order differences

$$\alpha\,\mathrm{TV}_1(x) = \sum_{l=0}^{1}\alpha\sum_{i=1}^{\lfloor\frac{N-1}{2}\rfloor} d_1(x_{2i-1+l},x_{2i-l}) =: J_2(x) + J_3(x)$$

- second order differences

$$\beta\,\mathrm{TV}_2(x) = \beta\sum_{i=2}^{N-1} d_2(x_{i-1},x_i,x_{i+1})$$

CPPA with Second Order TV for 1D Data on $\mathbb{S}^1$

How to split the higher order TV functional J?

- $F(x;f) = \frac{1}{2}\sum_{i=1}^{N} d_1(f_i,x_i)^2 =: J_1(x)$

  proximal mapping I (applied element-wise)

- first order differences

$$\alpha\,\mathrm{TV}_1(x) = \sum_{l=0}^{1} \alpha \sum_{i=1}^{\lfloor\frac{N-1}{2}\rfloor} d_1(x_{2i-1+l}, x_{2i-l}) =: J_2(x) + J_3(x)$$

  inner sum: distinct data $\Rightarrow$ proximal mapping II with $w = b_1$

- second order differences

$$\beta\,\mathrm{TV}_2(x) = \beta \sum_{i=2}^{N-1} d_2(x_{i-1}, x_i, x_{i+1})$$

CPPA with Second Order TV for 1D Data on $\mathbb{S}^1$

How to split the higher order TV functional J?

- $F(x;f) = \frac{1}{2}\sum_{i=1}^{N} d_1(f_i, x_i)^2 =: J_1(x)$

  proximal mapping I (applied element-wise)

- first order differences

$$\alpha\,\mathrm{TV}_1(x) = \sum_{l=0}^{1} \alpha \sum_{i=1}^{\lfloor \frac{N-1}{2} \rfloor} d_1(x_{2i-1+l}, x_{2i-l}) =: J_2(x) + J_3(x)$$

inner sum: distinct data $\Rightarrow$ proximal mapping II with $w = b_1$

- second order differences

$$\beta\,\mathrm{TV}_2(x) = \sum_{l=0}^{2} \beta \sum_{i=1}^{\lfloor \frac{N-1}{3} \rfloor} d_2(x_{3i-2+l}, x_{3i-1+l}, x_{3i+l}) =: \sum_{l=0}^{2} J_{4+l}(x)$$

inner sum: distinct data $\Rightarrow$ proximal mapping II with $w = b_2$

CPPA with Second Order TV for 1D Data on $\mathbb{S}^1$

How to split the higher order TV functional J?

- $F(x;f) = \frac{1}{2}\sum_{i=1}^{N} d_1(f_i,x_i)^2 =: J_1(x)$

  proximal mapping I (applied element-wise)

- first order differences

$$\alpha\,\mathrm{TV}_1(x) = \sum_{l=0}^{1} \alpha \sum_{i=1}^{\left\lfloor\frac{N-1}{2}\right\rfloor} d_1(x_{2i-1+l}, x_{2i-l}) =: J_2(x) + J_3(x)$$

  inner sum: distinct data $\Rightarrow$ proximal mapping II with $w = b_1$

- second order differences

$$\beta\,\mathrm{TV}_2(x) = \sum_{l=0}^{2} \beta \sum_{i=1}^{\left\lfloor\frac{N-1}{3}\right\rfloor} d_2(x_{3i-2+l}, x_{3i-1+l}, x_{3i+l}) =: \sum_{l=0}^{2} J_{4+l}(x)$$

  inner sum: distinct data $\Rightarrow$ proximal mapping II with $w = b_2$

$\Rightarrow$ $J(x) = \sum_{l=1}^{6} J_l(x)$, i.e., cycle length $c = 6$

Algorithm for CPP on $\mathbb{S}^1$

Input non-negative parameters $\lambda_0 > 0$ and α, β
data $f \in [-\pi, \pi)^N$

CPPA(α, β, λ_0, f)

