Westfälische
Wilhelms-Universität
Münster

Partial Difference Operators on Weighted Graphs for Point Cloud Processing

From 3D Surfaces to Machine Learning

Abderrahim Elmoataz[1,2], Francois Lozes[1], **Daniel Tenbrinck**[3]

[1]Université de Caen Normandie, Caen, France
[2]Université de Paris-Est Marne-la-Vallée, Paris, France
[3]Westfälische Wilhelms-Universität Münster, Münster, Germany

February 9th, 2016

Outline

Introduction

- Motivation
- Related Work

Methods

- Finite Weighted Graphs
- Graph Construction for 3D Point Cloud Data
- Partial Difference Operators on Weighted Graphs

Applications

- 3D Point Cloud Processing
- Machine Learning

Motivation

Processing of 3D point cloud surfaces gets increasingly important, due to:

- recent technical advances leading to huge amounts of 3D data

Recent technical advances

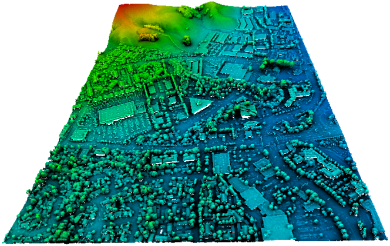
Past: 3D surface data acquisition from manually painted meshes [1].

[1] H. Gouraud: *Computer display of curved surfaces*. Technical report at University of Utah (1971)

Recent technical advances

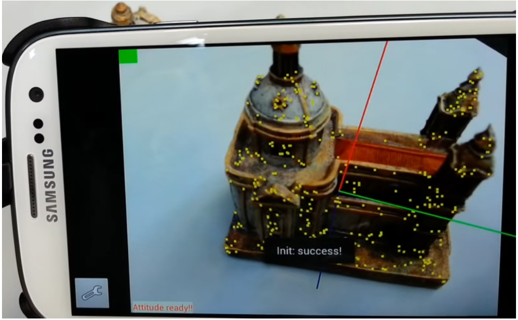
Today: 3D surface data acquisition with Microsoft Kinect devices

Recent technical advances



Today: Aerial 3D data acquisition for surface mapping using drones

Recent technical advances



Today: 3D surface data acquisition with common smartphone [2]

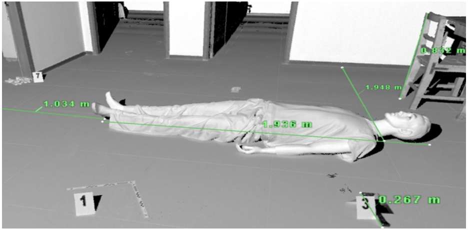
[2] P.Tanskanen, M. Pollefeys, et al.: *Live Metric 3D Reconstruction on Mobile Phones*. ICCV (2013)

Motivation

Processing of 3D point cloud surfaces gets increasingly important, due to:

- recent technical advances leading to huge amounts of 3D data
- new application fields

New applications: 3D digital forensics

3D scan of a crime scene as digital forensic tool for police investigations [3]

[3] Faro (http://www.faro.com/measurement-solutions/industries/forensics)

New applications: Digital archaeology

Destruction of cultural heritage site in Palmyra, Syria by ISIS in 2015

New applications: Digital archaeology

Digital preservation of cultural heritage at-risk in Middle East and North Africa [4,5]

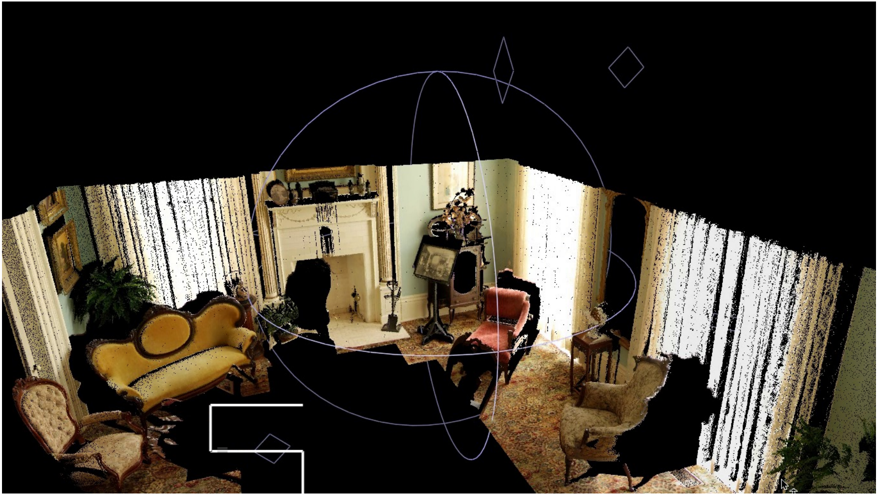
[4] The Insitute for Digital Archeology: The Million Image Database (http://digitalarchaeology.org.uk)
[5] CyArk, non-profit organization (http://www.cyark.org/)

Motivation

Processing of 3D point cloud surfaces gets increasingly important, due to:

- recent technical advances leading to huge amounts of 3D data
- new application fields
- challenging methodological problems

The challenges of 3D point cloud data

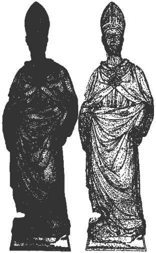


The challenges of 3D point cloud data

In many applications the acquired raw 3D point clouds:

- consist of massive amounts of (possibly noisy) data,
- contain gaps and sparse regions,
- are a-priori not connected.

→ There is a huge need to simplify, filter, and interpolate 3D point clouds.



Motivation

Processing of 3D point cloud surfaces gets increasingly important, due to:

- recent technical advances leading to huge amounts of 3D data
- new application fields
- challenging methodological problems

Lots of data representable as points in high-dimensional spaces, e.g.:

- patches in image processing and computer vision

Motivation

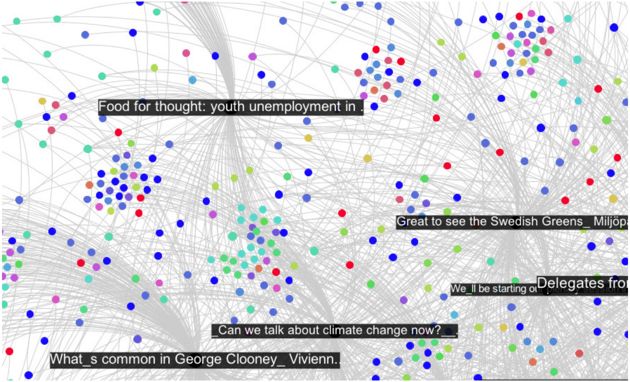
Processing of 3D point cloud surfaces gets increasingly important, due to:

- recent technical advances leading to huge amounts of 3D data
- new application fields
- challenging methodological problems

Lots of data representable as points in high-dimensional spaces, e.g.:

- patches in image processing and computer vision
- entities in network analysis

Example: Social networks



Visualization of a social media network with interacting users [6].

[6] *Netvizz*: https://tools.digitalmethods.net/netvizz/facebook/netvizz/

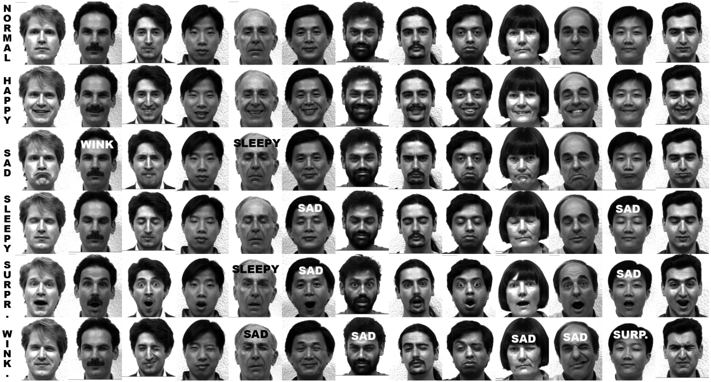
Motivation

Processing of 3D point cloud surfaces gets increasingly important, due to:

- recent technical advances leading to huge amounts of 3D data
- new application fields
- challenging methodological problems

Lots of data representable as points in high-dimensional spaces, e.g.:

- patches in image processing and computer vision
- entities in network analysis
- feature vectors in machine learning



Yale face database for training/evaluation of face recognition algorithms [7]

[7] A. Georghiades et al.: *"From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose"*. PAMI (2001)

Aim of this talk

Our goal is to use PDEs and variational methods to process:

- discrete representations of 3D surfaces,
- high-dimensional point cloud data of arbitrary topology.

Question:

How can we translate mathematical operators to general point cloud data?

