Nonlocal Inpainting of Manifold-valued Data on Finite Weighted Graphs

Ronny Bergmann[a]
University of Kaiserslautern

3rd Conference on Geometric Science of Information,

Paris, November 7, 2017

supported by

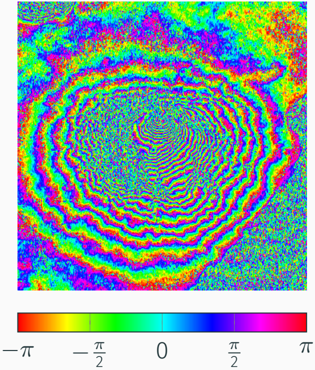
[a]joint work with D. Tenbrinck (WWU Münster)

Manifold-valued image processing

Manifold-valued images and data

New data aquisition modalities $\Rightarrow$ non-Euclidean range of data

- Interferometric synthetic aperture radar (InSAR)
- Surface normals
- Diffusion tensors in magnetic resonance imaging (DT-MRI)
- Electron backscattered diffraction (EBSD)
- Directional data: wind, flow, GPS,...



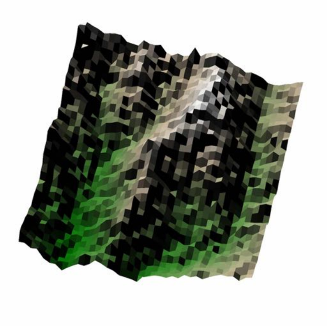
InSAR data of Mt. Vesuvius
[Rocca, Prati, Guarnieri 1997]

phase valued data, $\mathbb{S}^1$

Manifold-valued images and data

New data aquisition modalities $\Rightarrow$ non-Euclidean range of data

- Interferometric synthetic aperture radar (InSAR)
- Surface normals
- Diffusion tensors in magnetic resonance imaging (DT-MRI)
- Electron backscattered diffraction (EBSD)
- Directional data: wind, flow, GPS,...



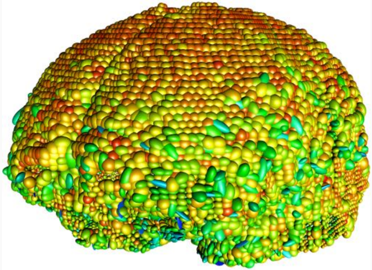
National elevation dataset
[Gesch, Evans, Mauck, 2009]

directional data, $\mathbb{S}^2$

Manifold-valued images and data

New data aquisition modalities $\Rightarrow$ non-Euclidean range of data

- Interferometric synthetic aperture radar (InSAR)
- Surface normals
- Diffusion tensors in magnetic resonance imaging (DT-MRI)
- Electron backscattered diffraction (EBSD)
- Directional data: wind, flow, GPS,…



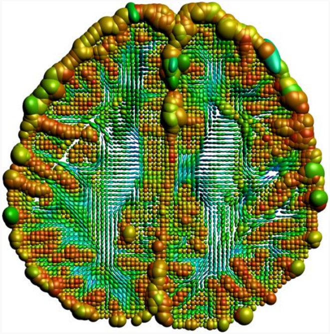
the Camino data set
http://cmic.cs.ucl.ac.uk/camino

sym. pos. def. Matrices, $\mathcal{P}(3)$

Manifold-valued images and data

New data aquisition modalities $\Rightarrow$ non-Euclidean range of data

- Interferometric synthetic aperture radar (InSAR)
- Surface normals
- Diffusion tensors in magnetic resonance imaging (DT-MRI)
- Electron backscattered diffraction (EBSD)
- Directional data: wind, flow, GPS,...



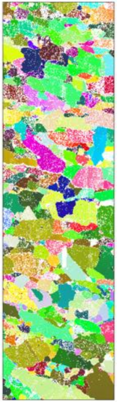
Slice # 28 from the Camino data set
http://cmic.cs.ucl.ac.uk/camino

sym. pos. def. Matrices, $\mathcal{P}(3)$

Manifold-valued images and data

New data aquisition modalities $\Rightarrow$ non-Euclidean range of data

- Interferometric synthetic aperture radar (InSAR)
- Surface normals
- Diffusion tensors in magnetic resonance imaging (DT-MRI)
- Electron backscattered diffraction (EBSD)
- Directional data: wind, flow, GPS,...



EBSD example from the MTEX toolbox
[Bachmann, Hielscher, since 2005]

rotations (mod. symmetry), $\mathrm{SO}(3)/\mathcal{S}$.