Initialize $x^{(0)} \leftarrow f$, $k \leftarrow 0$

Initialize the cycle length $c \leftarrow 6$

Repeat

For l from 1 to c

$x^{(k-1+\frac{l}{c})} \leftarrow \operatorname{prox}_{\lambda_k J_l}(x^{(k-1+\frac{l-1}{c})})$

$k \leftarrow k+1$

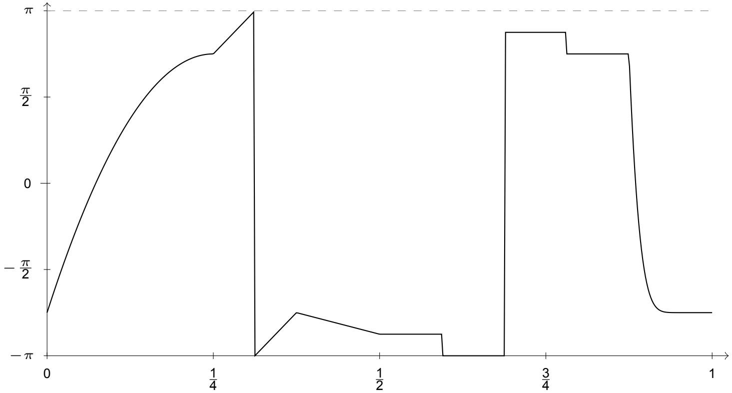
$\lambda_k \leftarrow \frac{\lambda_0}{k}$

Until a convergence criterion are reached

Return $x^{(k)}$

Example

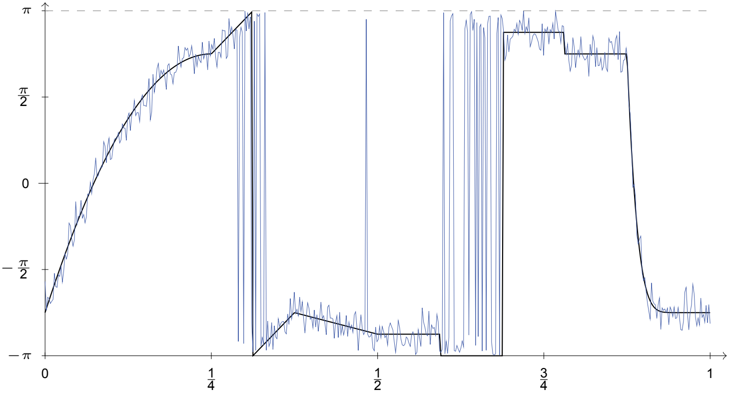
Denoising a 1D phase valued signal.



- function $f\colon [0,1] \to \mathbb{S}^1$ sampled to obtain data $f_{\text{o}} = (f_{\text{o},i})_{i=1}^{500}$
- jumps $> \pi$ at $\frac{5}{16}$ and $\frac{11}{16}$ are due to the representation system

Example

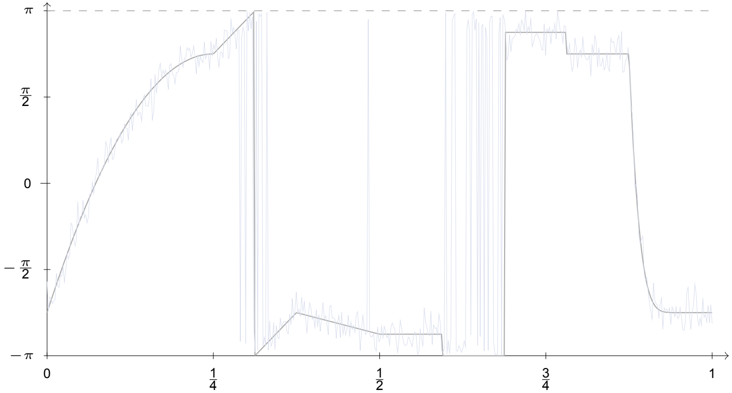
Denoising a 1D phase valued signal.



