Numerical inpainting techniques
–
Modeling with inherent discretization

Nada Sissouno, Universität Passau

joint work with Tomas Sauer

March 02, 2016

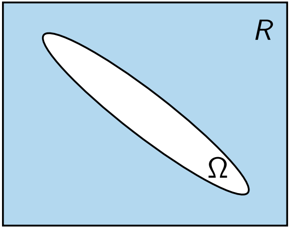
Inpainting: name origin

source: http://www.inrestauro.wordpress.com

Inpainting model

Given:

- domain of image R
- hole $\Omega \subset R$
- data/function g over $R \setminus \Omega$



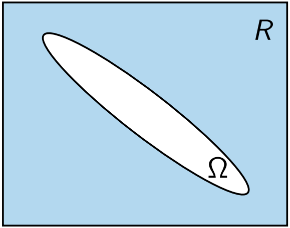
Inpainting model

Given:

- domain of image R
- hole $\Omega \subset R$
- data/function g over $R \setminus \Omega$

Find image u:

- $u = g$ outside of Ω
- $u|_{\Omega}$ "suitable" extension



Inpainting model

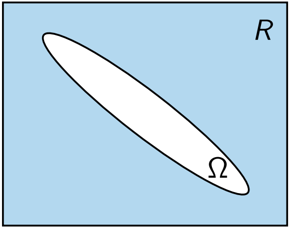
Given:

- domain of image R
- hole $\Omega \subset R$
- data/function g over $R \setminus \Omega$

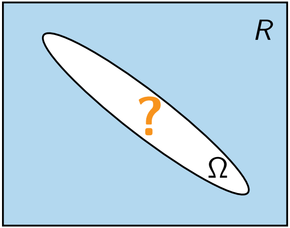
Find image u:

- $u = g$ outside of Ω
- $u|_\Omega$ "suitable" extension

$\Rightarrow$ **interpolation problem**

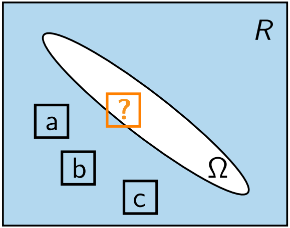


Inpainting methods



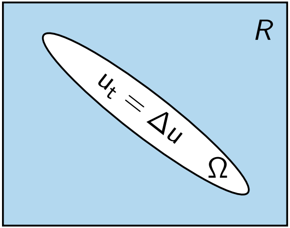
Inpainting methods

- pixel-/patch-based



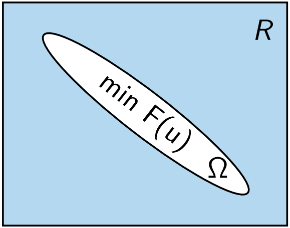
Inpainting methods

- pixel-/patch-based
- PDE-based/variational approaches



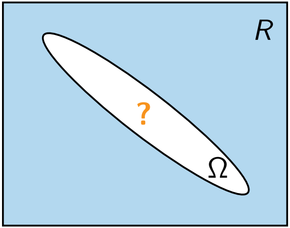
Inpainting methods

- pixel-/patch-based
- PDE-based/variational approaches



Inpainting methods

- pixel-/patch-based
- PDE-based/variational approaches
- combinations of methods
- ...



Inpainting methods

- pixel-/patch-based $\Rightarrow$ textured images
- PDE-based/variational approaches
- combinations of methods
- ...

Inpainting methods

- pixel-/patch-based $\Rightarrow$ textured images
- PDE-based/variational approaches $\Rightarrow$ structured/geometric images
- combinations of methods
- ...

Overview

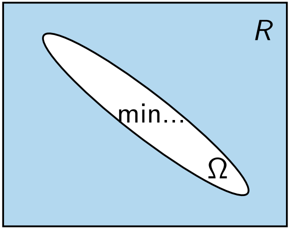
UNIVERSITÄT PASSAU
Fakultät für Informatik und Mathematik

1. Variational approaches
2. Modeling and discretization
3. Implementation
4. Example

Variational approaches

Minimization of

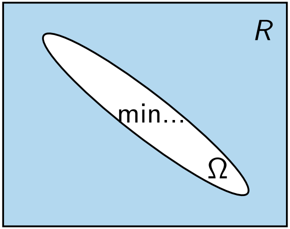
1. constrained problem: $F(u)$ such that $u = g$ on B
2. unconstrained problem: $F(u) + \lambda \|u - g\|_B$



Variational approaches

Minimization of

1. constrained problem: $F(u)$ such that $u = g$ on B
2. unconstrained problem: $F(u) + \lambda \|u - g\|_B$

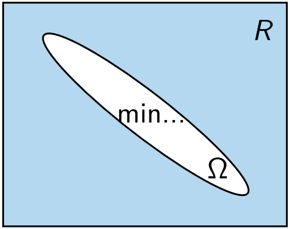


We can choose:

Variational approaches

Minimization of

1. constrained problem: $F(u)$ such that $u = g$ on B
2. unconstrained problem: $F(u) + \lambda \|u - g\|_B$



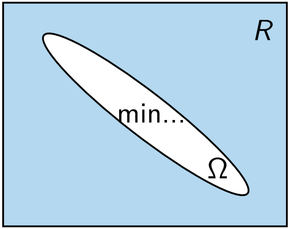
We can choose:

- $B \subseteq R \setminus \Omega$

Variational approaches

Minimization of

1. constrained problem: $F(u)$ such that $u = g$ on B
2. unconstrained problem: $F(u) + \lambda \|u - g\|_B$



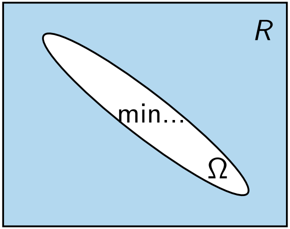
We can choose:

- $B \subseteq R \setminus \Omega$
- $\|\cdot\|_B$

Variational approaches

Minimization of

1. constrained problem: $F(u)$ such that $u = g$ on B
2. unconstrained problem: $F(u) + \lambda \|u - g\|_B$



We can choose:

- $B \subseteq R \setminus \Omega$
- $\|\cdot\|_B$
- $F(u) = \int\limits_{U(\Omega)} \Phi(u)\, d\mathbf{x}$

TV minimization (Chan and Shen[1])

$$F(u) = \int_{U(\Omega)} |\nabla u| \, d\mathbf{x}, \quad u = g \text{ on } B = U(\Omega) \setminus \Omega,$$

with $|\nabla u|^2 = (\partial_1 u)^2 + (\partial_2 u)^2$.

[1] Chan and Shen: "Mathematical models for local non-texture inpaintings", *SIAM Journal of Applied Mathematics*, 2002.

TV minimization (Chan and Shen[1])

$$F(u) = \int\limits_{U(\Omega)} |\nabla u| \, d\mathbf{x}, \quad u = g \text{ on } B = U(\Omega) \setminus \Omega,$$

with $|\nabla u|^2 = (\partial_1 u)^2 + (\partial_2 u)^2$.

- denoising: replace constraint by

$$\frac{1}{|B|} \int_B |u - g|^2 \, d\mathbf{x} = \sigma^2$$

[1] Chan and Shen: "Mathematical models for local non-texture inpaintings", *SIAM Journal of Applied Mathematics*, 2002.

TV approach

TV minimization (Chan and Shen[1])

$$F(u) = \int_{U(\Omega)} |\nabla u| \, d\mathbf{x}, \quad u = g \text{ on } B = U(\Omega) \setminus \Omega,$$

with $|\nabla u|^2 = (\partial_1 u)^2 + (\partial_2 u)^2$.