Manifold-valued images and data

New data aquisition modalities $\Rightarrow$ non-Euclidean range of data

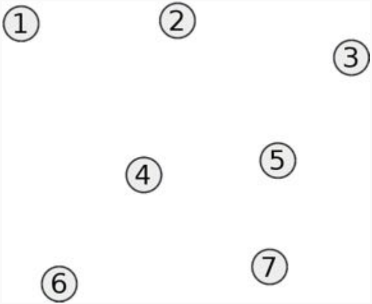
- Interferometric synthetic aperture radar (InSAR)
- Surface normals
- Diffusion tensors in magnetic resonance imaging (DT-MRI)
- Electron backscattered diffraction (EBSD)
- Directional data: wind, flow, GPS,...

Common properties

- The values lie on a Riemannian manifold
- tasks from "classical" image processing
- e.g. inpainting

Finite weighted graphs for data processing

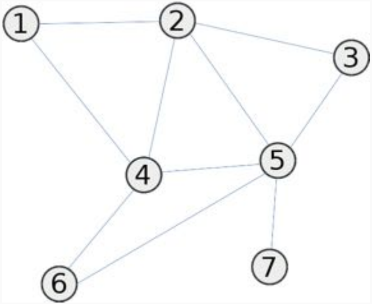
Finite weighted graphs



A finite weighted graph $G = (V, E, w)$ consists of

- a finite set of nodes V

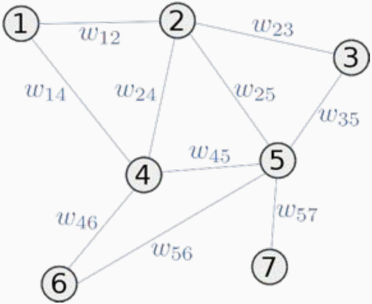
Finite weighted graphs



A finite weighted graph $G = (V, E, w)$ consists of

- a finite set of nodes V
- a finite set of directed edges $E \subset V \times V$

Finite weighted graphs



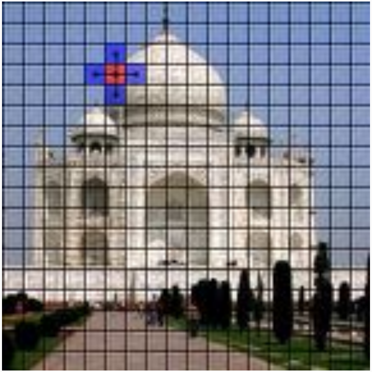
A finite weighted graph $G = (V, E, w)$ consists of

- a finite set of nodes V
- a finite set of directed edges $E \subset V \times V$
- a (symmetric) weight function $w : V \times V \to \mathbb{R}^+$,

$$w(u, v) = 0 \text{ for } v \not\sim u.$$

Finite weighted graphs for modeling discrete data

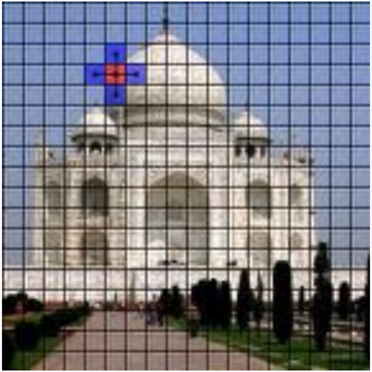
How can we apply graphs for image processing?



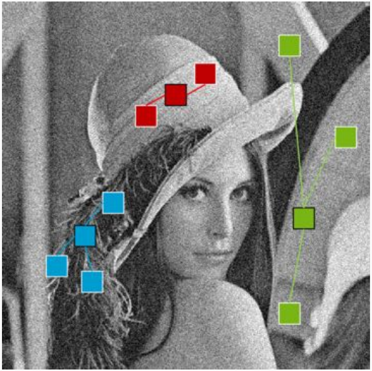
Local neighborhood
of a pixel

Finite weighted graphs for modeling discrete data

How can we apply graphs for image processing?



Local neighborhood
of a pixel



Nonlocal neighborhood
of a pixel

Finite weighted graphs for modeling discrete data

How can we apply graphs for polygon mesh processing?

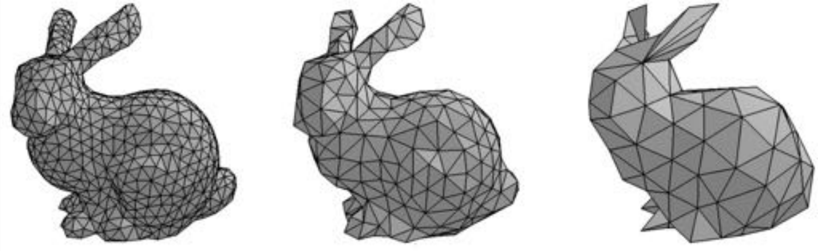


Image courtesy: Gabriel Peyré

Polygon mesh approximation of a 3D surface.