- function $f\colon [0,1] \to \mathbb{S}^1$ sampled to obtain data $f_{\mathrm{o}} = (f_{\mathrm{o},i})_{i=1}^{500}$
- adding wrapped Gaussian noise, $\sigma = 0.2$
- noisy data $f_{\mathrm{n}} = (f_0 + \eta)_{2\pi}$

Example

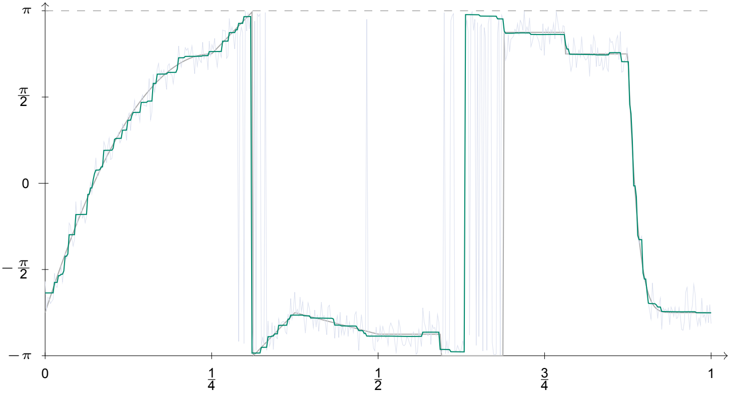
Denoising a 1D phase valued signal.



- comparison of f_o & f_n with

Example

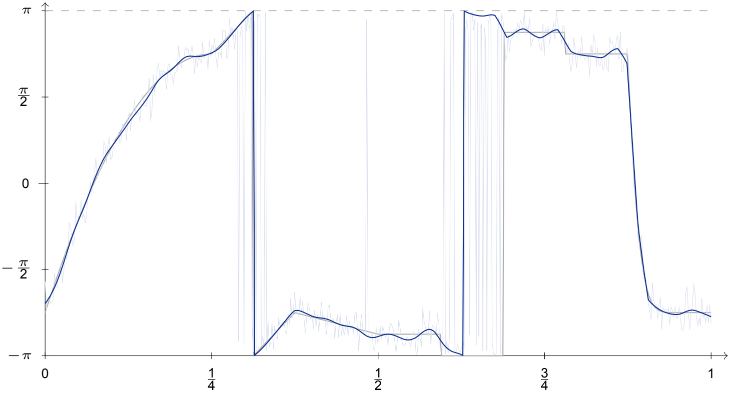
Denoising a 1D phase valued signal.



- comparison of f_o & f_n with f_1
- denoising: just TV_1: $\alpha = \frac{3}{4}$, $\beta = 0$
- but: stair casing

Example

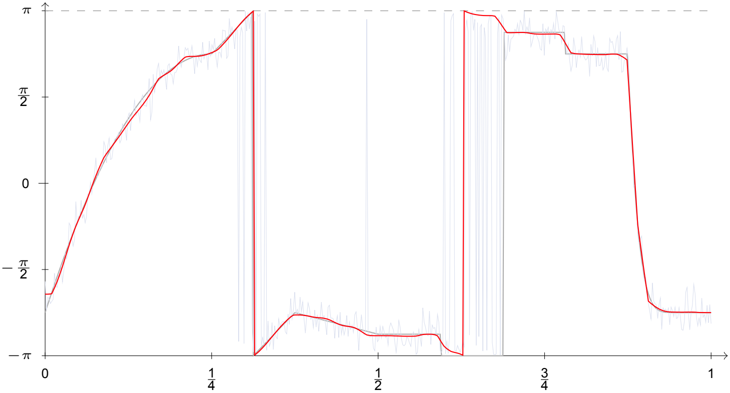
Denoising a 1D phase valued signal.



- comparison of f_o & f_n with f_2
- denoising: just TV_2: $\alpha = 0$, $\beta = \frac{3}{2}$
- but: no plateaus

Example

Denoising a 1D phase valued signal.