- denoising: replace constraint by

$$\frac{1}{|B|} \int_B |u - g|^2 \, d\mathbf{x} = \sigma^2$$

- $\Omega = \emptyset$: TV restoration model Rudin, Osher and Fatemi[2]

[1] Chan and Shen: "Mathematical models for local non-texture inpaintings", *SIAM Journal of Applied Mathematics*, 2002.

[2] Rudin et al.: "Nonlinear total variation based noise removal algorithms", *Physica D*, 1992.

TV approach

UNIVERSITÄT PASSAU
Fakultät für Informatik und Mathematik

TV minimization (Chan and Shen[1])

$$F(u) = \int_{U(\Omega)} |\nabla u| \, d\mathbf{x}, \quad u = g \text{ on } B = U(\Omega) \setminus \Omega,$$

with $|\nabla u|^2 = (\partial_1 u)^2 + (\partial_2 u)^2$.

Discretization for image pixel (i, j):

TV approach

TV minimization (Chan and Shen[1])

$$F(u) = \int_{U(\Omega)} |\nabla u| \, d\mathbf{x}, \quad u = g \text{ on } B = U(\Omega) \setminus \Omega,$$

with $|\nabla u|^2 = (\partial_1 u)^2 + (\partial_2 u)^2$.

Discretization for image pixel (i, j):

- $\partial_1 u_{i,j} := u_{i+1,j} - u_{i,j}$
- $\partial_2 u_{i,j} := u_{i,j+1} - u_{i,j}$

TV approach

TV minimization (Chan and Shen[1])

$$F(u) = \int_{U(\Omega)} |\nabla u| \, d\mathbf{x}, \quad u = g \text{ on } B = U(\Omega) \setminus \Omega,$$

with $|\nabla u|^2 = (\partial_1 u)^2 + (\partial_2 u)^2$.

Discretization for image pixel (i, j):

- $\partial_1 u_{i,j} := u_{i+1,j} - u_{i,j}$
- $\partial_2 u_{i,j} := u_{i,j+1} - u_{i,j}$
- $\nabla_{i,j} u := (\partial_1 u_{i,j}, \partial_2 u_{i,j})^T$

Discrete constraint TV inpainting

$$\begin{aligned} \text{minimize} \quad & \sum_{i,j} \|\nabla_{i,j} u\|_2 \\ \text{such that} \quad & u_{i,j} = g_{i,j} \quad \text{on } B \end{aligned}$$

Discretization for image pixel (i, j):

- $\partial_1 u_{i,j} := u_{i+1,j} - u_{i,j}$
- $\partial_2 u_{i,j} := u_{i,j+1} - u_{i,j}$
- $\nabla_{i,j} u := (\partial_1 u_{i,j}, \partial_2 u_{i,j})^T$

TV approach

Discrete constraint TV inpainting

$$\begin{aligned} \text{minimize} \quad & \sum_{i,j} \|\nabla_{i,j} u\|_2 \\ \text{such that} \quad & u_{i,j} = g_{i,j} \quad \text{on } B \end{aligned}$$

Standard approaches: $u \in BV(R)$

TV approach

Discrete constraint TV inpainting

$$\begin{aligned} \text{minimize} \quad & \sum_{i,j} \|\nabla_{i,j} u\|_2 \\ \text{such that} \quad & u_{i,j} = g_{i,j} \quad \text{on } B \end{aligned}$$

Standard approaches: $u \in BV(R)$

Idea: Model problem with tensor product splines

Modeling with tensor product splines

UNIVERSITÄT PASSAU
Fakultät für Informatik und Mathematik

1. Why tensor product splines?

Modeling with tensor product splines

1. Why tensor product splines?
   - explicit expression of function and derivatives

Modeling with tensor product splines

1. Why tensor product splines?
   - explicit expression of function and derivatives
   - small compact support

Modeling with tensor product splines

UNIVERSITÄT PASSAU
Fakultät für Informatik und Mathematik

1. Why tensor product splines?
   - explicit expression of function and derivatives
   - small compact support
   - adaptable to data

Modeling with tensor product splines

1. Why tensor product splines?
   - explicit expression of function and derivatives
   - small compact support
   - adaptable to data
   - . . .

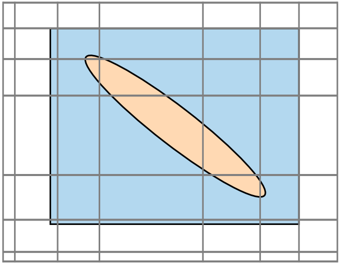
Modeling with tensor product splines

1. Why tensor product splines?
   - explicit expression of function and derivatives
   - small compact support
   - adaptable to data
   - ...
2. How to choose the grid?

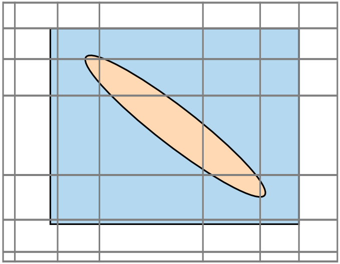
Modeling with tensor product splines

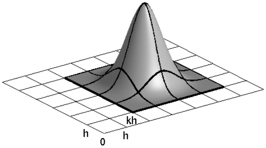
1. Why tensor product splines?
   - explicit expression of function and derivatives
   - small compact support
   - adaptable to data
   - ...
2. How to choose the grid?
3. How to model the problem?

Step 1: How to choose the grid?



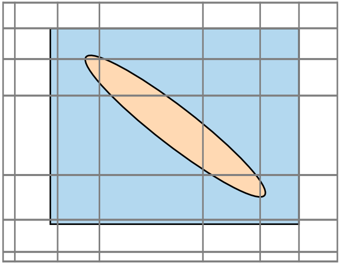
- non-uniform knots $T := T_1 \otimes T_2$
- grid width $\mathbf{h} = (h_1, h_2)$





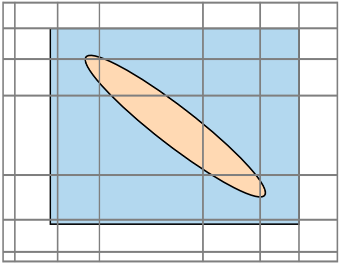
- non-uniform knots $T := T_1 \otimes T_2$
- grid width $\mathbf{h} = (h_1, h_2)$
- order $\mathbf{n} = (n_1, n_2) \in \mathbb{N}^2$
- tensor product B-splines $b_{\mathbf{k}}(\mathbf{x})$, $\mathbf{k} \in K$ and $\mathbf{x} \in R$
- tensor product spline space $\mathcal{S}_{\mathbf{n}}(T, R)$

Step 1: How to choose the grid?



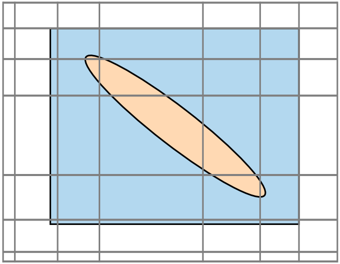
- non-uniform knots T:
  i) stable basis
  ii) no artifacts at ∂R

Step 1: How to choose the grid?



- non-uniform knots T:
  - i) stable basis
  - ii) no artifacts at ∂R

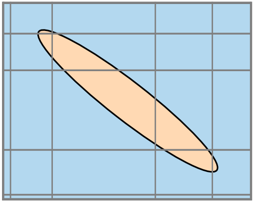
Step 1: How to choose the grid?