Finite weighted graphs for modeling discrete data

How can we apply graphs for point cloud processing?

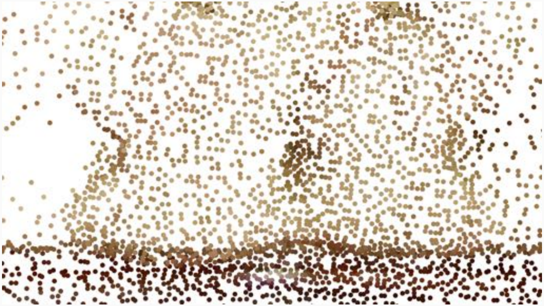


Image courtesy: François Lozes

Colored 3D point cloud data of a scanned chair.

Finite weighted graphs for modeling discrete data

How can we apply graphs for point cloud processing?

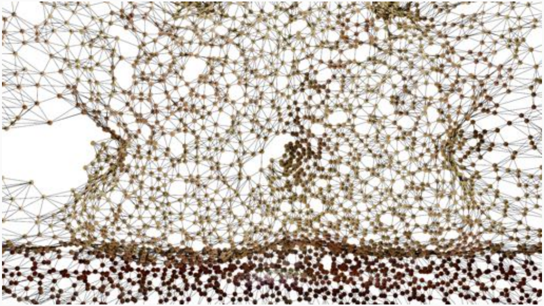
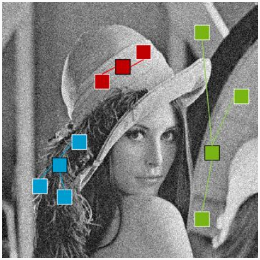


Image courtesy: François Lozes

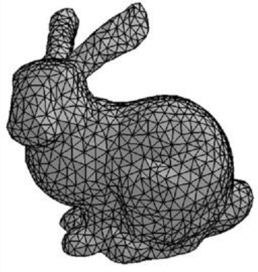
Graph construction on a 3D point cloud

Euclidean graph framework

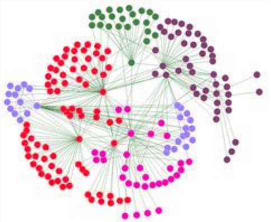
Application data on



a nonlocal neighborhood



a surface



Source: Wikipedia

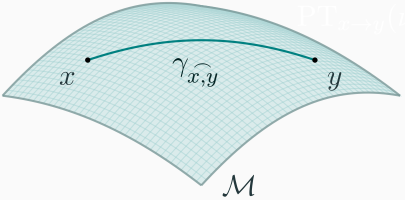
a social network graph

is represented by a vertex function $f: V \to \mathbb{R}^m$

"Anything can be modeled as a graph"

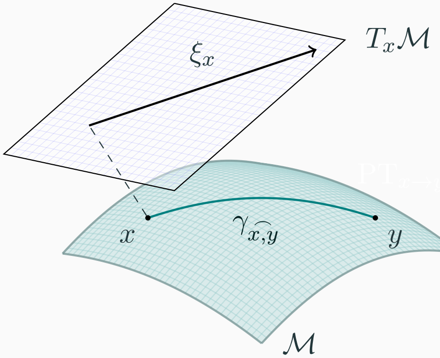
A manifold-valued graph framework & The ∞-graph Laplace

Notations on a Riemannian manifold $\mathcal{M}$



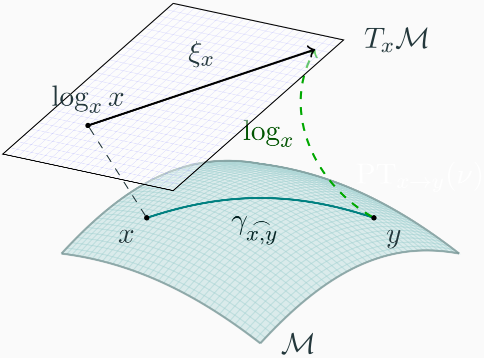
geodesic $\gamma_{\widehat{x,y}}$ shortest path (on $\mathcal{M}$) connecting $x, y \in \mathcal{M}$.