Our approach:

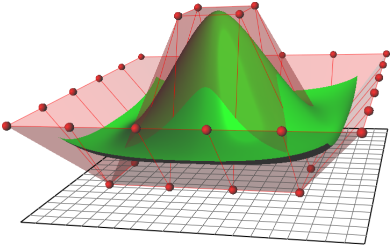
We represent surfaces and point clouds as weighted finite graphs and introduce a simple, discrete calculus to tackle challenging tasks, e.g.,

- filtering, clustering, inpainting, classification, ...

Outline

Related work: Explicit methods

- ✓ Differential operators on surface are given directly by parametrization
- ✓ Computations are efficient
- × Parametrization of complex surfaces difficult
- × Dynamic surfaces need reparametrization in each timestep
- × Topology changes are cumbersome



Parametrized surface using NURBS
(Image courtesy: Wikimedia Commons)

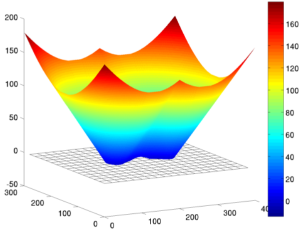
Related work: Explicit methods

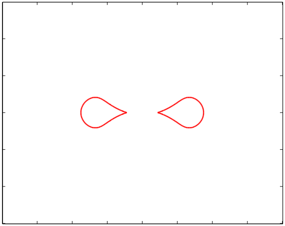
- ✓ Differential operators on surface are given directly by parametrization
- ✓ Computations are efficient
- × Parametrization of complex surfaces difficult
- × Dynamic surfaces need reparametrization in each timestep
- × Topology changes are cumbersome

[8] L. Piegl, W. Tiller: *The NURBS Book*. Springer (1997)
[9] K. Höllig: *Finite Element Methods with B-Splines*. SIAM (2003)
[10] J. Stam: *Flows on Surfaces of Arbitrary Topology* ACM T. Graphics 22 (2003)
[11] A. Spira, R. Kimmel: *Geometric Curve Flows on Parametric Manifolds*. J. Comput Phys. 223 (2007)
[12] L. Lopez-Perez, R. Deriche, N. Sochen: *The Beltrami Flow over Triangulated Manifolds*. Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis 3117 (2004)
[13] X. Gu, S.T. Yau: *Global Conformal Surface Parameterization*. Symposium on Geometry Processing (2003)
[14] L.M. Lui, X. Gu, T.F. Chan, S.T. Yau: *Variational Method on Riemann Surfaces using Conformal Parameterization and Its Applications to Image Processing*. Methods and Applications of Analysis 15 (2008)

Related work: Implicit methods

- ✓ No explicit surface operators needed
- ✓ Dynamic surfaces and topological changes are easy to handle
- × Calculations can be cost intensive (esp. in high-dimensional spaces)





Implicit representation of two-dimensional contour as zero-level set

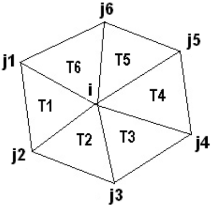
Related work: Implicit methods

- ✓ No explicit surface operators needed
- ✓ Dynamic surfaces and topological changes are easy to handle
- × Calculations can be cost intensive (esp. in high-dimensional spaces)

[15] M. Bertalmio, L.T. Cheng, S. Osher, G. Sapiro: *Variational Problems and Partial Differential Equations on Implicit Surfaces*. J. Comput Phys. 174 (2001)
[16] L.T. Cheng, P. Burchard, B. Merriman, S. Osher: *Motion of Curves Constrained on Surfaces Using a Level-Set Approach*. J. Comput Phys. 175 (2002)
[17] A. Oberman, S. Osher, R. Takei, R. Tsai: *Numerical Methods for Smooth and Crystalline Mean Curvature Flow*. Commun. Math. Sci. 9 (2011)
[18] B. Dong, J. Ye, S. Osher, I Dinov: *Level Set Based Nonlocal Surface Restauration* Multiscale Model. Simul., 7 (2008)
[19] S.J. Ruuth, B. Merriman: *A Simple Embedding Method for Solving Partial Differential Equations on Surfaces*. J. Comput Phys. 227 (2008)
[20] A. Rätz, A. Voigt: *PDEs on Surfaces - a Diffuse Interface Approach*. Communications in Mathematical Sciences 4 (2006)
[21] M. Burger: *Finite element approximation of elliptic partial differential equations on implicit surfaces*. Computing and Visualization in Science 12 (2009)

Related work: Intrinsic methods

- ✓ Calculations are based directly on mesh points
- ✓ Computations are efficient
- × Triangulation as pre-processing needed
- × Topological changes are cumbersome



$$\nabla_M^d f(p_i) = \frac{1}{\sum_l Area(T_l)} \sum_l Area(T_l) \nabla_{T_l}^d f(p_0)$$

$$\mathrm{div}_M^d \mathbb{V}(p_i) = \frac{1}{\sum_l Area(T_l)} \sum_l Area(T_l) \mathrm{div}_{T_l}^d \mathbb{V}(p_0)$$

Related work: Intrinsic methods

- ✓ Calculations are based directly on mesh points
- ✓ Computations are efficient
- × Triangulation as pre-processing needed
- × Topological changes are cumbersome

[22] M. Meyer, M. Desbrun, P. Schröder, A.H. Barr: *Discrete Differential-Geometry Operators for Triangulated 2-Manifolds* in: Visualization and Mathematics III, Springer (2003)
[23] G. Xu: *Convergent Discrete Laplace–Beltrami Operator over Triangular Surfaces*. in: Proceedings of Geometric Modelling and Processing (2004)
[24] U. Clarenz, M. Rumpf, A. Telea: *Surface Processing Methods for Point Sets Using Finite Elements*. Journal on Computers and Graphics 28 (2004)
[25] R. Lai, T.F. Chan: *A framework for intrinsic image processing on surfaces*. Comput. Vis. Image Und. 115 (2011)
[26] R. Lai, J. Liang, H.K. Zhao: *A Local Mesh Method for Solving PDEs on Point Clouds*. Inv. Probl. Imag. 7 (2013)
[27] X. Li, G. Xu, Y.J. Zhang: *Localized discrete Laplace-Beltrami operator over triangular mesh*. Computer Aided Geometric Design 39 (2015)
[28] I. Kokkinos, M. Bronstein, R. Litman, A. Bronstein: *Intrinsic Shape Context Descriptors for Deformable Shapes*. CVPR (2012)

Outline

Introduction

- Motivation
- Related Work

Methods

- Finite Weighted Graphs
- Graph Construction for 3D Point Cloud Data
- Partial Difference Operators on Weighted Graphs

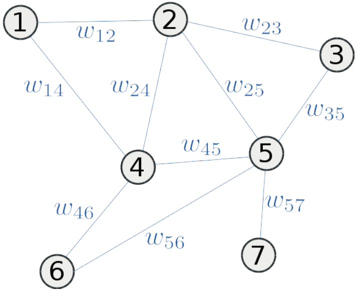
Applications

- 3D Point Cloud Processing
- Machine Learning

Finite weighted graphs

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

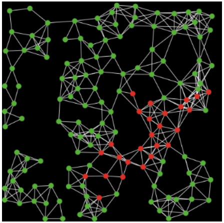
- a finite set of vertices $V = (v_1, \ldots, v_n)$
- a finite set of edges $E \subset V \times V$
- a weight function $w\colon E \to [0, 1]$

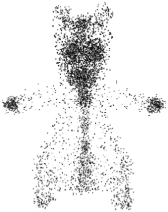


Vertex and edge functions

A vertex function $f: V \to R^m$ assigns each $v \in V$ a feature vector, e.g.,

- node labels
- grayscale values or RGB color vectors
- 3D coordinates





An edge function $F: E \to \mathbb{R}^k$ assigns each edge $(u, v) \in E$ a vector, e.g., the weight function $w: E \to [0, 1]$ for $k = 1$.

Measures and function spaces

We define the integral of a vertex function $f\colon V \to \mathbb{R}^m$ as:

$$\int_V f = \sum_{u \in V} f(u) \tag{1}$$

We can measure vertex functions using the following l^p-norms:

$$||f||_p = \begin{cases} \left(\sum_{u\in V} ||f(u)||^p\right)^{1/p}, & 1 \leq p < \infty \\ \max_{u \in V} ||f(u)||, & p = \infty \end{cases} \tag{2}$$

The space of vertex functions $H(V)$ for $G(V, E, w)$ is a finite-dimensional Hilbert space with the following norm:

$$||f||_{H(V)} = \sqrt{\sum_{v \in V} \langle f(v), f(v) \rangle_{\mathbb{R}^m}} \tag{3}$$

Choice of weight function

Construct weight functions $w\colon E \to [0,1]$ based on similarity of features.