- comparison of f_o & f_n with f_3
- denoising: TV$_1$ & TV$_2$: $\alpha = \frac{1}{2}$, $\beta = 1$
- smallest mean squared error

CPPA with Second Order TV for 2D data on $\mathbb{S}^1$

Splitting the functional J for an $N \times M$ pixel image using mainly things we already know.

data $f := (f_{i,j})_{i,j=1}^{N,M} \in [-\pi, \pi)^{N\times M}$ and $\alpha = (\alpha_1, \alpha_2)$, $\beta = (\beta_1, \beta_2)$, γ

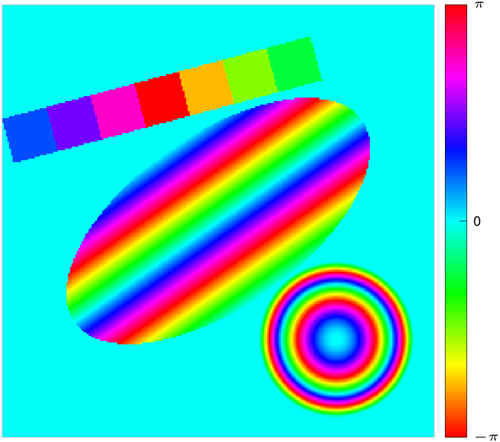
- $F(x; f)$ element-wise distance as before
- $$\alpha\, \mathrm{TV}_1(x) := \alpha_1 \sum_{i,j=1}^{N-1,M} d_1(x_{i,j}, x_{i+1,j}) + \alpha_2 \sum_{i,j=1}^{N,M-1} d_1(x_{i,j}, x_{i,j+1})$$
- $$\beta\, \mathrm{TV}_2^{\mathrm{hv}}(x) := \beta_1 \sum_{i=1,j=2}^{N-1,M} d_2(x_{i-1,j}, x_{i,j}, x_{i+1,j}) + \beta_2 \sum_{i=2,j=1}^{N,M-1} d_2(x_{i,j-1}, x_{i,j}, x_{i,j+1})$$
- $$\gamma\, \mathrm{TV}_2^{\mathrm{d}}(x) := \gamma \sum_{i,j=1}^{N-1,M-1} d_{1,1}(x_{i,j}, x_{i+1,j}, x_{i,j+1}, x_{i+1,j+1})$$

$\Rightarrow$ minimizing $J(x) := F(x; f) + \alpha\, \mathrm{TV}_1(x) + \beta\, \mathrm{TV}_2^{\mathrm{hv}}(x) + \gamma\, \mathrm{TV}_2^{\mathrm{d}}(x)$

- data term, 2×2 TV_1 terms, 2×3 $\mathrm{TV}_2^{\mathrm{hv}}$ terms, 4 $\mathrm{TV}_2^{\mathrm{d}}$ terms $\Rightarrow c = 15$

Example

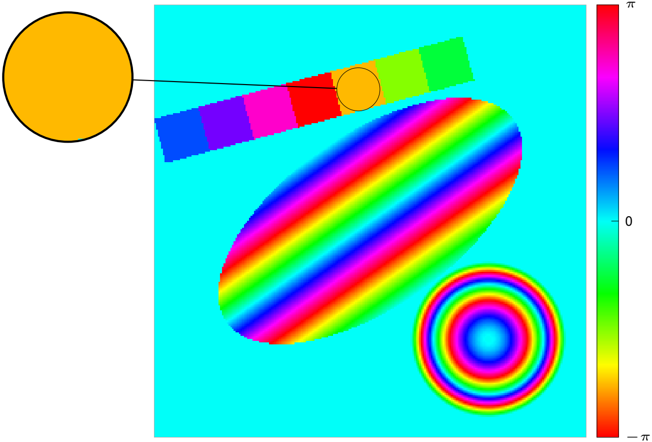
Denoising artificial phase valued data.