Notations on a Riemannian manifold $\mathcal{M}$



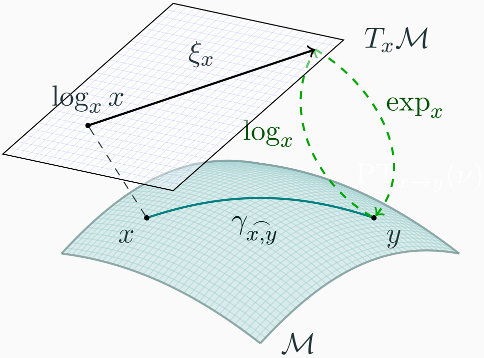
geodesic $\gamma_{\widehat{x,y}}$ shortest path (on $\mathcal{M}$) connecting $x, y \in \mathcal{M}$.
tangential plane $\mathrm{T}_x\mathcal{M}$ at x, $\mathrm{T}\mathcal{M} \coloneqq \cup_{x\in\mathcal{M}}\mathrm{T}_x\mathcal{M}$

Notations on a Riemannian manifold $\mathcal{M}$



geodesic $\gamma_{\widehat{x,y}}$ shortest path (on $\mathcal{M}$) connecting $x, y \in \mathcal{M}$.
tangential plane $\mathrm{T}_x\mathcal{M}$ at x, $\mathrm{T}\mathcal{M} \coloneqq \cup_{x\in\mathcal{M}}\mathrm{T}_x\mathcal{M}$
logarithmic map $\log_x y = \dot\gamma_{\widehat{x,y}}(0)$, “velocity towards y”

Notations on a Riemannian manifold $\mathcal{M}$



geodesic $\gamma_{\widehat{x,y}}$ shortest path (on $\mathcal{M}$) connecting $x, y \in \mathcal{M}$.
tangential plane $\mathrm{T}_x\mathcal{M}$ at x, $\mathrm{T}\mathcal{M} \coloneqq \cup_{x\in\mathcal{M}}\mathrm{T}_x\mathcal{M}$
logarithmic map $\log_x y = \dot\gamma_{\widehat{x,y}}(0)$, “velocity towards y”
exponential map $\exp_x \xi_x = \gamma(1)$, where $\gamma(0) = x$, $\dot\gamma(0) = \xi_x$

The real-valued ∞-Laplacian

Let $\Omega \subset \mathbb{R}^d$ be a bounded, open set and $f\colon \Omega \to \mathbb{R}$ smooth.

The infinity Laplacian $\Delta_\infty f$ in $x \in \Omega$ is defined as

[Crandall, Evans, Gariepy '01]

$$\Delta_\infty f(x) \;=\; \sum_{j=1}^{d}\sum_{k=1}^{d} \frac{\partial f}{\partial x_j}\frac{\partial f}{\partial x_k}\frac{\partial^2 f}{\partial x_j x_k}(x).$$

Applications in image interpolation and (stucture) inpainting.

[Caselles, Morel, Sbert '98]

A min-max discretization

Based on a simple approximation by min- and max-values in a neighborhood [Obermann, '04]

$$\Delta_\infty f(x) \;=\; \frac{1}{r^2}\left(\min_{y\in B_r(x)} f(y) + \max_{y\in B_r(x)} f(y) - 2f(x)\right) \;+\; \mathcal{O}(r^2).$$

a real-valued graph-based variant reads

[Elmoataz, Desquensnes, Lakhdari '14]

$$\begin{aligned}\Delta_\infty f(u) \;&=\; ||\nabla^+ f(u)||_\infty - ||\nabla^- f(u)||_\infty \\ &=\; \max_{v\sim u} |\min(\sqrt{w(u,v)}(f(v) - f(u)), 0)| \\ &\quad -\; \max_{v\sim u} |\max(\sqrt{w(u,v)}(f(v) - f(u)), 0)|\end{aligned}$$

Connection to AML extensions

Observation [Aronsson '67; Jensen '93]

Any (unique) viscosity solution f^* of the Dirichlet problem

$$\begin{cases} -\Delta_\infty f(x) \;=\; 0, & \text{for } x \in \Omega, \\ f(x) \;=\; \varphi(x), & \text{for } x \in \partial\Omega, \end{cases}$$

is an absolutely minimizing Lipschitz extension (AML) of φ, i.e.,

$$f^*(x) \;=\; g(x) \;\text{ for } x \in \partial\Sigma \;\Rightarrow\; ||Df^*||_{L^\infty(\Sigma)} \leq ||Dg||_{L^\infty(\Sigma)},$$

for every open, bounded subset $\Sigma \subset \Omega$ and every $g \in C(\overline{\Sigma})$

$\Rightarrow$ minimize locally the discrete Lipschitz constant [Obermann, '04]

$$\min_{f(x_0)} L(f(x_0)) \quad \text{with} \quad L(f(x_0)) \;=\; \max_{x_j \sim x_0} \frac{|f(x_0) - f(x_j)|}{|x_0 - x_j|}$$

$\Rightarrow$ consistent scheme for solving $-\Delta_\infty f = 0$.