Given a vertex function $f\colon V \to \mathbb{R}$. To compute a weight $w(u,v) = w(v,u)$ between two nodes $u, v \in V$ we need:

- a symmetric distance function $d\colon \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^+$

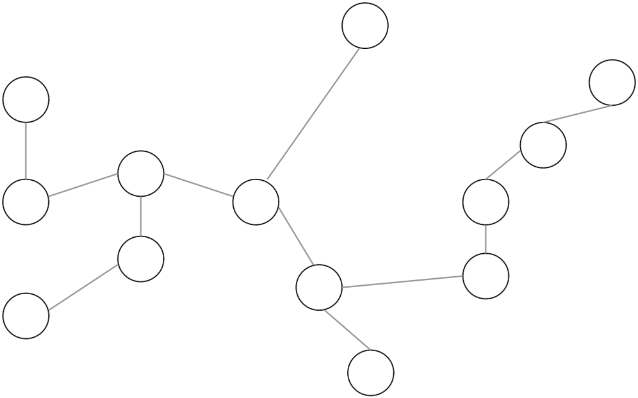
  e.g., Euclidean distance, patch distance, ...
- a normalized similarity function $s\colon \mathbb{R}^+ \to [0,1]$

  e.g., constants, probability density functions, ...

Example: Patch-based weight function

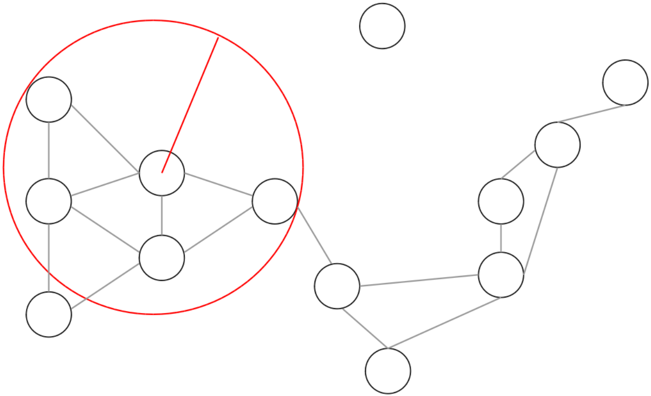
$$w(u,v) = \exp\left(-\frac{d(\mathcal{P}(u),\mathcal{P}(v))^2}{\sigma^2}\right) \quad \text{with} \quad d(\mathcal{P}(u),\mathcal{P}(v)) = ||\mathcal{P}(u) - \mathcal{P}(v)||_2$$

General graph construction



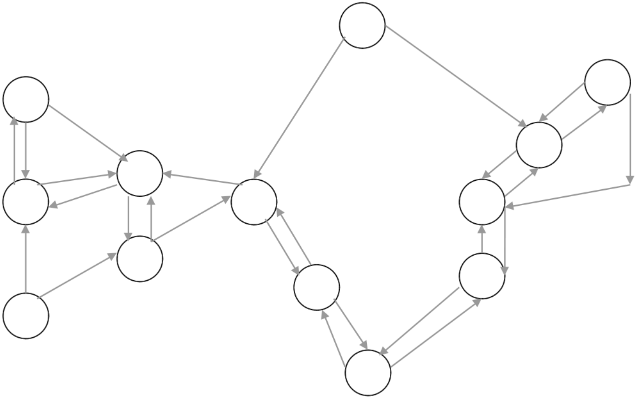
Minimum-Spanning-Tree graph

General graph construction



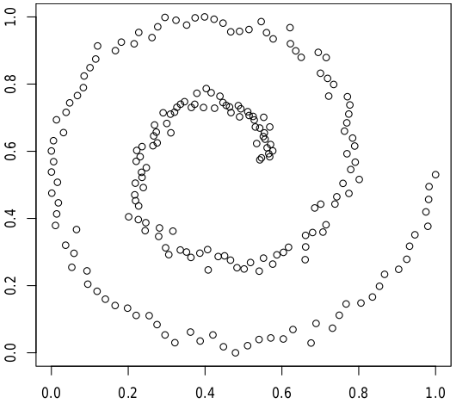
ϵ-ball graph

General graph construction



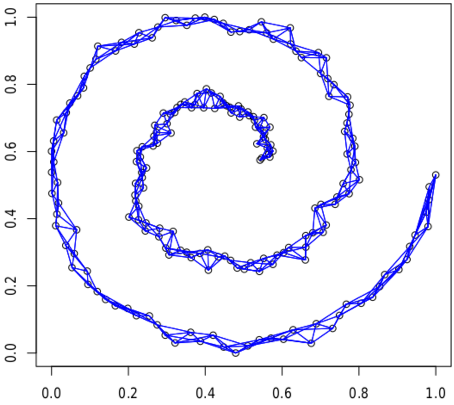
k-Nearest-Neighbor graph (k=2)

General graph construction



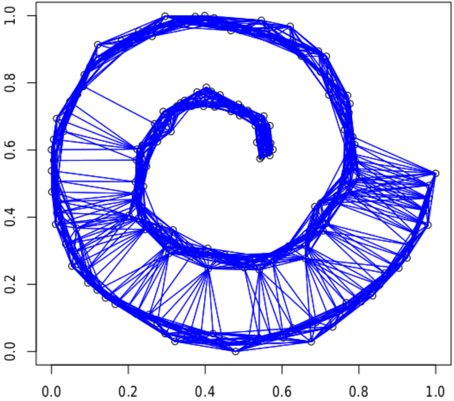
Distribution of 2D points

General graph construction



k-NN graph for k=3

General graph construction



k-NN graph for k=15

Outline

Introduction

- Motivation
- Related Work

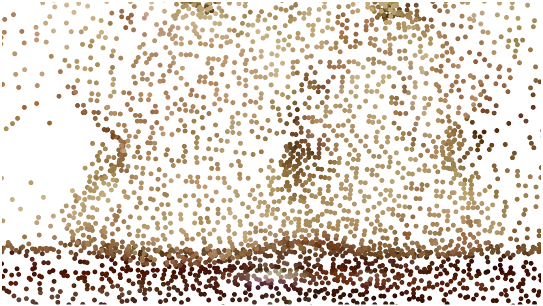
Methods

- Finite Weighted Graphs
- Graph Construction for 3D Point Cloud Data
- Partial Difference Operators on Weighted Graphs

Applications

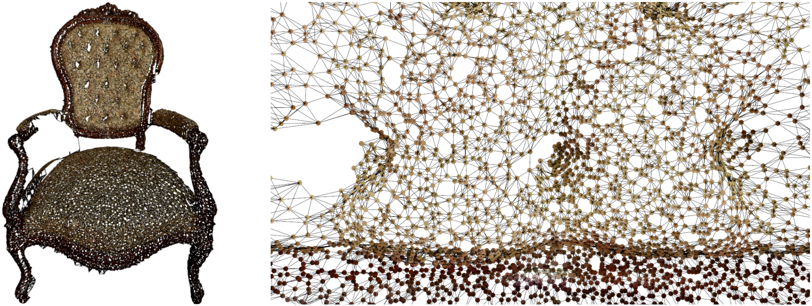
- 3D Point Cloud Processing
- Machine Learning

Geometric graph construction for 3D point clouds

Colored 3D point cloud data of a scanned chair

Geometric graph construction for 3D point clouds



Construction of symmetric k-NN graph from 3D point cloud using local geometry

Geometric graph construction for 3D point clouds

Observation:

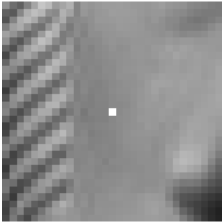
Using weight functions based directly on only local features (often) leads to problems, due to:

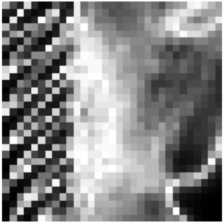
- missing information
- geometric noise
- uncertainty in features, e.g., RGB color noise

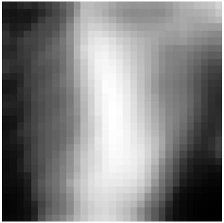
Goal:

Introduce a more robust distance function for 3D point clouds.

Patches in image processing


Pixel of interest


Intensity-based distance


Patch-based distance

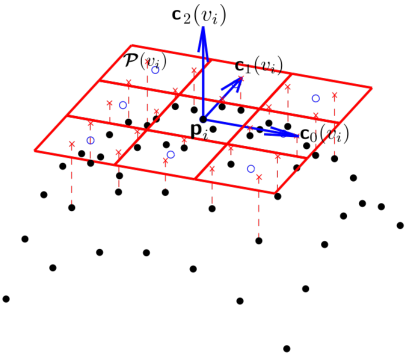
Observation:

Comparing patches implicitly induces local regularity!