original data f_o, 256×256 pixel image

Example

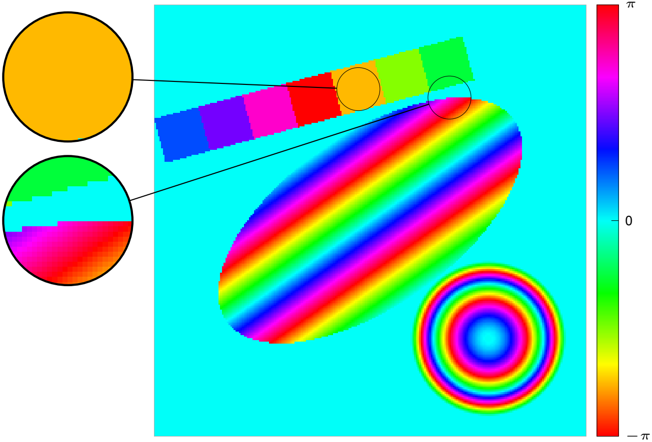
Denoising artificial phase valued data.



original data f_o, 256×256 pixel image

Example

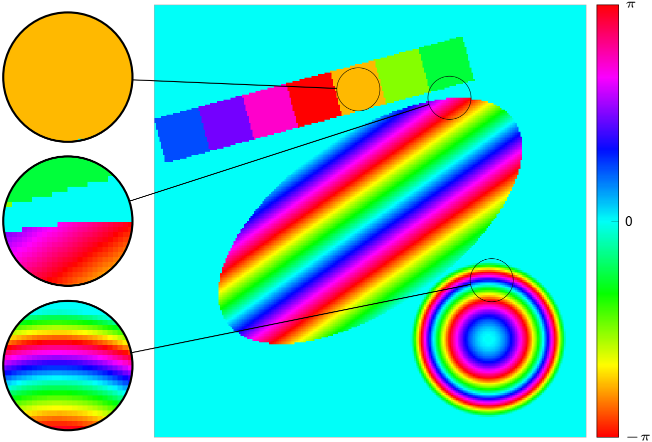
Denoising artificial phase valued data.



original data f_o, 256×256 pixel image

Example

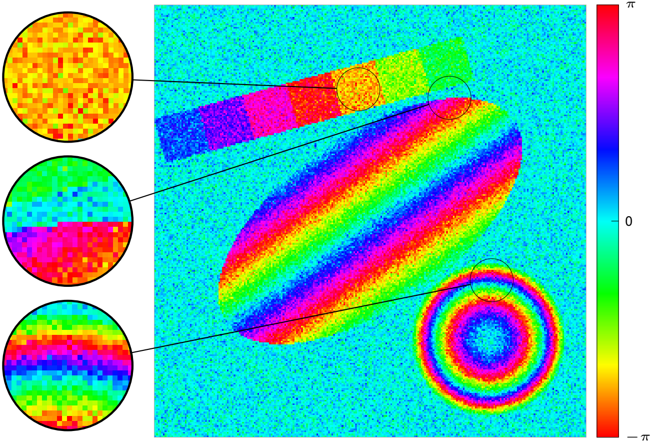
Denoising artificial phase valued data.



original data f_o, 256×256 pixel image

Example

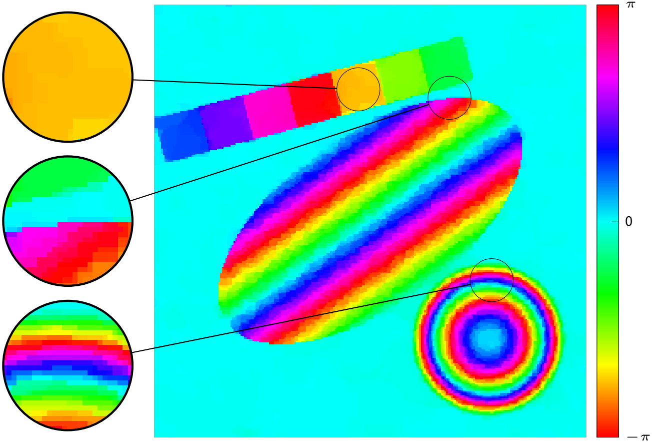
Denoising artificial phase valued data.



noisy data f_{n}, $\sigma = 0.3$

Example

Denoising artificial phase valued data.