- non-uniform knots T:
  - i) stable basis
  - ii) no artifacts at ∂R

$\Rightarrow$ knots on ∂R

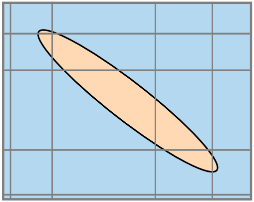
Step 1: How to choose the grid?



- non-uniform knots T:
  - i) stable basis
  - ii) no artifacts at ∂R

$\Rightarrow$ multiple knots on ∂R

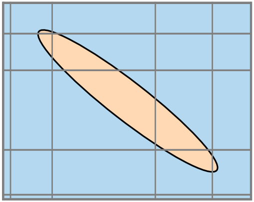
Step 1: How to choose the grid?



- non-uniform knots T:
  - i) stable basis
  - ii) no artifacts at ∂R

  $\Rightarrow$ multiple knots on ∂R

Step 1: How to choose the grid?



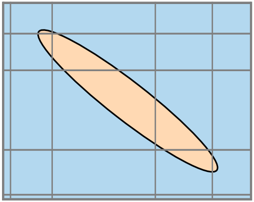
- non-uniform knots T:
  - i) stable basis
  - ii) no artifacts at ∂R

  $\Rightarrow$ multiple knots on ∂R

- tensor product spline $u \in \mathcal{S}_{\mathbf{n}}(T, R)$:

$$u(\mathbf{x}) = \sum_{\mathbf{k} \in K} f_{\mathbf{k}} b_{\mathbf{k}}(\mathbf{x})$$

Step 1: How to choose the grid?



- non-uniform knots T:
  - i) stable basis
  - ii) no artifacts at ∂R

  $\Rightarrow$ multiple knots on ∂R

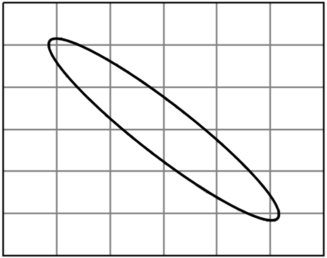
- tensor product spline $u \in \mathcal{S}_{\mathbf{n}}(T, R)$:

$$u(\mathbf{x}) = \sum_{\mathbf{k} \in K} f_{\mathbf{k}} b_{\mathbf{k}}(\mathbf{x})$$

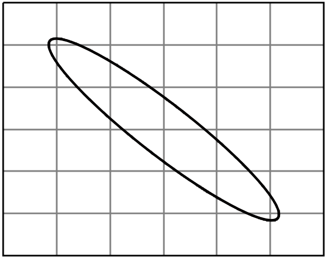
$\Rightarrow$ Determine $\mathbf{f} := (f_{\mathbf{k}})_{\mathbf{k} \in K}$!

Step 2: side conditions

Reproduce image g over $R^* := R \setminus \Omega$

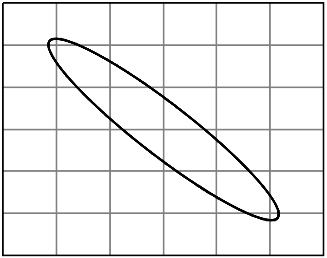


Reproduce image g over $R^* := R \setminus \Omega$ by interpolation:



Step 2: side conditions

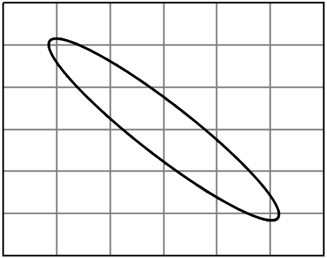
Reproduce image g over $R^* := R \setminus \Omega$ by interpolation:



Spline interpolation at $\mathbf{x_k}$, $\mathbf{k} \in K$:

Step 2: side conditions

Reproduce image g over $R^* := R \setminus \Omega$ by interpolation:

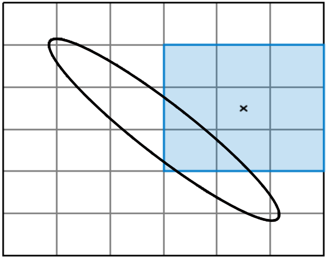


Spline interpolation at $\mathbf{x_k}$, $\mathbf{k} \in K$:

- unique solution if $b_\mathbf{k}(\mathbf{x_k}) > 0$ (Schoenberg-Whitney)

Step 2: side conditions

Reproduce image g over $R^* := R \setminus \Omega$ by interpolation:



order $\mathbf{n} = (3, 3)$

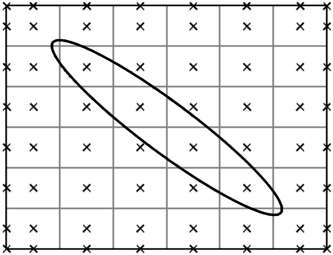
Spline interpolation at $\mathbf{x_k}$, $\mathbf{k} \in K$:

- unique solution if $b_{\mathbf{k}}(\mathbf{x_k}) > 0$ (Schoenberg-Whitney)
- fulfilled for Greville abscissae $\xi_{\mathbf{k}} \in \Xi$, $\mathbf{k} \in K$

Step 2: side conditions

Reproduce image g over $R^* := R \setminus \Omega$ by interpolation:

Greville abscissae



order $\mathbf{n} = (3, 3)$

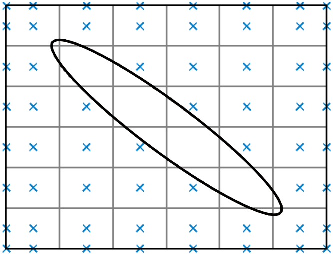
Spline interpolation at $\mathbf{x_k}$, $\mathbf{k} \in K$:

- unique solution if $b_{\mathbf{k}}(\mathbf{x_k}) > 0$ (Schoenberg-Whitney)
- fulfilled for Greville abscissae $\xi_{\mathbf{k}} \in \Xi$, $\mathbf{k} \in K$

Step 2: side conditions

Reproduce image g over $R^* := R \setminus \Omega$ by interpolation:

Interpolation points



order $\mathbf{n} = (3, 3)$

Spline interpolation at $\mathbf{x_k}$, $\mathbf{k} \in K$:

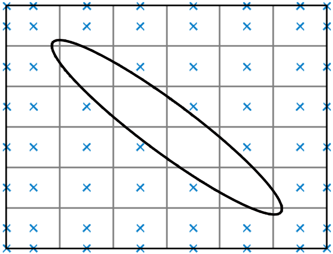
- unique solution if $b_\mathbf{k}(\mathbf{x_k}) > 0$
  (Schoenberg-Whitney)
- fulfilled for Greville abscissae
  $\xi_\mathbf{k} \in \Xi$, $\mathbf{k} \in K$
- restriction to R^*
  $\xi_\mathbf{k} \in \Xi^* := \Xi \cap R^*$, $\mathbf{k} \in K_{\Xi^*}$

Step 2: side conditions

Reproduce image g over $R^* := R \setminus \Omega$ by interpolation:

$$u(\xi_{\mathbf{k}}) = g(\xi_{\mathbf{k}}), \quad \xi_{\mathbf{k}} \in \Xi^*.$$

Interpolation points



order $\mathbf{n} = (3, 3)$

Spline interpolation at $\mathbf{x_k}$, $\mathbf{k} \in K$:

- unique solution if $b_{\mathbf{k}}(\mathbf{x_k}) > 0$ (Schoenberg-Whitney)
- fulfilled for Greville abscissae $\xi_{\mathbf{k}} \in \Xi$, $\mathbf{k} \in K$
- restriction to R^* $\xi_{\mathbf{k}} \in \Xi^* := \Xi \cap R^*$, $\mathbf{k} \in K_{\Xi^*}$