→ How to construct patches on 3D point cloud data?

[29] J. Digne: *Similarity Based Filtering of Point Clouds*. CVPR Workshop (2012)
[30] R.F. Wang, W.Z. Chen, S.Y. Zhang, Y. Zhang, X.Z. Ye: *Similarity-Based Denoising of Point-Sampled Surfaces*. Journal of Zhejiang University 9 (2008)
[31] A. Wetzler, G. Rosman, R. Kimmel: *Patch-Space Beltrami Denoising of 3D Point Clouds*. IEEE Convention of Electrical & Electronics Engineers in Israel (2012)

Patch construction for 3D point clouds



Construction of oriented patch in tangential plane with length *l*

[32] F. Lozes, A. Elmoataz, O. Lezoray: *Partial Difference Operators on Weighted Graphs for Image Processing on Surfaces and Point Clouds*. IEEE TIP 23 (2014)

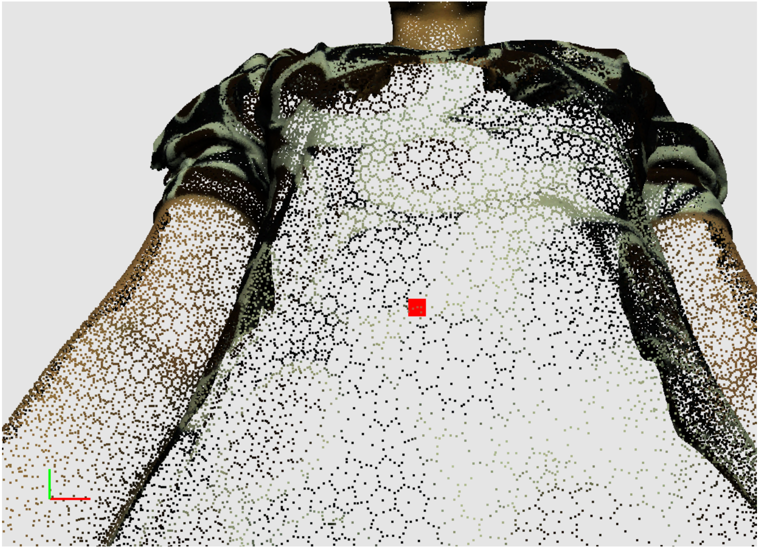
Example: 3D color patches

Color patch construction on 3D point cloud of a scanned woman

Example: 3D color patches

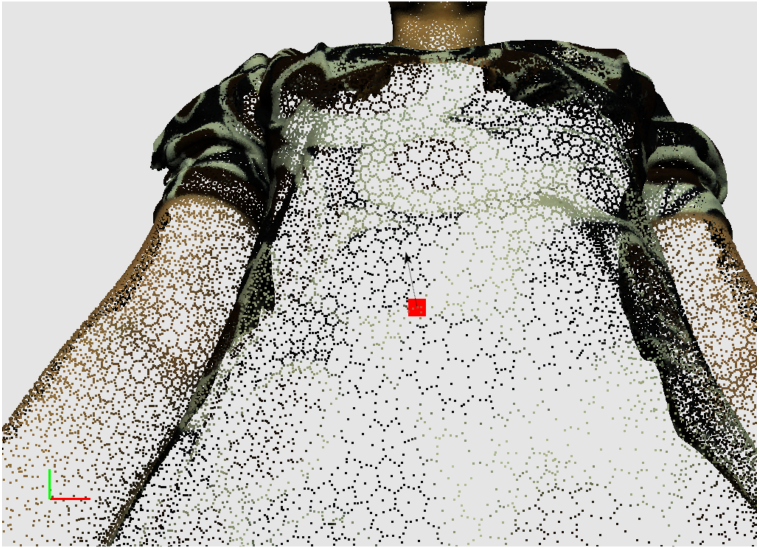
Color patch construction on 3D point cloud of a scanned woman

Example: 3D color patches



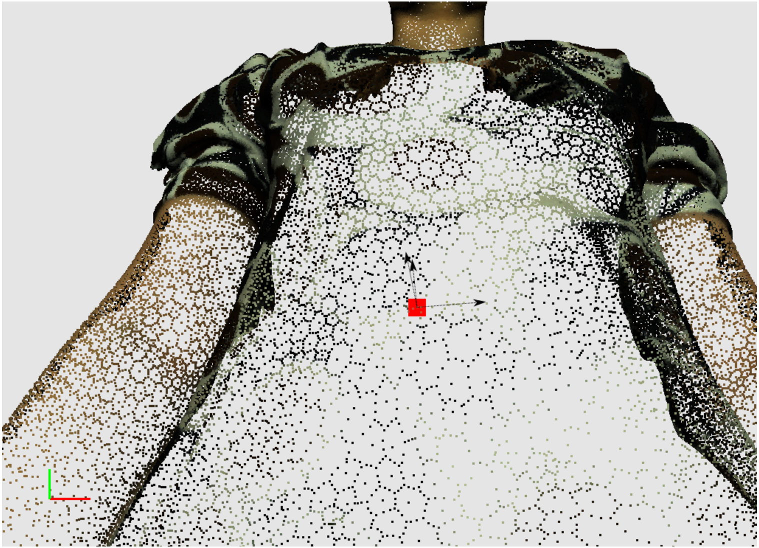
Color patch construction on 3D point cloud of a scanned woman

Example: 3D color patches



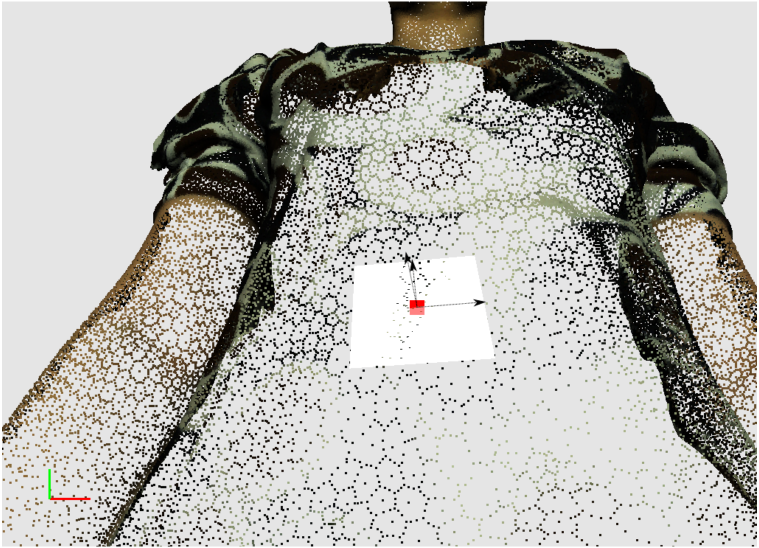
Color patch construction on 3D point cloud of a scanned woman

Example: 3D color patches



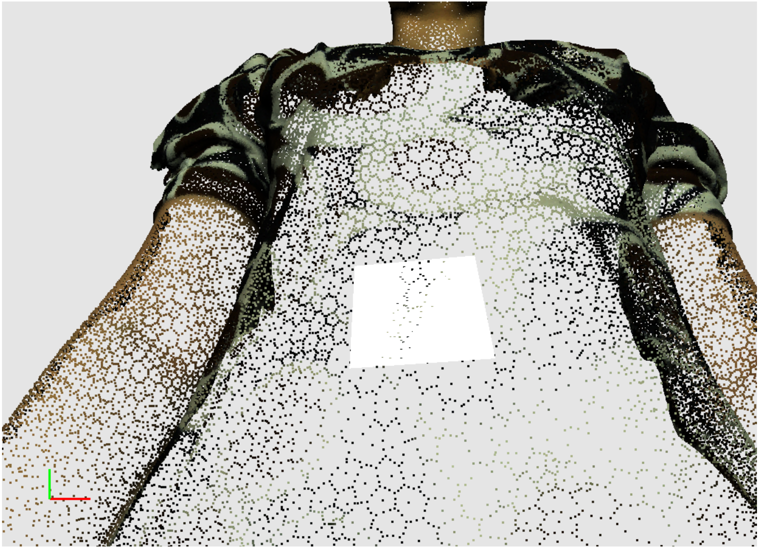
Color patch construction on 3D point cloud of a scanned woman