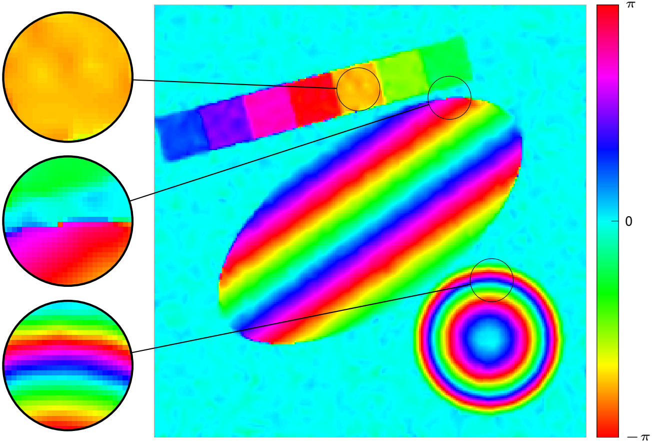


denoising f_n: f_1 with just TV_1

$\alpha_1 = \frac{3}{8}, \alpha_2 = \frac{1}{4}, \beta_1 = \beta_2 = \gamma = 0$: stair casing

Example

Denoising artificial phase valued data.

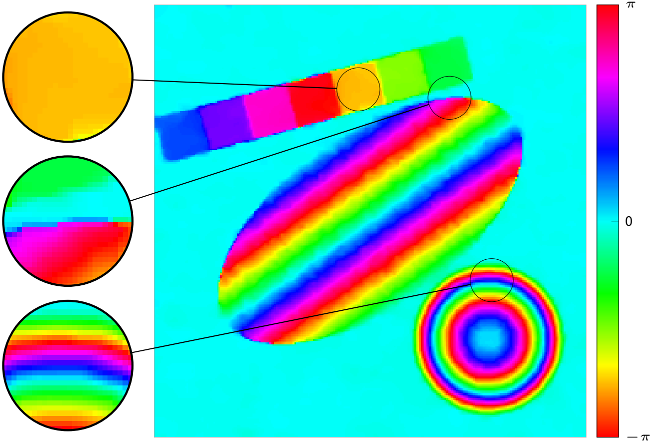


denoising f_n: f_2 with just TV_2

$\alpha_1 = \alpha_2 = 0$, $\beta_1 = \beta_2 = \gamma = \frac{1}{8}$: no plateaus

Example

Denoising artificial phase valued data.



denoising f_n: f_3 with TV_1 & TV_2

$\alpha_1 = \frac{1}{4}$, $\alpha_2 = \beta_1 = \beta_2 = \frac{1}{8}, \gamma = 0$: smallest mean squared error

Convergence of CPPA on $\mathbb{S}^1$

Comparison to $\mathbb{R}$ and challenges.

On $\mathbb{R}$ and Hadamard spaces [Bačák, 2013]
(e.g. Riemannian manifold, non-pos. curv.,simply connected)

- $\sum_{k=0}^{\infty} \lambda_k = \infty$ and $\sum_{k=0}^{\infty} \lambda_k^2 < \infty$

$\Rightarrow$ CPPA on $\mathbb{R}$ converges (weakly) to a global minimizer

- proof uses i.a. convexity of J_i
- How to define convexity on $\mathbb{S}^1$?

Example

For $x_0 \in \mathbb{S}^1$ take

$$f\colon \mathbb{R} \to \mathbb{R}, \quad f(x) := d_1\bigl(x_0, (x)_{2\pi}\bigr).$$

Then f is not convex.

Convergence of CPPA on $\mathbb{S}^1$

With restriction on data f and λ_0.