Step 3: TV minimization

UNIVERSITÄT PASSAU
Fakultät für Informatik und Mathematik

Minimization over u:

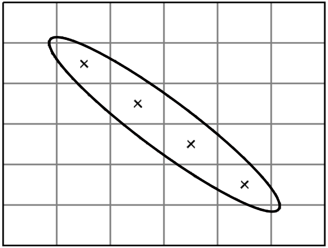
$$F(u) = \int |\nabla u| \, d\mathbf{x}$$

Step 3: TV minimization

Minimization over u:

$$F(u) = \int |\nabla u| \, d\mathbf{x}$$

Greville abscissae



order $\mathbf{n} = (3, 3)$

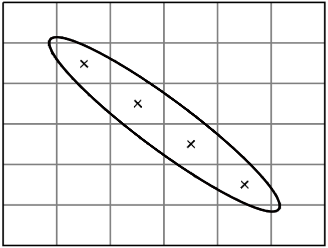
- Greville abscissae $\boldsymbol{\xi}_\mathbf{k} \in \Omega$

Step 3: TV minimization

Minimization over u:

$$F(u) = \int_{U_T^{\mathbf{n}}(\Omega)} |\nabla u| \, d\mathbf{x}$$

Greville abscissae



order $\mathbf{n} = (3,3)$

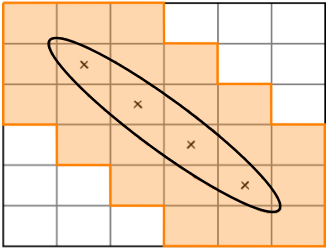
- Greville abscissae $\xi_{\mathbf{k}} \in \Omega$
- domain of integration:

$$U_T^{\mathbf{n}}(\Omega) := \bigcup \{\text{supp } b_{\mathbf{k}} | \; \xi_{\mathbf{k}} \in \Omega\}$$

Step 3: TV minimization

Minimization over u:

$$F(u) = \int_{U_T^{\mathbf{n}}(\Omega)} |\nabla u| \, d\mathbf{x}$$



order $\mathbf{n} = (3, 3)$

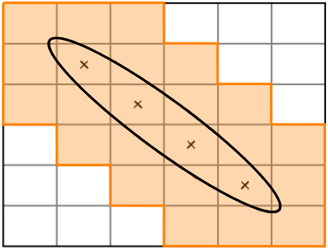
- Greville abscissae $\xi_{\mathbf{k}} \in \Omega$
- domain of integration:

$$U_T^{\mathbf{n}}(\Omega) := \bigcup \{\text{supp}\, b_{\mathbf{k}} | \; \xi_{\mathbf{k}} \in \Omega\}$$

Step 3: TV minimization

Minimization over u:

$$F(u) = \int_{U_T^{\mathbf{n}}(\Omega)} |\nabla u| \, d\mathbf{x}$$



order $\mathbf{n} = (3, 3)$

- Greville abscissae $\xi_{\mathbf{k}} \in \Omega$
- domain of integration:

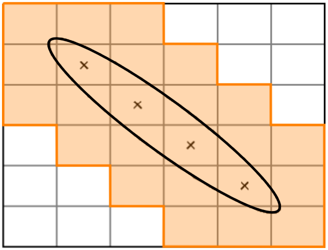
  $$U_T^{\mathbf{n}}(\Omega) := \bigcup \{\text{supp } b_{\mathbf{k}} | \, \xi_{\mathbf{k}} \in \Omega\}$$

  - union of grid cells Z

Step 3: TV minimization

Minimization over u:

$$F(u) = \sum_{Z \subset U_T^{\mathbf{n}}(\Omega)} \left(\int_Z |\nabla u| \, d\mathbf{x} \right)$$



order $\mathbf{n} = (3, 3)$

- Greville abscissae $\xi_{\mathbf{k}} \in \Omega$
- domain of integration:

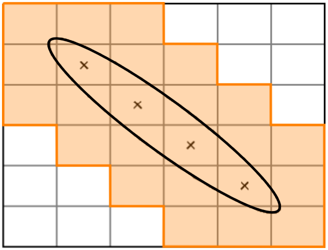
$$U_T^{\mathbf{n}}(\Omega) := \bigcup \{\text{supp } b_{\mathbf{k}} | \ \xi_{\mathbf{k}} \in \Omega\}$$

  - union of grid cells Z

Step 3: TV minimization

Minimization over u:

$$F(u) = \sum_{Z \subset U_T^{\mathbf{n}}(\Omega)} \left(\int_Z |\nabla u| \, d\mathbf{x} \right)$$



order $\mathbf{n} = (3, 3)$

- Greville abscissae $\xi_{\mathbf{k}} \in \Omega$
- domain of integration:

  $$U_T^{\mathbf{n}}(\Omega) := \bigcup \{\mathrm{supp}\, b_{\mathbf{k}} | \, \xi_{\mathbf{k}} \in \Omega\}$$

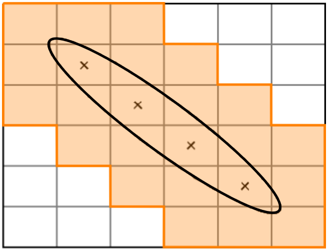
  - union of grid cells Z
- partial derivative:

  $$\partial_j u(\mathbf{x}) = \sum_{\mathbf{k}} f_{\mathbf{k}} \, \partial_j b_{\mathbf{k}}(\mathbf{x})$$

Step 3: TV minimization

Minimization over $\mathbf{f}$:

$$F(u) = \sum_{Z \subset U_T^{\mathbf{n}}(\Omega)} \left(\int_Z |\nabla u| \, d\mathbf{x} \right)$$



order $\mathbf{n} = (3, 3)$

- Greville abscissae $\xi_{\mathbf{k}} \in \Omega$
- domain of integration:

$$U_T^{\mathbf{n}}(\Omega) := \bigcup \{\text{supp } b_{\mathbf{k}} | \ \xi_{\mathbf{k}} \in \Omega\}$$

  - union of grid cells Z

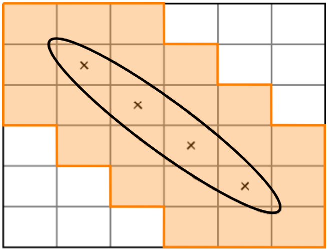
- partial derivative:

$$\partial_j u(\mathbf{x}) = \sum_{\mathbf{k}} f_{\mathbf{k}} \, \partial_j b_{\mathbf{k}}(\mathbf{x})$$

Step 3: TV minimization

Minimization over $\mathbf{f}$:

$$F(u) = \sum_{Z \subset U_T^{\mathbf{n}}(\Omega)} \left(\int_Z |\nabla u| \, d\mathbf{x} \right)$$



order $\mathbf{n} = (3, 3)$

- Greville abscissae $\xi_{\mathbf{k}} \in \Omega$
- domain of integration:

$$U_T^{\mathbf{n}}(\Omega) := \bigcup \{\operatorname{supp} b_{\mathbf{k}} | \ \xi_{\mathbf{k}} \in \Omega\}$$

  - union of grid cells Z
  - coordinate-wise Gauss quadrature
- partial derivative:

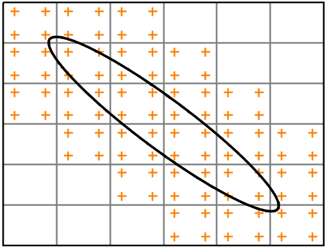
$$\partial_j u(\mathbf{x}) = \sum_{\mathbf{k}} f_{\mathbf{k}} \, \partial_j b_{\mathbf{k}}(\mathbf{x})$$