Example: 3D color patches



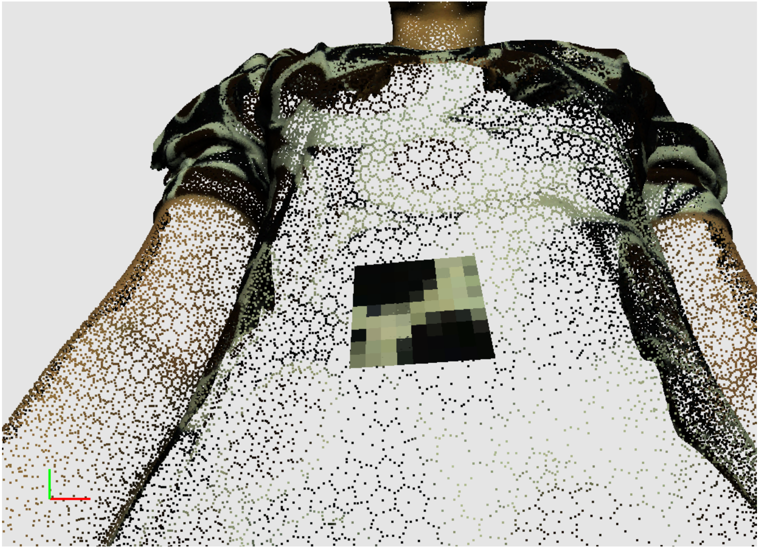
Color patch construction on 3D point cloud of a scanned woman

Example: 3D color patches



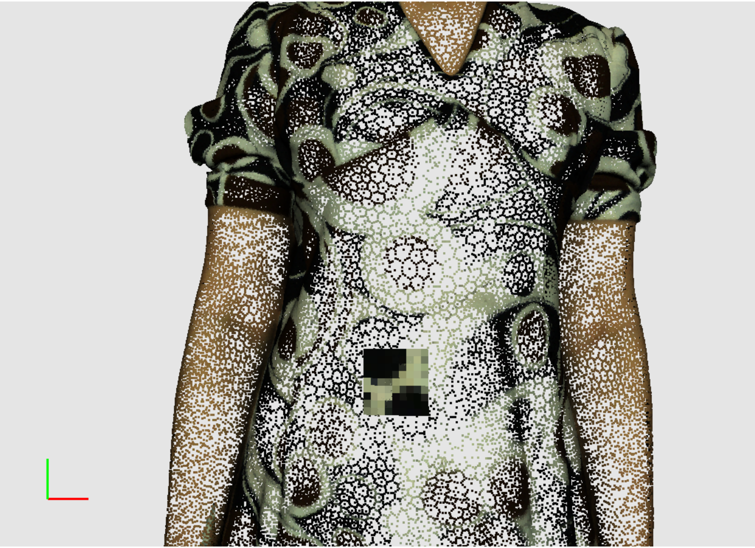
Color patch construction on 3D point cloud of a scanned woman

Example: 3D color patches



Color patch construction on 3D point cloud of a scanned woman

Example: 3D color patches



Color patch construction on 3D point cloud of a scanned woman

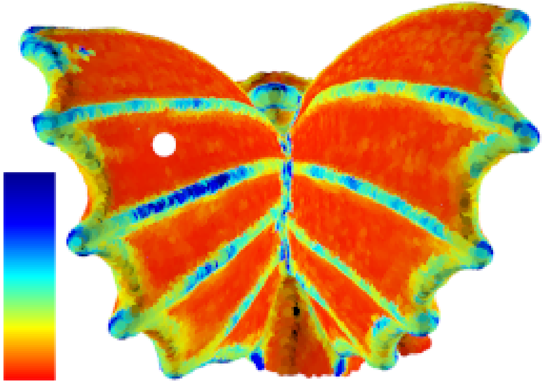
Example: 3D color patches

Color patch construction on 3D point cloud of a scanned woman

Example: Height patches

Height patch on 3D point cloud of a scanned gargoyle

Example: Height patches



Visualization of Euclidean distance between current height patch (white) to all other patches

Outline

Introduction

- Motivation
- Related Work

Methods

- Finite Weighted Graphs
- Graph Construction for 3D Point Cloud Data
- Partial Difference Operators on Weighted Graphs

Applications

- 3D Point Cloud Processing
- Machine Learning

Weighted finite differences

Let (V, E, w) be a weighted graph and let $f\colon V \to \mathbb{R}^m$ be a vertex function. The weighted finite difference $d_w\colon H(V) \to H(E)$ of $f \in H(V)$ along an edge $(u, v) \in E$ is given as:

$$d_w f(u,v) = \sqrt{w(u,v)}(f(v) - f(u)) \tag{4}$$

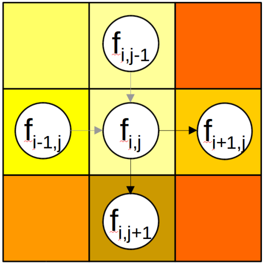
Then the weighted gradient of f in a vertex $u \in V$ is given as:

$$\nabla_w f(u) = (\partial_v f(u))_{v \in V} \quad \text{with} \quad \partial_v f(u) = d_w f(u,v) \tag{5}$$

Special case: local image processing

Let $G = (V, E, w)$ be a directed 2-neighbour grid graph with:

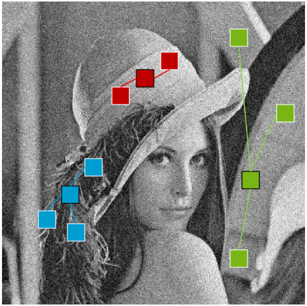
$$\partial_v f(u) = \sqrt{w(u,v)}(f(v) - f(u)) \quad \text{and} \quad w(u,v) = \begin{cases} \frac{1}{h^2}, & \text{if } u \sim v \\ 0, & \text{else} \end{cases}$$



$\rightarrow$ Weighted finite differences correspond to forward differences!

Special case: nonlocal image processing

Idea: Relate vertices based on feature similarity and not only proximity.



→ Graph framework allows to unify local and nonlocal methods!

Special case: nonlocal image processing

Idea: Relate vertices based on feature similarity and not only proximity.

[33] A. Buades: *A Nonlocal Algorithm for Image Denoising*. CVPR (2005)
[34] G. Gilboa, S. Osher: *Nonlocal Operators with Applications to Image Processing*. Multiscale Model. Simul. 7 (2008)
[35] B. Dong, J. Ye, S. Osher, I Dinov: *Level Set Based Nonlocal Surface Restauration* Multiscale Model. Simul., 7 (2008)
[36] Y. Lou, X. Zhang, S. Osher, A. Bertozzi: *Image Recovery via Nonlocal Operators*. J. Sci. Comp. 42 (2009)
[37] Y. van Gennip, N. Guillen, B. Osting, A.L. Bertozzi: *Mean Curvature, Threshold Dynamics, and Phase Field Theory on Finite Graphs*. Milan Journal of Mathematics 82 (2014)
[38] Q. Du, M. Gunzburger, R.B. Lehoucq, K. Zhou: *A Nonlocal Vector Calculus, Nonlocal Volume-Constrained Problems, and Nonlocal Balance Laws*. Math. Models Methods Appl. Sci. 23 (20013)
[39] C. Louchet, L. Moisan: *Total Variation as a Local Filter*. SIAM J. Imaging Sci. 4 (2011)
[40] A. Sawatzky: *(Nonlocal) Total Variation in Medical Imaging*. Ph.D. thesis at WWU Münster (2011)
[41] C. Sutour, C. Deledalle, J.F. Aujol: *Adaptive Regularization of the NL-Means: Application to Image and Video Denoising*. IEEE TIP 23 (2014)
[42] J.F. Aujol, G. Gilboa, N. Papadakis: *Nonlocal Total Variation Spectral Framework*. SSMV (2015)
[43] O. Lézoray, Leo Grady: *Image Processing and Analysis with Graphs: Theory and Practice*. CRC Press (2012)

Adjoint operator and divergence

Let $f \in H(V)$ be a vertex function and let $G \in H(E)$ be an edge function.

One can deduce the adjoint operator $d_w^* : H(E) \to H(V)$ of $d_w : H(V) \to H(E)$ by the following property:

$$\langle d_w f, G \rangle_{H(E)} = \langle f, d_w^* G \rangle_{H(V)} \tag{6}$$

Then the divergence $\operatorname{div}_w : H(E) \to H(V)$ of G in a vertex $u \in V$ is given as:

$$\operatorname{div}_w G(u) = -d_w^* G(u) = \sum_{v \sim u} \sqrt{w(u,v)} (G(u,v) - G(v,u)) \tag{7}$$

We have in particular the following conservation law:

$$\sum_{u \in V} \operatorname{div}_w G(u) = 0 \tag{8}$$

Higher-order operators

Let $f \in H(V)$ be a vertex function on a graph $G(V, E, w)$ and $1 \leq p < \infty$.