Constructing discrete Lipschitz extensions

On $\mathbb{R}^m$ the infinity Laplace operator can be approximated by

$$\Delta_\infty f(x_0) = \frac{1}{|x_0 - x_j^*| + |x_0 - x_i^*|} \left(\frac{f(x_0) - f(x_j^*)}{|x_0 - x_j^*|} + \frac{f(x_0) - f(x_i^*)}{|x_0 - x_i^*|} \right)$$

where the neighbors (x_i^*, x_j^*) are determined by [Obermann, '04]

$$(x_i^*, x_j^*) = \operatorname*{argmax}_{x_i, x_j \sim x_0} \frac{|\ f(x_i) \qquad\qquad - \ f(x_j) \qquad\qquad |}{|x_0 - x_i| + |x_0 - x_j|}$$

Constructing discrete Lipschitz extensions

On $\mathbb{R}^m$ the infinity Laplace operator can be approximated by

$$\Delta_\infty f(x_0) \; = \; \frac{1}{|x_0 - x_j^*| + |x_0 - x_i^*|} \left(\frac{f(x_0) - f(x_j^*)}{|x_0 - x_j^*|} + \frac{f(x_0) - f(x_i^*)}{|x_0 - x_i^*|} \right)$$

where the neighbors (x_i^*, x_j^*) are determined by [Obermann, '04]

$$(x_i^*, x_j^*) \; = \; \operatorname*{argmax}_{x_i, x_j \sim x_0} \frac{|(f(x_i) - f(x_0)) - (f(x_j) - f(x_0))|}{|x_0 - x_i| + |x_0 - x_j|}$$

Graph ∞-Laplacian for manifold-valued data

We define the graph-∞-Laplace operator for manifold valued data $\Delta_\infty f$ in a vertex $u \in V$ as

$$\Delta_\infty f(u) \ := \ \frac{\sqrt{w(u,v_1^*)}\log_{f(u)} f(v_1^*) \ + \ \sqrt{w(u,v_2^*)}\log_{f(u)} f(v_2^*)}{\sqrt{w(u,v_1^*)} + \sqrt{w(u,v_2^*)}},$$

where $v_1^*, v_2^* \in \mathcal{N}(u)$ maximize the discrete Lipschitz constant in the local tangent space $T_{f(u)}\mathcal{M}$ among all neighbors, i.e.,

$$\begin{aligned}&(v_1^*, v_2^*)\\ &\ = \ \operatorname*{argmax}_{(v_1,v_2)\in\mathcal{N}^2(u)} \left\| \sqrt{w(u,v_1)}\log_{f(u)} f(v_1) - \sqrt{w(u,v_2)}\log_{f(u)} f(v_2) \right\|_{f(u)}\end{aligned}$$

Numerical iteration scheme

to solve

$$\begin{cases} \Delta_\infty f(u) = 0 & \text{for all } u \in U, \\ f(u) = g(u) & \text{for all } u \in V/U. \end{cases}$$

we introduce an artificial time dimension t, i.e.

$$\begin{cases} \frac{\partial f}{\partial t}(u,t) = \Delta_\infty f(u,t) & \text{for all } u \in U,\ t \in (0,\infty), \\ f(u,0) = f_0(u) & \text{for all } u \in U, \\ f(u,t) = g(u,t) & \text{for all } u \in V/U, t \in [0,\infty). \end{cases}$$

and propose an explicit Euler scheme with step size $\tau > 0$ using $f_k(u) := f(u, k\tau)$ we obtain

$$f_{k+1}(u) = \exp_{f_k(u)}\bigl(\tau \Delta_\infty f_k(u)\bigr), \text{ for all } u \in V$$

Numerical examples

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck ’16]

Interpolation of structure

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:

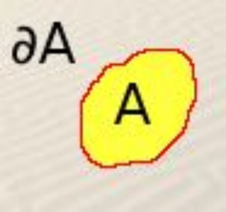
Interpolation of structure

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone (red)
   $\rightarrow$ local connection for inner nodes in A



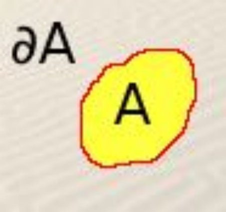
Interpolation of structure

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone (red)
   $\rightarrow$ local connection for inner nodes in A
2. Solve $\Delta_\infty f(u) = 0$ for all vertices $u \in A \subset V$



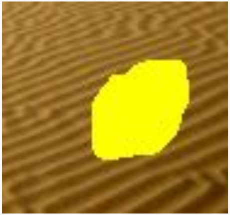
Interpolation of structure

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone
   $\rightarrow$ local connection for inner nodes in A
2. Solve $\Delta_\infty f(u) = 0$ for all vertices $u \in A \subset V$
3. Add border nodes to ∂A and repeat until $A = \emptyset$.