Step 3: TV minimization

Minimization over $\mathbf{f}$:

$$F(u) = \sum_{Z \subset U_T^{\mathbf{n}}(\Omega)} \left(\int_Z |\nabla u| \, d\mathbf{x} \right)$$

quadrature points



order $\mathbf{n} = (3, 3)$

- Greville abscissae $\xi_{\mathbf{k}} \in \Omega$
- domain of integration:

$$U_T^{\mathbf{n}}(\Omega) := \bigcup \{\text{supp}\, b_{\mathbf{k}} |\ \xi_{\mathbf{k}} \in \Omega\}$$

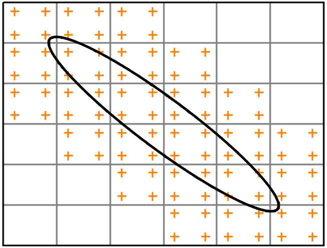
  - union of grid cells Z
  - coordinate-wise Gauss quadrature
- partial derivative:

$$\partial_j u(\mathbf{x}) = \sum_{\mathbf{k}} f_{\mathbf{k}} \, \partial_j b_{\mathbf{k}}(\mathbf{x})$$

Minimization over **f**:

$$F(u) = \sum_{Z \subset U_T^{\mathbf{n}}(\Omega)} \left(\int_Z |\nabla u| \, d\mathbf{x} \right) \approx \sum_{\boldsymbol{\theta} \in \Theta} w_{\boldsymbol{\theta}} \|\nabla u(\boldsymbol{\theta})\|_2$$

quadrature points



order $\mathbf{n} = (3, 3)$

- Greville abscissae $\boldsymbol{\xi}_{\mathbf{k}} \in \Omega$
- domain of integration:

$$U_T^{\mathbf{n}}(\Omega) := \bigcup \{\operatorname{supp} b_{\mathbf{k}} | \; \boldsymbol{\xi}_{\mathbf{k}} \in \Omega\}$$

  - union of grid cells Z
  - coordinate-wise Gauss quadrature
- partial derivative:

$$\partial_j u(\theta) = \sum_{\mathbf{k}} f_{\mathbf{k}} \, \partial_j b_{\mathbf{k}}(\theta)$$

Tensor product idea

1. inpainting function:

$$u(\mathbf{x}) = \sum_{\mathbf{k}\in K} f_{\mathbf{k}} b_{\mathbf{k}}(\mathbf{x}),\ \mathbf{x} \in R$$

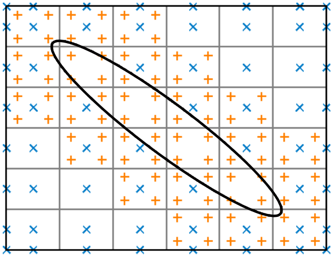
2. side condition:

$$u(\boldsymbol{\xi}) = g(\boldsymbol{\xi})\ ,\ \forall \boldsymbol{\xi} \in \Xi^*$$

3. TV minimization:

$$\min_{\mathbf{f}} \sum_{\boldsymbol{\theta}\in\Theta} w_{\boldsymbol{\theta}} \|\nabla u(\boldsymbol{\theta})\|_2$$

interpolation and quadrature points



order $\mathbf{n} = (3, 3)$

Tensor product idea

1. inpainting function:

$$u(\mathbf{x}) = \sum_{\mathbf{k}\in K} f_{\mathbf{k}} b_{\mathbf{k}}(\mathbf{x}),\ \mathbf{x} \in R$$

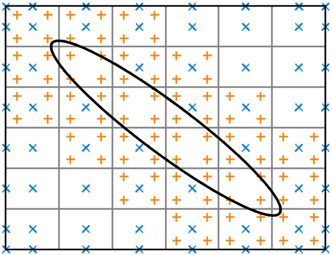
2. side condition:

$$u(\boldsymbol{\xi}) = g(\boldsymbol{\xi})\ ,\ \forall \boldsymbol{\xi} \in \Xi^*$$

3. TV minimization:

$$\min_{\mathbf{f}} \sum_{\boldsymbol{\theta}\in\Theta} w_{\boldsymbol{\theta}} \|\nabla u(\boldsymbol{\theta})\|_2$$

interpolation and quadrature points



order $\mathbf{n} = (3, 3)$

How to solve this?

Second order cone program

Definition SOCP[1]

Determine $\mathbf{x} \in \mathbb{R}^n$ by

$$\begin{aligned} \text{minimize} \quad & \mathbf{t}^T \mathbf{x} \\ \text{such that} \quad & \|\mathbf{A}_i \mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T \mathbf{x} + d_i, \quad i = 1, \ldots, N \end{aligned}$$

with parameters $\mathbf{t}, \mathbf{c}_i \in \mathbb{R}^n$, $\mathbf{b}_i \in \mathbb{R}^{n_i - 1}$, $d_i \in \mathbb{R}$ and $\mathbf{A}_i \in \mathbb{R}^{(n_i - 1) \times n}$.

[1] Lobo et al.: "Applications of second-order cone programming", *Linear Algebra and its Applications*, 1998.

Definition SOCP[1]

Determine $\mathbf{x} \in \mathbb{R}^n$ by

$$\begin{array}{ll} \text{minimize} & \mathbf{t}^T\mathbf{x} \\ \text{such that} & \|\mathbf{A}_i\mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i, \quad i = 1, \ldots, N \end{array}$$

with parameters $\mathbf{t}, \mathbf{c}_i \in \mathbb{R}^n$, $\mathbf{b}_i \in \mathbb{R}^{n_i-1}$, $d_i \in \mathbb{R}$ and $\mathbf{A}_i \in \mathbb{R}^{(n_i-1)\times n}$.

- second order (convex) cone of dimension n_i:

$$\mathcal{K}_{n_i} := \left\{ \left(\begin{smallmatrix}\mathbf{y}\\ s\end{smallmatrix}\right) \middle| \, \mathbf{y} \in \mathbb{R}^{n_i-1}, \, s \in \mathbb{R}, \|\mathbf{y}\|_2 \leq s \right\}$$

[1] Lobo et al.: "Applications of second-order cone programming", *Linear Algebra and its Applications*, 1998.

Definition SOCP[1]

Determine $\mathbf{x} \in \mathbb{R}^n$ by

$$\begin{aligned} &\text{minimize} && \mathbf{t}^T\mathbf{x} \\ &\text{such that} && \|\mathbf{A}_i\mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i, \quad i = 1, \dots, N \end{aligned}$$

with parameters $\mathbf{t}, \mathbf{c}_i \in \mathbb{R}^n$, $\mathbf{b}_i \in \mathbb{R}^{n_i-1}$, $d_i \in \mathbb{R}$ and $\mathbf{A}_i \in \mathbb{R}^{(n_i-1)\times n}$.

- second order (convex) cone of dimension n_i:

$$\mathcal{K}_{n_i} := \left\{ \left(\begin{smallmatrix}\mathbf{y}\\ s\end{smallmatrix}\right) \middle| \, \mathbf{y} \in \mathbb{R}^{n_i-1}, s \in \mathbb{R}, \|\mathbf{y}\|_2 \leq s \right\}$$

- second order cone constraint of dimension n_i:

$$\|\mathbf{A}_i\mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i \quad \Longleftrightarrow \quad \begin{pmatrix}\mathbf{A}_i \\ \mathbf{c}_i^T\end{pmatrix}\mathbf{x} + \begin{pmatrix}\mathbf{b}_i \\ d_i\end{pmatrix} \in \mathcal{K}_{n_i}$$

[1] Lobo et al.: "Applications of second-order cone programming", *Linear Algebra and its Applications*, 1998.