The isotropic graph p-Laplacian operator in an vertex $u \in V$ is given as:

$$\Delta_{w,p}^{i} f(u) = \frac{1}{2} \operatorname{div}_w \left(||\nabla_w f||^{p-2} d_w f \right)(u) \tag{9}$$

The anisotropic graph p-Laplacian operator in an vertex $u \in V$ is given as:

$$\begin{aligned} \Delta_{w,p}^{a} f(u) &= \frac{1}{2} \operatorname{div}_w \left(|d_w f|^{p-2} d_w f \right)(u) \\ &= \sum_{v \sim u} (w(u,v))^{p/2} |f(v) - f(u)|^{p-2} (f(v) - f(u)) \end{aligned} \tag{10}$$

[44] A. Elmoataz, O. Lézoray, S. Bougleux: *Nonlocal Discrete Regularization on Weighted Graphs: a Framework for Image and Manifold Processing*. IEEE TIP 17 (2008)

Higher-order operators

We introduce the weighted upwind gradient to mimic upwind differences:

$$\begin{aligned} \nabla^+ f(u) &= (\partial_v^+ f(u))_{v\in V} = (\max(0, \partial_v f(u)))_{v\in V} \\ \nabla^- f(u) &= (\partial_v^- f(u))_{v\in V} = (\min(0, \partial_v f(u)))_{v\in V} \end{aligned} \tag{11}$$

The graph ∞-Laplacian operator in an vertex $u \in V$ is given as:

$$\Delta_{w,\infty} f(u) = \frac{1}{2}(||\nabla_w^+ f(u)||_\infty - ||\nabla_w^- f(u)||_\infty) \tag{12}$$

[45] A. Elmoataz, X. Desquesnes, O. Lézoray: *Non-Local Morphological PDEs and p-Laplacian Equation on Graphs With Applications in Image Processing and Machine Learning*. IEEE Selected Topics in Sig. Proc. 6 (2012)

Connection to discretization schemes

Choosing the right graph construction and weight function we recover:

Discretization schemes for local Laplacian operators:

- Discretization of the anisotropic p-Laplacian
- Obermann discretization of the ∞-Laplacian

Discretization schemes for nonlocal Laplacian operators:

- Discretization of the nonlocal fractional p-Laplacian
- Discretization of the nonlocal Hölder ∞-Laplacian

Discretization schemes for gradient operators:

- Osher-Sethian upwind discretization scheme
- Gudonov discretization scheme

[46] A. Elmoataz, M. Toutain, D. Tenbrinck: *On the p-Laplacian and ∞-Laplacian on Graphs with Applications in Image and Data Processing*. SIAM Journal on Imaging Sciences 8 (2015) .

Consistency results

The transition from finite weighted graphs to the continuum is an active field of research with many open questions:

- graph construction?
- correct scaling rates?
- consistency of weighted graph operators?
- convergence of solutions?

[47] M. Belkin, J. Sun, Y. Wang: *Discrete Laplace Operator on Meshed Surfaces*. SOCG (2008)
[48] M. Belkin, J. Sun, Y. Wang: *Constructing Laplace Operator from Point Clouds in* $\mathbb{R}^d$. SODA (2009)
[49] M. Belkin, P. Niyogi: *Convergence of Laplacian Eigenmaps*. NIPS 19 (2006)
[50] M. Belkin, P. Niyogi: *Towards a Theoretical Foundation for Laplacian-Based Manifold Methods*. J. Comput. System Sci. 74 (2008)
[51] U. von Luxburg, M. Belkin, O. Bousquet: *Consistency of Spectral Clustering*. The Annals of Statistics 36 (2008)

Consistency results

The transition from finite weighted graphs to the continuum is an active field of research with many open questions:

- graph construction?
- correct scaling rates?
- consistency of weighted graph operators?
- convergence of solutions?

[52] E. Arias-Castro, B. Pelletier, P. Pudlo: *The Normalized Graph Cut and Cheeger Constant: From Discrete to Continuous*. Advances in Applied Probability 44 (2012)
[53] N. Garcia Trillos, D. Slepcev, J. von Brecht, T. Laurent, X. Bresson: *Consistency of Cheeger and Ratio Graph Cuts*. arXiv:1411.6590 (2014)
[54] N. Garcia Trillos, D. Slepcev: *Continuum Limit of Total Variation on Point Clouds*. arXiv:1403.6355 (2014)
[55] N. Garcia Trillos, D. Slepcev: *A Variational Approach to the Consistency of Spectral Clustering*. arXiv:1508.01928 (2015)
[56] Z. Shi, J. Sun: *Convergence of Laplacian Spectra from Point Clouds*. arXiv:1506.01788 (2015)

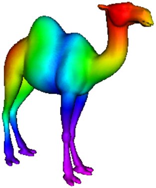
Translating PDEs to weighted graphs

Example: Heat equation

Let $G(V, E, w)$ be a weighted graph and let $f : V \times [0, T] \to \mathbb{R}$ be a vertex function. One important partial difference equation (PdE) is a surface graph diffusion process of the form:

$$\begin{cases} \frac{\partial f(u,t)}{\partial t} &= \Delta_w f(u,t) \,, \\ f(u, t = 0) &= f_0(u) \,, \end{cases} \tag{13}$$

for which $f_0 : V \to \mathbb{R}$ is the initial value of f at time $t = 0$.



Outline

Introduction

- Motivation
- Related Work

Methods

- Finite Weighted Graphs
- Graph Construction for 3D Point Cloud Data
- Partial Difference Operators on Weighted Graphs

Applications

- 3D Point Cloud Processing
- Machine Learning

Diffusion processes on graphs

The initial value problem for graph p-Laplacian diffusion is given as:

$$\begin{cases} \frac{\partial f(u,t)}{\partial t} & = \Delta_{w,p}^{a} f(u,t) \, , \\ f(u, t=0) & = f_0(u) \, , \end{cases} \tag{14}$$

Applying forward Euler time discretization leads to an iterative scheme:

$$f^{n+1}(u) \;=\; f^n(u) + \Delta t \sum_{v \sim u} (w(u,v)^{p/2} |f(v) - f(u)|^{p-2} (f(v) - f(u)) \tag{15}$$

l^∞-norm stability can be guaranteed under the following CFL condition:

$$1 \;\geq\; \Delta t \sum_{v \sim u} (w(u,v)^{p/2} |f(v) - f(u)|^{p-2} \tag{16}$$

Geometric filtering using height patches

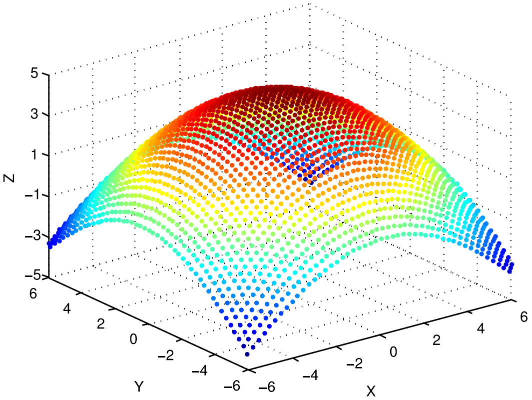
Problem formulation:

- $G = (V, E, w)$ is symmetric k-NN proximity graph
- weight function $w \colon E \to [0, 1]$ is based on height patch similarity
- $f_0 \colon V \to \mathbb{R}^3$ represents coordinates of 3D points

Algorithm: Geometric filtering

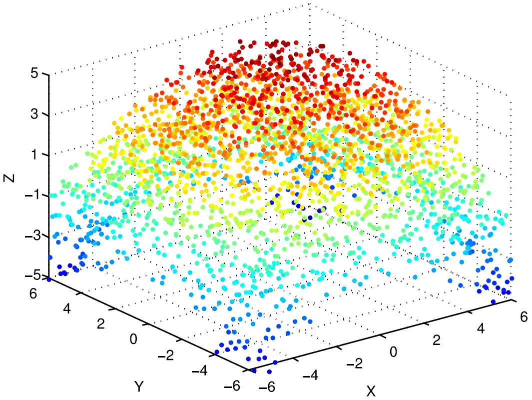
1. Filter f_0 by solving the diffusion equations in (15)

Geometric filtering using height patches



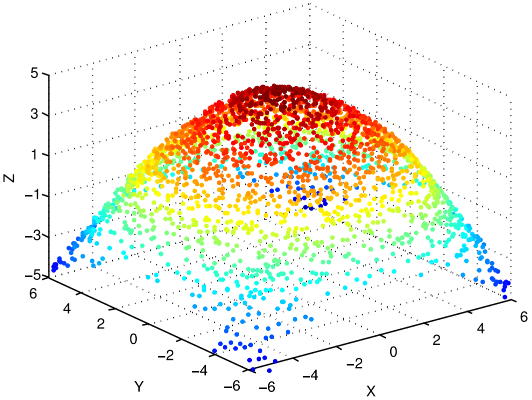
Original 3D point cloud without noise

Geometric filtering using height patches



3D point cloud perturbed by Gaussian noise

Geometric filtering using height patches



Denoised 3D point cloud

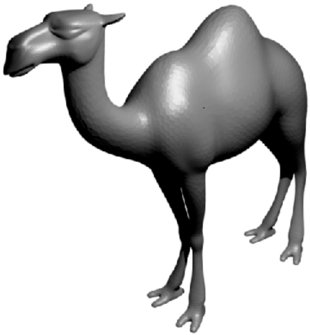
Geometric filtering using height patches