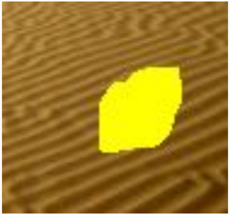
Interpolation of structure

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone
   $\rightarrow$ local connection for inner nodes in A
2. Solve $\Delta_\infty f(u) = 0$ for all vertices $u \in A \subset V$
3. Add border nodes to ∂A and repeat until $A = \emptyset$.



Interpolation of structure

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone
   $\rightarrow$ local connection for inner nodes in A
2. Solve $\Delta_\infty f(u) = 0$ for all vertices $u \in A \subset V$
3. Add border nodes to ∂A and repeat until $A = \emptyset$.

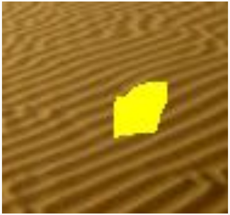
Interpolation of structure

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone
   $\rightarrow$ local connection for inner nodes in A
2. Solve $\Delta_\infty f(u) = 0$ for all vertices $u \in A \subset V$
3. Add border nodes to ∂A and repeat until $A = \emptyset$.



Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone
   $\rightarrow$ local connection for inner nodes in A
2. Solve $\Delta_\infty f(u) = 0$ for all vertices $u \in A \subset V$
3. Add border nodes to ∂A and repeat until $A = \emptyset$.

Interpolation of structure

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone
   $\rightarrow$ local connection for inner nodes in A
2. Solve $\Delta_\infty f(u) = 0$ for all vertices $u \in A \subset V$
3. Add border nodes to ∂A and repeat until $A = \emptyset$.

Interpolation of structure

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone
   $\rightarrow$ local connection for inner nodes in A
2. Solve $\Delta_\infty f(u) = 0$ for all vertices $u \in A \subset V$
3. Add border nodes to ∂A and repeat until $A = \emptyset$.

Interpolation of structure

Goal

Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone
   $\rightarrow$ local connection for inner nodes in A
2. Solve $\Delta_\infty f(u) = 0$ for all vertices $u \in A \subset V$
3. Add border nodes to ∂A and repeat until $A = \emptyset$.

Interpolation of structure

Goal

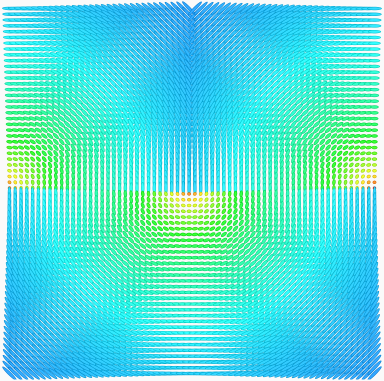
Inpaint $A \subset V$ using information in $\partial A = V/A$.

[Elmoataz, Toutain, Tenbrinck '16]

1. Build a graph using image patches and local neighbors:
   $\rightarrow$ nonlocal relationships for vertices in border zone
   $\rightarrow$ local connection for inner nodes in A
2. Solve $\Delta_\infty f(u) = 0$ for all vertices $u \in A \subset V$
3. Add border nodes to ∂A and repeat until $A = \emptyset$.

Inpainting of symmetric positive definite matrices

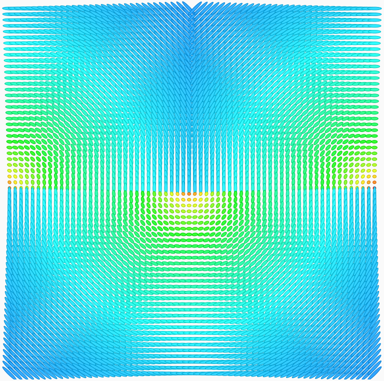
manifold $\mathcal{M} = \mathcal{P}(2)$, graph construction from previous slide