SOCP for spline inpainting

UNIVERSITÄT PASSAU
Fakultät für Informatik und Mathematik

SOCP

$$\min_{\mathbf{x}} \quad \mathbf{t}^T\mathbf{x}$$
$$\text{s.t.} \quad \|\mathbf{A}_i\mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i$$

Discrete spline inpainting

$$\min_{\mathbf{f}} \quad \sum_{\boldsymbol{\theta}\in\Theta} w_{\boldsymbol{\theta}}\|\nabla u(\boldsymbol{\theta})\|_2$$
$$\text{s.t.} \quad u(\boldsymbol{\xi}) = g(\boldsymbol{\xi}),\ \forall\boldsymbol{\xi} \in \Xi^*$$

SOCP for spline inpainting

SOCP

$$\min_{\mathbf{x}} \quad \mathbf{t}^T\mathbf{x}$$
$$\text{s.t.} \quad \|\mathbf{A}_i\mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i$$

Discrete spline inpainting

$$\min_{\mathbf{f}} \quad \sum_{\boldsymbol{\theta}\in\Theta} w_{\boldsymbol{\theta}} \|\nabla u(\boldsymbol{\theta})\|_2$$
$$\text{s.t.} \quad u(\xi) = g(\xi),\ \forall \xi \in \Xi^*$$

SOCP for spline inpainting

For some auxiliary variables $u_{\boldsymbol{\theta}}$, $\boldsymbol{\theta} \in \Theta$,

$$w_{\boldsymbol{\theta}} \|\nabla u(\boldsymbol{\theta})\|_2 \leq u_{\boldsymbol{\theta}}, \quad \forall \boldsymbol{\theta} \in \Theta,$$

SOCP for spline inpainting

SOCP

$$\min_{\mathbf{x}} \quad \mathbf{t}^T\mathbf{x}$$
$$\text{s.t.} \quad \|\mathbf{A}_i\mathbf{x}+\mathbf{b}_i\|_2 \le \mathbf{c}_i^T\mathbf{x}+d_i$$

Discrete spline inpainting

$$\min_{\mathbf{f}} \quad \sum_{\boldsymbol{\theta}\in\Theta} w_{\boldsymbol{\theta}}\|\nabla u(\boldsymbol{\theta})\|_2$$
$$\text{s.t.} \quad u(\xi)=g(\xi),\ \forall\xi\in\Xi^*$$

SOCP for spline inpainting

For some auxiliary variables $u_{\boldsymbol{\theta}}$, $\boldsymbol{\theta}\in\Theta$, determine

$$\min_{u_{\boldsymbol{\theta}}} \quad \sum_{\boldsymbol{\theta}\in\Theta} u_{\boldsymbol{\theta}}$$
$$\text{such that} \quad w_{\boldsymbol{\theta}}\|\nabla u(\boldsymbol{\theta})\|_2 \le u_{\boldsymbol{\theta}}, \quad \forall\boldsymbol{\theta}\in\Theta,$$

SOCP for spline inpainting

SOCP

$$\min_{\mathbf{x}} \quad \mathbf{t}^T\mathbf{x}$$
$$\text{s.t.} \quad \|\mathbf{A}_i\mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i$$

Discrete spline inpainting

$$\min_{\mathbf{f}} \quad \sum_{\boldsymbol{\theta}\in\Theta} w_{\boldsymbol{\theta}} \|\nabla u(\boldsymbol{\theta})\|_2$$
$$\text{s.t.} \quad u(\boldsymbol{\xi}) = g(\boldsymbol{\xi}),\ \forall \boldsymbol{\xi} \in \Xi^*$$

SOCP for spline inpainting

For some auxiliary variables $u_{\boldsymbol{\theta}}$, $\boldsymbol{\theta} \in \Theta$, determine

$$\min_{u_{\boldsymbol{\theta}}} \quad \sum_{\boldsymbol{\theta}\in\Theta} u_{\boldsymbol{\theta}}$$

$$\text{such that} \quad w_{\boldsymbol{\theta}} \|\nabla u(\boldsymbol{\theta})\|_2 \leq u_{\boldsymbol{\theta}}, \quad \forall \boldsymbol{\theta} \in \Theta,$$

$$\text{and} \quad u(\boldsymbol{\xi}) = g(\boldsymbol{\xi}), \quad \forall \boldsymbol{\xi} \in \Xi^*.$$

SOCP for spline inpainting

SOCP

$$\min_{\mathbf{x}} \quad \mathbf{t}^T\mathbf{x}$$
$$\text{s.t.} \quad \|\mathbf{A}_i\mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i$$

Discrete spline inpainting

$$\min_{\mathbf{f}} \quad \sum_{\boldsymbol{\theta}\in\Theta} w_{\boldsymbol{\theta}}\|\nabla u(\boldsymbol{\theta})\|_2$$
$$\text{s.t.} \quad u(\boldsymbol{\xi}) = g(\boldsymbol{\xi}),\ \forall \boldsymbol{\xi} \in \Xi^*$$

SOCP for spline inpainting

For some auxiliary variables $u_{\boldsymbol{\theta}}$, $\boldsymbol{\theta} \in \Theta$, determine

$$\min_{u_{\boldsymbol{\theta}}} \quad \sum_{\boldsymbol{\theta}\in\Theta} u_{\boldsymbol{\theta}}$$
$$\text{such that} \quad w_{\boldsymbol{\theta}}\|\nabla u(\boldsymbol{\theta})\|_2 \leq u_{\boldsymbol{\theta}}, \quad \forall \boldsymbol{\theta} \in \Theta,$$
$$\text{and} \quad u(\boldsymbol{\xi}) = g(\boldsymbol{\xi}), \quad \forall \boldsymbol{\xi} \in \Xi^*.$$

$$u(\mathbf{x}) = \sum_{\mathbf{k}} f_{\mathbf{k}} b_{\mathbf{k}}(\mathbf{x}) \quad \text{and} \quad \partial_j u(\mathbf{x}) = \sum_{\mathbf{k}} f_{\mathbf{k}}\, \partial_j b_{\mathbf{k}}(\mathbf{x})$$

SOCP for spline inpainting

SOCP

$$\min_{\mathbf{x}} \quad \mathbf{t}^T\mathbf{x}$$
$$\text{s.t.} \quad \|\mathbf{A}_i\mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i$$

Discrete spline inpainting

$$\min_{\mathbf{f}} \quad \sum_{\boldsymbol{\theta}\in\Theta} w_{\boldsymbol{\theta}} \|\nabla u(\boldsymbol{\theta})\|_2$$
$$\text{s.t.} \quad u(\boldsymbol{\xi}) = g(\boldsymbol{\xi}),\ \forall \boldsymbol{\xi} \in \Xi^*$$

SOCP for spline inpainting

For some auxiliary variables $u_{\boldsymbol{\theta}}$, $\boldsymbol{\theta} \in \Theta$, determine

$$\min_{u_{\boldsymbol{\theta}},\mathbf{f}} \quad \sum_{\boldsymbol{\theta}\in\Theta} u_{\boldsymbol{\theta}}$$