Original data

Noisy data

Geometric filtering using height patches

Original data



Filtered data

Geometric denoising for data simplification

Problem formulation:

- $G = (V, E, w)$ is symmetric k-NN proximity graph
- weight function $w\colon E \to [0, 1]$ is based on Euclidean distance
- $f_0\colon V \to \mathbb{R}^3$ represents coordinates of 3D points

Algorithm: Data simplification [21, 30]

1. Filter f_0 by solving the diffusion equations in (15)
2. Remove redundant points from the 3D point cloud data

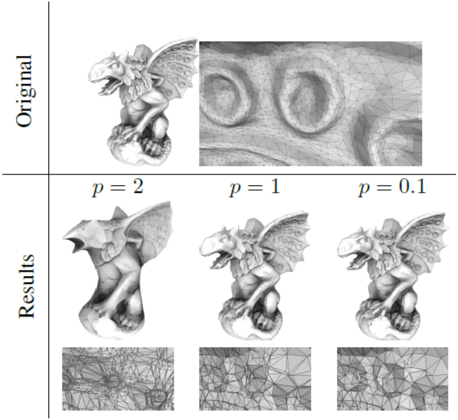
[32] F. Lozes, A.Elmoataz, O. Lezoray: *Partial Difference Operators on Weighted Graphs for Image Processing on Surfaces and Point Clouds*. IEEE TIP 23 (2014)
[57] F. Lozes, A. Elmoataz, O. Lezoray: *PDE-Based Graph Signal Processing for 3-D Color Point Clouds: Opportunities for Cultural Heritage*. IEEE SPMag (2015)

Geometric denoising for data simplification

Original 3D point cloud data
(∼201.000 points)

Simplified 3D point cloud data
(∼45.000 points) → 77.6% reduction

Geometric denoising for data simplification



Mesh visualization of simplified data for different *p* values

Color denoising

Problem formulation:

- $G = (V, E, w)$ is symmetric k-NN proximity graph with:
  $\rightarrow k = 8$ (local)
  $\rightarrow k = 1000$ (nonlocal)
- weight function $w\colon E \rightarrow [0, 1]$ is either:
  $\rightarrow$ constant with $w(u_i, u_j) = 1$ for all $(u_i, u_j) \in E$ (local)
  $\rightarrow$ based on color patch similarity (nonlocal)
- $f_0 : V \rightarrow \mathbb{R}^3$ represents RGB color vector of that point

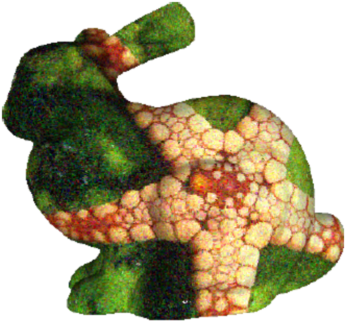
Algorithm: Color denoising [21]

1. Filter f_0 by solving the diffusion equations in (15)

[32] F. Lozes, A.Elmoataz, O. Lezoray: *Partial Difference Operators on Weighted Graphs for Image Processing on Surfaces and Point Clouds*. IEEE TIP 23 (2014)

Color denoising

Original 3D point cloud data



Noisy 3D point cloud data

Color denoising

Original 3D point cloud data

Filtered 3D point cloud data (local)

Color denoising

Original 3D point cloud data

Filtered 3D point cloud data (nonlocal)

Discrete optimization problems on graphs

Mimic variational models on graphs as discrete optimization problems.

Example: Rudin-Osher-Fatemi TV denoising model [31]

$$||f||_{TV} = \begin{cases} \sum_{x_i \in \mathcal{V}} ||(\nabla_w f)(x_i)||_p = \sum_{x_i \in \mathcal{V}} \left(\sum_{x_j \sim x_i} |(d_w f)(x_i, x_j)|^p \right)^{1/p}, & 1 \leq p < \infty, \\ \sum_{x_i \in \mathcal{V}} ||(\nabla_w f)(x_i)||_\infty = \sum_{x_i \in \mathcal{V}} \max_{x_j \sim x_i} |(d_w f)(x_i, x_j)|, & p = \infty. \end{cases}$$

For $\lambda > 0$ find a minimizer $u \in H(V)$ of the energy

$$E : H(V) \to \mathbb{R}, \qquad E(u) = \lambda ||u - f||_2^2 + ||u||_{TV}$$

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

Total variation denoising for color denoising

Noisy data

Result (local)
1200 iterations

Result (nonlocal)
1200 iterations

[59] D. Tenbrinck, F. Lozes, A. Elmoataz: *Solving Minimal Surface Problems on Surfaces and Point Clouds*. SSVM (2015)

Total variation for geometric filtering

Noisy data

Filtered data

[59] D. Tenbrinck, F. Lozes, A. Elmoataz: *Solving Minimal Surface Problems on Surfaces and Point Clouds*. SSVM (2015)

Interpolation problems

Another class of PdEs on graphs are interpolation problems of the form:

$$\begin{cases} \Delta_{w,\infty} f(u) = 0 \,, & \text{for } u \in A \,, \\ f(u) = g(u) \,, & \text{for } u \in \partial A \,. \end{cases} \tag{17}$$

for which $A \subset V$ is a subset of vertices and $\partial A = V \backslash A$ and the given information g are application dependent.

Solving this Dirichlet problem amounts in finding the stationary solution of a diffusion process with fixed boundary conditions.

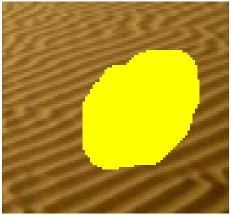
$$\begin{cases} \frac{\partial f(u,t)}{\partial t} = \Delta_{w,\infty} f(u,t) \,, & \text{for } u \in A \,, \\ f(u) = g(u) \,, & \text{for } u \in \partial A \,. \end{cases} \tag{18}$$

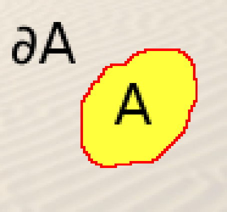
Interpolation problems

1. Use the same numerical scheme as for diffusion processes:

$$f^{n+1}(u) = f^n(u) + \frac{\Delta t}{2}\Big(\max_{v\sim u}\sqrt{w(u,v)}(f(v)-f(u)) + \min_{v\sim u}\sqrt{w(u,v)}(f(u)-f(v))\Big)$$

2. Iterate until no more changes in border zone
3. Add border nodes to ∂A and repeat with 1 until $A = \emptyset$.



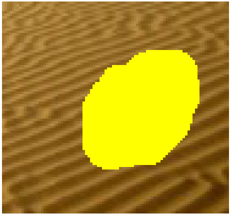


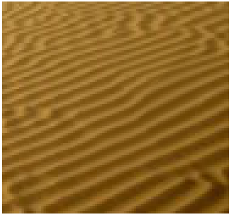
Interpolation problems

1. Use the same numerical scheme as for diffusion processes:

$$f^{n+1}(u) = f^n(u) + \frac{\Delta t}{2}\left(\max_{v\sim u}\sqrt{w(u,v)}(f(v)-f(u)) + \min_{v\sim u}\sqrt{w(u,v)}(f(u)-f(v))\right)$$

2. Iterate until no more changes in border zone
3. Add border nodes to ∂A and repeat with 1 until $A = \emptyset$.





Color inpainting

Problem formulation:

- $G = (V, E, w)$ is symmetric k-NN proximity graph
- weight function $w\colon E \to [0, 1]$ is based on color patch distance
- $f_0\colon V \to \mathcal{C} \subset \mathbb{N}$ is given color on vertices $u \in \partial A^0 \subset V$

Algorithm: Color inpainting [30]

1. Solve interpolation problem (18) for vertices $u \in \partial^- A^n$ in border region
2. Add these vertices to set ∂A^n
3. Set $n \to n + 1$ and repeat algorithm until $A^n = \emptyset$

[57] F. Lozes, A. Elmoataz, O. Lezoray: *PDE-Based Graph Signal Processing for 3-D Color Point Clouds: Opportunities for Cultural Heritage*. IEEE SPMag (2015)

Color inpainting

3D point cloud of a scanned person

User-defined region for color inpainting

Color inpainting

3D point cloud of a scanned person

Result of color inpainting (local)