Theorem (B., Laus, Steidl, Weinmann)

Let $x^{(0)} = f$. And for an $\varepsilon > 0$

- $\mathrm{TV}_1(f) + \mathrm{TV}_2^{hv}(f) + \mathrm{TV}_2^{d}(f) \leq \frac{\varepsilon^2}{\max\{\alpha_1,\alpha_2,\beta_1,\beta_2,\gamma\}}$
- $\max_{i,j} \max\{d_1(f_{i,j}, f_{i,j+1}), d_1(f_{i,j}, f_{i+1,j})\} \leq \frac{\pi}{8}$
- ε, λ_0 *and* $\|\lambda\|_2^2$ *are "small enough"*

Then the CPPA on $\mathbb{S}^1$ converges to a minimizer x^*.

Ideas of the proof:

- "control" $\sum d_1(x^{(k+\frac{i}{c})}, f)$ and from $x_{i,j}^{(k+\frac{i}{c})}$ to its 4-neighborhood
- assure, that this still holds after applying the proximal mappings

$\Rightarrow$ all involved J_i have a convex analogue on $\mathbb{R}$

Denoising of InSAR Data

Measuring earth elevation from radar data.

Synthetic Aperture Radar

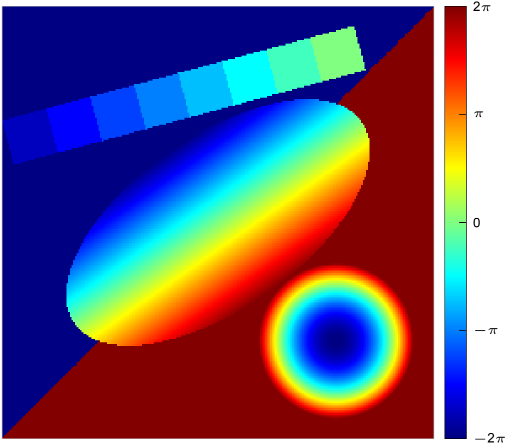
- emit radar & use motion of antenna (i.e. speed of airplane)
- record amplitude and phase of an backscattered signal
- amplitude: reflectivity of the surface
- phase: both elevation and reflection properties
- ! phase of one SAR data rather arbitrary
- record certain area $\Rightarrow$ SAR image

Interferometry

- take two SAR images with different (but known) angles or locations
- $\Rightarrow$ phase difference: principal or wrapped phase
- encodes elevation, but is noisy

Artificial Example

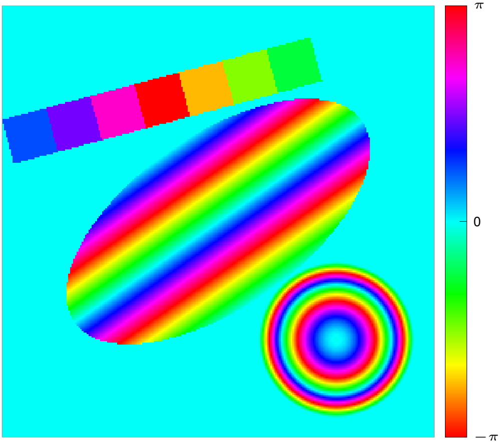
Illustrating the effect of wrapped phase & noise



elevation profile

Artificial Example

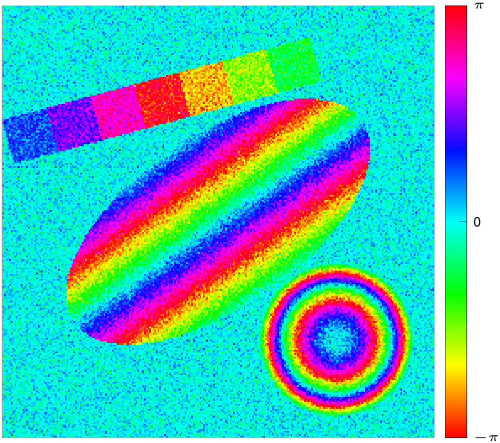
Illustrating the effect of wrapped phase & noise