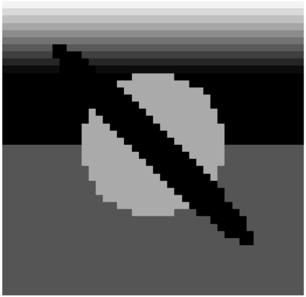
$$\text{such that} \quad w_{\boldsymbol{\theta}} \|\nabla u(\boldsymbol{\theta})\|_2 \leq u_{\boldsymbol{\theta}}, \quad \forall \boldsymbol{\theta} \in \Theta,$$

$$\text{and} \quad u(\boldsymbol{\xi}) = g(\boldsymbol{\xi}), \quad \forall \boldsymbol{\xi} \in \Xi^*.$$

$$u(\mathbf{x}) = \sum_{\mathbf{k}} f_{\mathbf{k}} b_{\mathbf{k}}(\mathbf{x}) \quad \text{and} \quad \partial_j u(\mathbf{x}) = \sum_{\mathbf{k}} f_{\mathbf{k}}\, \partial_j b_{\mathbf{k}}(\mathbf{x})$$

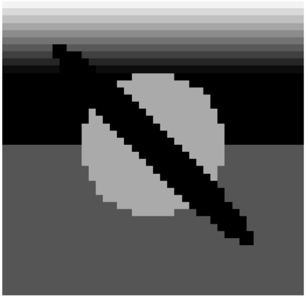
Example: interpolation

Image with gap

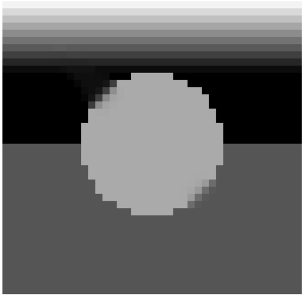


Example: interpolation

Image with gap

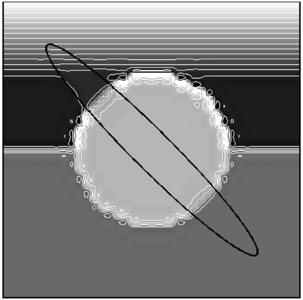


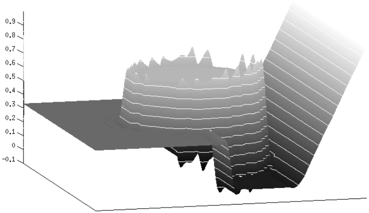
Spline image



Example: interpolation

Spline function





Quasi-interpolants

Quasi-interpolant of order $\boldsymbol{\nu}$

Linear map $Q : \mathcal{C}(R) \to \mathcal{S}_{\mathbf{n}}(T, R)$ such that

$$Qg = \sum_{\mathbf{k}} \lambda_{\mathbf{k}}(g) b_{\mathbf{k}} \text{ and } Qp = p \ \forall p \in \mathbb{P}_{\boldsymbol{\nu}}$$

with uniformly bounded $\lambda_{\mathbf{k}} : \mathcal{C}(R) \to \mathbb{R}$ given by

$$\lambda_{\mathbf{k}}(g) = \sum_{\mathbf{i}(\mathbf{k})} q_{\mathbf{i}(\mathbf{k})}\, g(t_{\mathbf{i}(\mathbf{k})}) \quad \text{for } t_{\mathbf{i}(\mathbf{k})} \in U(\operatorname{supp} b_{\mathbf{k}})$$

Example of order 2: Schoenberg quasi-interpolant

UNIVERSITÄT PASSAU
Fakultät für Informatik und Mathematik

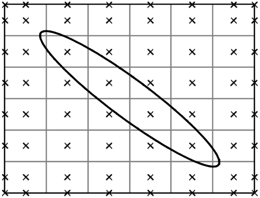
Example of order 2: Schoenberg quasi-interpolant

$$\lambda_{\mathbf{k}}(g) := g(\boldsymbol{\xi}_{\mathbf{k}}), \text{ for Greville abscissae } \boldsymbol{\xi}_{\mathbf{k}} \in R.$$

Example of order 2: Schoenberg quasi-interpolant

$$\lambda_{\mathbf{k}}(g) := g(\boldsymbol{\xi}_{\mathbf{k}}), \text{ for Greville abscissae } \boldsymbol{\xi}_{\mathbf{k}} \in R.$$

Greville abscissae



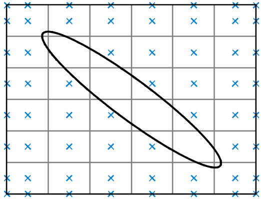
order $\mathbf{n} = (3, 3)$

Example of order 2: Schoenberg quasi-interpolant

$$\lambda_{\mathbf{k}}(g) := g(\boldsymbol{\xi}_{\mathbf{k}}), \text{ for Greville abscissae } \boldsymbol{\xi}_{\mathbf{k}} \in R.$$

Inpainting problem:

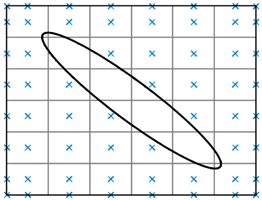
Interpolation points



order $\mathbf{n} = (3, 3)$

Example of order 2: Schoenberg quasi-interpolant

$$\lambda_{\mathbf{k}}(g) := g(\boldsymbol{\xi}_{\mathbf{k}}), \text{ for Greville abscissae } \boldsymbol{\xi}_{\mathbf{k}} \in R.$$



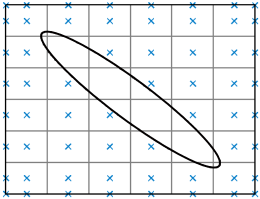
Inpainting problem:

- side condition: $\lambda_{\mathbf{k}}(g) = g(\boldsymbol{\xi}_{\mathbf{k}})$, $\boldsymbol{\xi}_{\mathbf{k}} \in R^*$

Example of order 2: Schoenberg quasi-interpolant

$$\lambda_{\mathbf{k}}(g) := g(\boldsymbol{\xi}_{\mathbf{k}}), \text{ for Greville abscissae } \boldsymbol{\xi}_{\mathbf{k}} \in R.$$

Interpolation points



order $\mathbf{n} = (3, 3)$

Inpainting problem:

- side condition: $\lambda_{\mathbf{k}}(g) = g(\boldsymbol{\xi}_{\mathbf{k}}), \boldsymbol{\xi}_{\mathbf{k}} \in R^*$
- minimization $\Rightarrow \lambda_{\mathbf{k}}(g)$ for $\boldsymbol{\xi}_{\mathbf{k}} \in \Omega$

Quasi-interpolants for spline inpainting

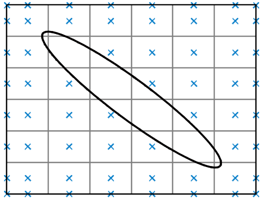
Example of order 2: Schoenberg quasi-interpolant

$$\lambda_{\mathbf{k}}(g) := g(\xi_{\mathbf{k}}),\ \text{for Greville abscissae } \xi_{\mathbf{k}} \in R.$$

Inpainting problem:

- side condition: $\lambda_{\mathbf{k}}(g) = g(\xi_{\mathbf{k}}),\ \xi_{\mathbf{k}} \in R^*$
- minimization $\Rightarrow \lambda_{\mathbf{k}}(g)$ for $\xi_{\mathbf{k}} \in \Omega$

Interpolation points



order $\mathbf{n} = (3,3)$

Quasi-interpolant

$$Q(g) = \sum_{\mathbf{k}} \lambda_{\mathbf{k}}(g) b_{\mathbf{k}}$$

with $\lambda_{\mathbf{k}}(g) := \sum_{\mathbf{i(k)}} q_{\mathbf{i(k)}}\, g(t_{\mathbf{i(k)}})$

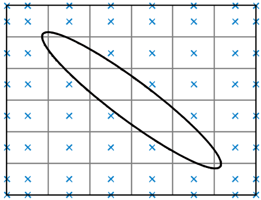
Quasi-interpolants for spline inpainting