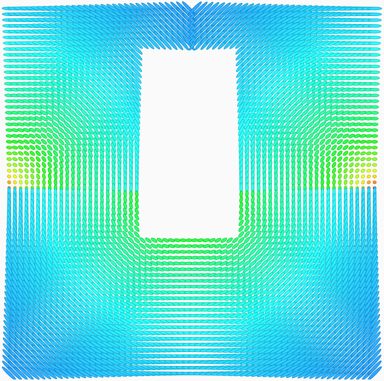
Original data

Inpainting of symmetric positive definite matrices

manifold $\mathcal{M} = \mathcal{P}(2)$, graph construction from previous slide



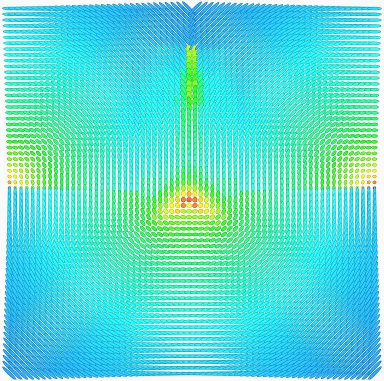
Original data



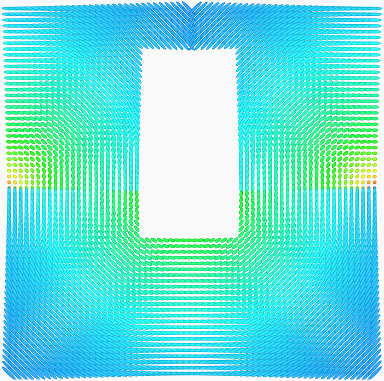
Given (lossy) data

Inpainting of symmetric positive definite matrices

manifold $\mathcal{M} = \mathcal{P}(2)$, graph construction from previous slide



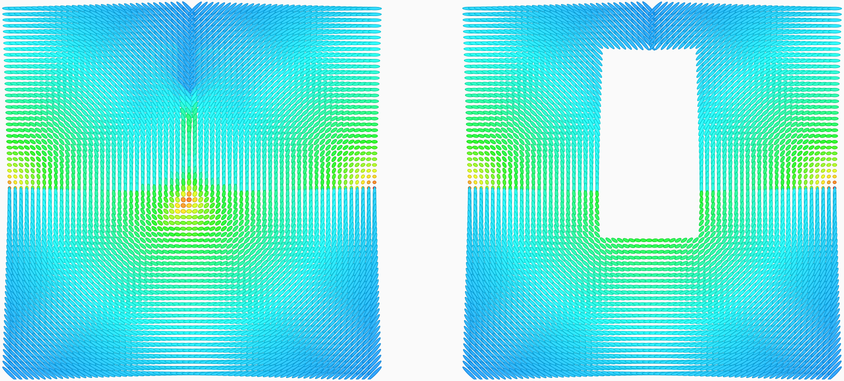
Inpainting with
25 neighbors, patch size 6



Given (lossy) data

Inpainting of symmetric positive definite matrices

manifold $\mathcal{M} = \mathcal{P}(2)$, graph construction from previous slide

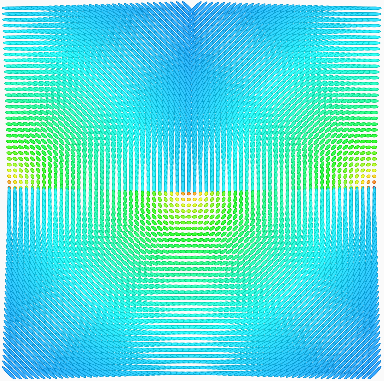


Inpainting with
5 neighbors, patch size 6

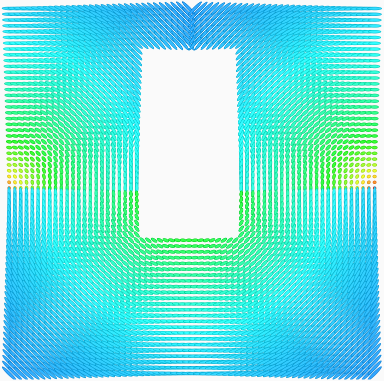
Given (lossy) data

Inpainting of symmetric positive definite matrices

manifold $\mathcal{M} = \mathcal{P}(2)$, graph construction from previous slide



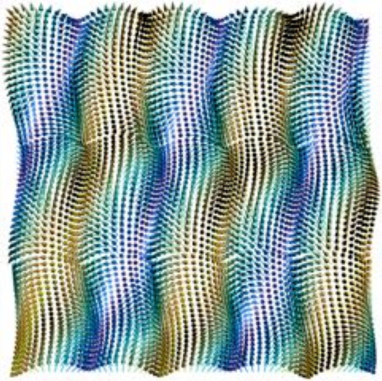
Original data



Given (lossy) data

Inpainting of directional data

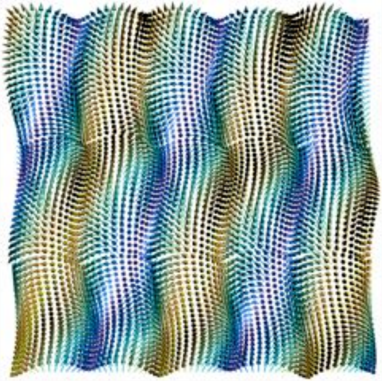
manifold $\mathcal{M} = \mathbb{S}^2$, graph construction from previous slide.