wrapped phase

Artificial Example

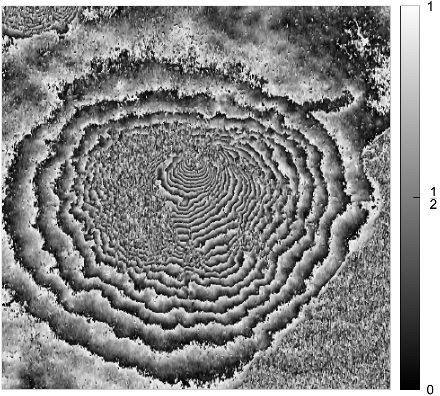
Illustrating the effect of wrapped phase & noise



added noise

Mount Vesuvius

The following image is InSAR data from Mount Vesuvius, Italy.[1]

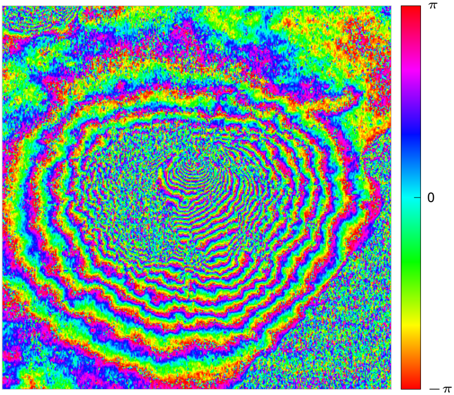


original data, 432×426 pixel

[1] https://earth.esa.int/workshops/ers97/program-details/speeches/rocca-et-al/

Mount Vesuvius

The following image is InSAR data from Mount Vesuvius, Italy.[1]

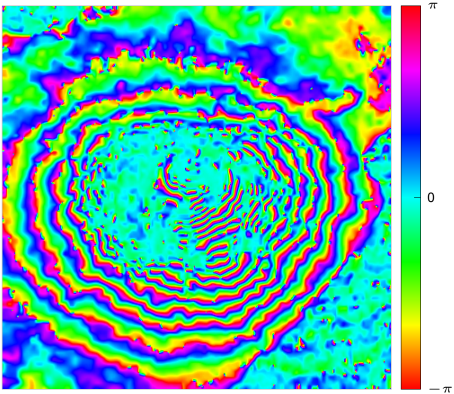


adapted just the coloring

[1] https://earth.esa.int/workshops/ers97/program-details/speeches/rocca-et-al/

Mount Vesuvius

The following image is InSAR data from Mount Vesuvius, Italy.[1]



denoised: $\alpha_1 = \alpha_2 = \frac{1}{4}, \beta_1 = \beta_2 = \gamma = \frac{3}{4}$

[1] https://earth.esa.int/workshops/ers97/program-details/speeches/rocca-et-al/

Conclusion

We derived for $\mathbb{S}^1$-valued 1D & 2D data f

- higher order differences
- proximal mappings for first and second order differences
- higher order TV functional J
- an efficient CPPA to minimize J
- convergence
- application: InSAR data denoising $\Rightarrow$ goal: unwrapping

Future work

- loosen contraints of convergence
- further applications of TV (impainting,...)

Literature

M. Bačák. Computing medians and means in Hadamard spaces. Preprint, 2013.

B., F. Laus, G. Steidl, A. Weinmann Second order differences of cyclic data and applications in variational denoising, in preparation.

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

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

E. Strekalovskiy and D. Cremers. Total cyclic variation and generalizations. J. Math. Imaging Vis., 47(3):258–277, 2013.

A. Weinmann, L. Demaret, and M. Storath. Total variation regularization for manifold-valued data. Preprint, 2013.

Literature

M. Bačák. Computing medians and means in Hadamard spaces. Preprint, 2013.

B., F. Laus, G. Steidl, A. Weinmann Second order differences of cyclic data and applications in variational denoising, in preparation.

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

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

E. Strekalovskiy and D. Cremers. Total cyclic variation and generalizations. J. Math. Imaging Vis., 47(3):258–277, 2013.

A. Weinmann, L. Demaret, and M. Storath. Total variation regularization for manifold-valued data. Preprint, 2013.

Thank you for your attention.