UNIVERSITÄT PASSAU
Fakultät für Informatik und Mathematik

Example of order 2: Schoenberg quasi-interpolant

$$\lambda_{\mathbf{k}}(g) := g(\boldsymbol{\xi}_{\mathbf{k}}),\ \text{for Greville abscissae } \boldsymbol{\xi}_{\mathbf{k}} \in R.$$

Interpolation points



order $\mathbf{n} = (3, 3)$

Inpainting problem:

- side condition: $\lambda_{\mathbf{k}}(g) = g(\boldsymbol{\xi}_{\mathbf{k}}),\ \boldsymbol{\xi}_{\mathbf{k}} \in R^*$
- minimization $\Rightarrow \lambda_{\mathbf{k}}(g)$ for $\boldsymbol{\xi}_{\mathbf{k}} \in \Omega$

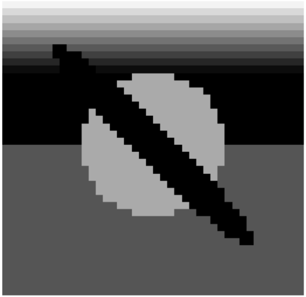
Quasi-interpolant for image g

$$Q(g) = \sum_{\mathbf{j}} g(t_{\mathbf{j}})\tilde{b}_{\mathbf{j}}$$

with $\tilde{b}_{\mathbf{j}} := \sum_{\mathbf{i(j)}} \tilde{q}_{\mathbf{i(j)}}\, b_{\mathbf{i(j)}}$.

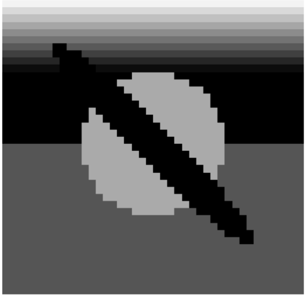
Example: quasi-interpolant

Image with gap

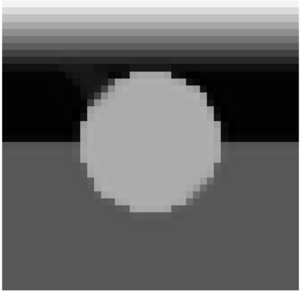


Example: quasi-interpolant

Image with gap



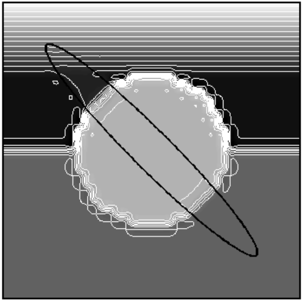
Spline image



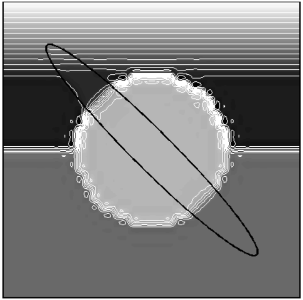
Example: quasi-interpolant

Spline functions

Quasi-interpolant



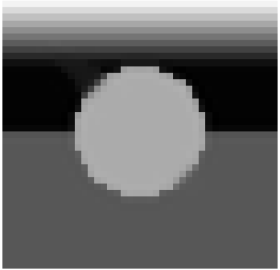
Interpolant



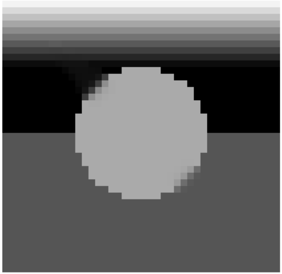
Example: comparison with classical TV

Inpainting results

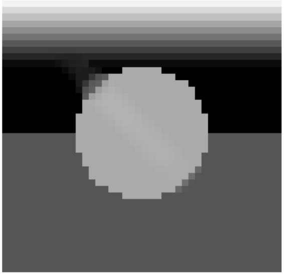
Quasi-interpolant



Interpolant



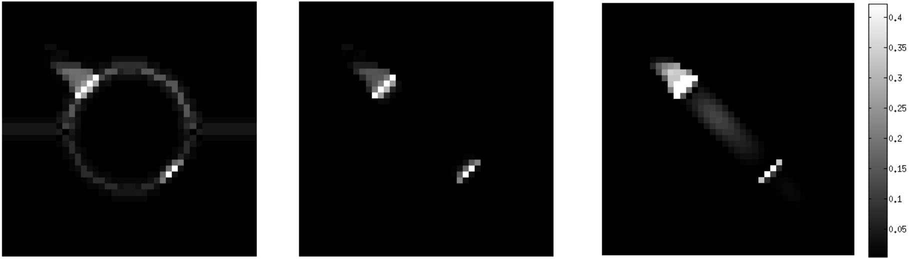
TV



Example: comparison with classical TV

Error of inpainting results

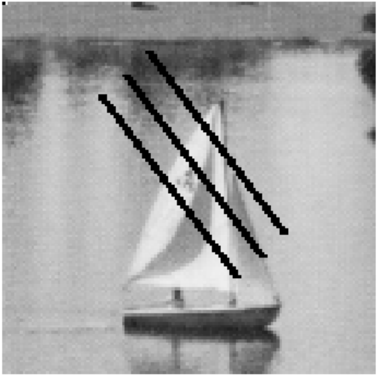
Quasi-interpolant | Interpolant | TV



	max error	mse	$\geq 1/256$	≥ 0.1
Quasi-interpolant	0.4194	$4.9 \cdot 10^{-04}$	318px	10px
Interpolant	0.4229	$4 \cdot 10^{-04}$	57px	9px
TV	$\sim$0.3456	$\sim 5.6 \cdot 10^{-04}$	$\sim$136px	$\sim$17px

Example: sailboat

Image with scratch



Example: sailboat

Inpainting results

Quasi-interpolant

Interpolant

TV

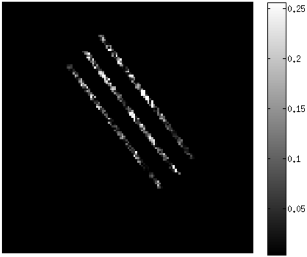
Example: sailboat

Error of inpainting results

Quasi-interpolant

Interpolant

TV



	max error	mse	$\geq 1/256$	≥ 0.1
Quasi-interpolant	0.2257	$7.5 \cdot 10^{-05}$	5962px	13px
Interpolant	0.2171	$5.1 \cdot 10^{-05}$	496px	13px
TV	$\sim$0.2141	$\sim 6.1 \cdot 10^{-05}$	$\sim$516px	$\sim$17px

Summary

Tensor product spline inpainting:

- grid with multiple knots on boundary $\Rightarrow$ stable basis
- (quasi-)interpolation at Greville abscissae $\Rightarrow$ Schoenberg-Whitney
- minimization over union of grid cells $\Rightarrow$ Gauss quadrature
- spline and derivatives with same coefficients $\mathbf{f}$ $\Rightarrow$ optimization w.r.t. $\mathbf{f}$

Summary

Tensor product spline inpainting:

- grid with multiple knots on boundary $\Rightarrow$ stable basis
- (quasi-)interpolation at Greville abscissae $\Rightarrow$ Schoenberg-Whitney
- minimization over union of grid cells $\Rightarrow$ Gauss quadrature
- spline and derivatives with same coefficients $\mathbf{f}$ $\Rightarrow$ optimization w.r.t. $\mathbf{f}$

Some possible modifications:

- other functionals
- 2 step method for optimization
- iterative solver
- adapt grid or basis

Thank you for your attention!