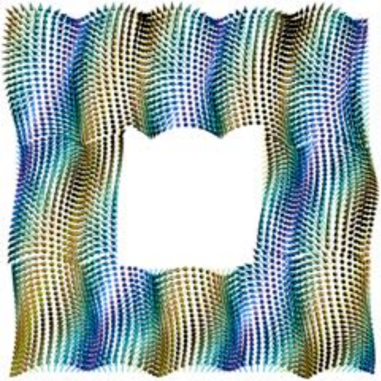
Original data

Inpainting of directional data

manifold $\mathcal{M} = \mathbb{S}^2$, graph construction from previous slide.



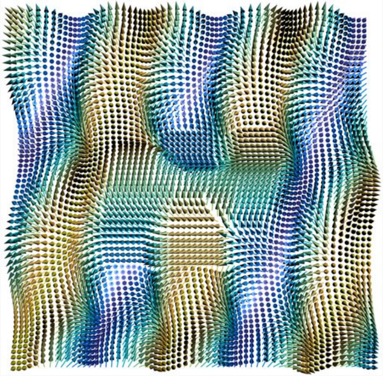
Original data



Given (lossy) data

Inpainting of directional data

manifold $\mathcal{M} = \mathbb{S}^2$, graph construction from previous slide.

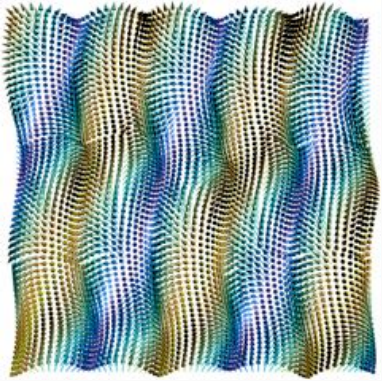


Inpainting with
first and second order TV

Given (lossy) data

Inpainting of directional data

manifold $\mathcal{M} = \mathbb{S}^2$, graph construction from previous slide.

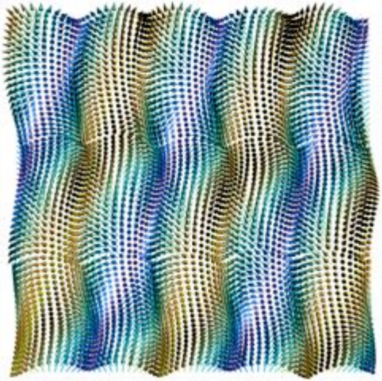


Inpainted with graph ∞-Laplace

Given (lossy) data

Inpainting of directional data

manifold $\mathcal{M} = \mathbb{S}^2$, graph construction from previous slide.



Original data

Given (lossy) data

Conclusion

Conclusion

- manifold-valued graph ∞-Laplacian for inpainting
- model local and nonlocal features
- inpaint structure on manifold-valued data

Conclusion

- manifold-valued graph ∞-Laplacian for inpainting
- model local and nonlocal features
- inpaint structure on manifold-valued data

Future work

- consistency
- other graph based PDEs
- other image processing tasks (segmentation)
- other numerical schemes

Literature

RB and D. Tenbrinck. “A Graph Framework for manifold-valued Data”. In: SIAM J. Imaging Sci. (2017). accepted. arXiv: 1702.05293.

RB and D. Tenbrinck. Nonlocal Inpainting of Manifold-valued Data on Finite Weighted Graphs. GSI’17. 2017. arXiv: 1702.05293.

A. Elmoataz, M. Toutain, and D. Tenbrinck. “On the p-Laplacian and ∞-Laplacian on Graphs with Applications in Image and Data Processing”. In: SIAM J. Imag. Sci. 8.4 (2015), pp. 2412–2451.

Adam M. Oberman. “A convergent difference Scheme for the Infinity Laplacian: Construction of absolutely minimizing Lipschitz extensions”. In: Math. Comp. 74.251 (2004), pp. 1217–1230.

Open source Matlab software MVIRT:
www.mathematik.uni-kl.de/imagepro/members/bergmann/mvirt/