Color inpainting

3D point cloud of a scanned person

Result of color inpainting (nonlocal)

Color inpainting

3D point cloud of a scanned ancient wall

User-defined region for color inpainting

Color inpainting

3D point cloud of a scanned ancient wall

Result of color inpainting

Geometric inpainting

Problem formulation:

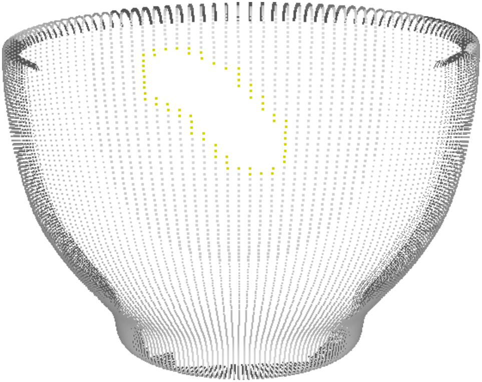
- $G = (V, E, w)$ is symmetric k-NN proximity graph
- weight function $w \colon E \to [0, 1]$ is based on height patch distance
- $f_0 \colon V \to \mathbb{R}^3$ represents coordinates of 3D points

Algorithm: Geometric inpainting [30]

1. Solve interpolation problem (18) for vertices $u \in \partial^- A^n$ in border region
2. Add these vertices to set ∂A^n
3. Set $n \to n + 1$ and repeat algorithm until $A^n = \emptyset$

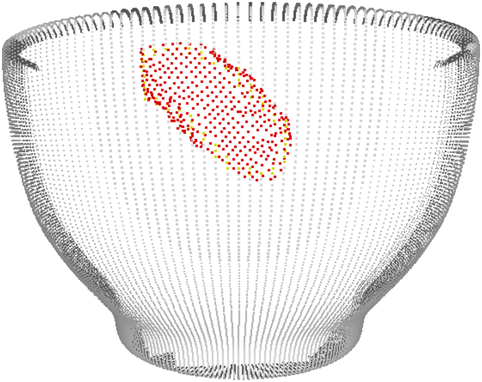
[57] F. Lozes, A. Elmoataz, O. Lezoray: *PDE-Based Graph Signal Processing for 3-D Color Point Clouds: Opportunities for Cultural Heritage*. IEEE SPMag (2015)

Geometric inpainting



3D point cloud of a scanned cup with artificial hole in wall

Geometric inpainting

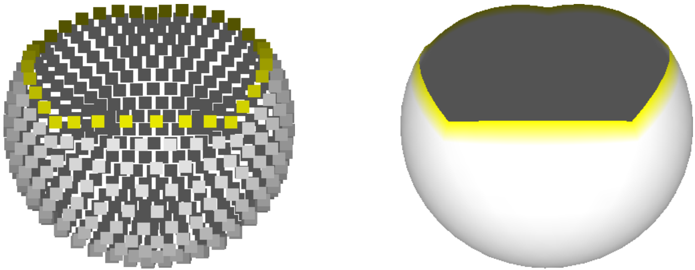


Filled in points (red) after geometric inpainting

Geometric inpainting

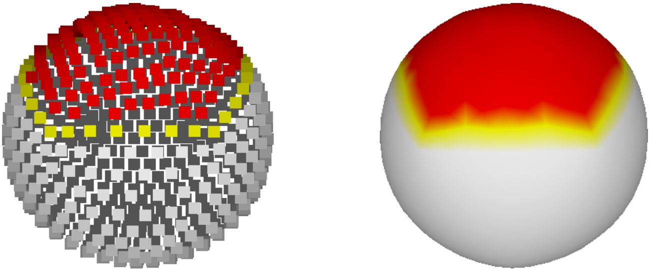
Rendering of inpainted cup

Geometric inpainting



3D point cloud of an incomplete sphere (left) and rendered visualization (right)

Geometric inpainting



3D point cloud of inpainted sphere (left) and rendered visualization (right)

Geometric inpainting

Scanned vase with hole

Geometric inpainting result

Geometric inpainting

Scanned vase with hole

Color inpainting result

Colorization

Problem formulation:

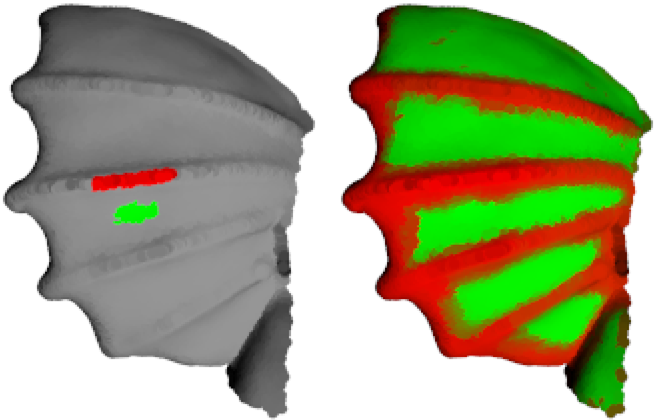
- $G = (V, E, w)$ is symmetric k-NN proximity graph
- weight function $w\colon E \to [0, 1]$ is based on height patch distance
- $f_0\colon V \to \mathcal{C} \subset \mathbb{N}$ is given color on vertices $u \in \partial A^0 \subset V$

Algorithm: Point cloud colorization [30]

1. Solve interpolation problem (18) for vertices $u \in \partial^- A^n$ in border region
2. Add these vertices to set ∂A^n
3. Set $n \to n + 1$ and repeat algorithm until $A^n = \emptyset$

[30] F. Lozes, A. Elmoataz, O. Lezoray: *PDE-Based Graph Signal Processing for 3-D Color Point Clouds: Opportunities for Cultural Heritage*. IEEE SPMag (2015)

Colorization

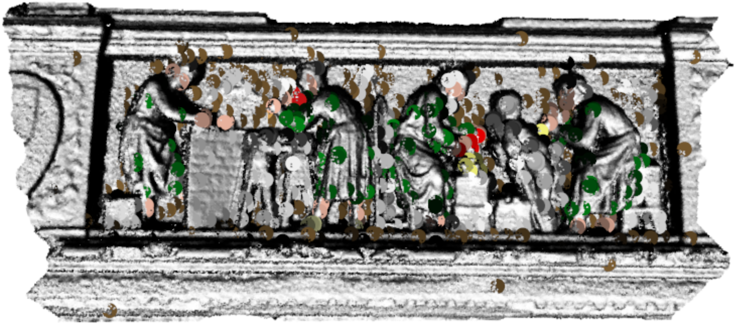


User-defined color scribbles on 3D point cloud (left) and colorization result (right)

Colorization

User-defined color scribbles on 3D point cloud (left) and colorization result (right)

Colorization



User-defined color scribbles on 3D point cloud

Colorization

Colorization result

Colorization

User-defined color scribbles on 3D point cloud (left) and colorization result (right)

Computing general distances on point clouds

Translate nonlinear PDEs in order to compute general distances on graphs:

$$\begin{cases} \|\nabla_w^- f(u)\|_p = g(u)\,, & \text{for } u \in A\,, \\ f(u) = 0\,, & \text{for } u \in \partial A\,. \end{cases} \tag{19}$$

- Existence and uniqueness of solutions proved
- Efficient numerical solver based on "Fast Marching" [62]

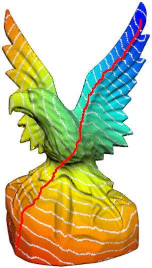
[60] X. Desquesnes, A. Elmoataz, O. Lézoray: *Eikonal Equation Adaptation on Weighted Graphs: Fast Geometric Diffusion Process for Local and Non-Local Image and Data Processing*. JMIV (2013)
[61] M. Toutain, A. Elmoataz, F. Lozes, A. Mansouri: *Nonlocal Discrete ∞-Poisson and Hamilton Jacobi Equations: from Stochastic Games to Generalized Distances on Images, Meshes, and Point Clouds*. JMIV, in press (2015)
[62] J. N. Tsitsiklis: *Efficient Algorithms for Globally Optimal Trajectories*. IEEE TAC 40 (1995)

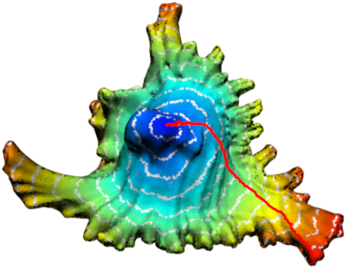
Computing general distances on point clouds

Special case: Eikonal equation for $g(u) \equiv 1$ and $p = 2$

We compute shortest paths on 3D point clouds by solving:

$$\begin{cases} \|\nabla_w^- f(u)\|_2 = 1\,, & \text{for } u \in A\,, \\ f(u) = 0\,, & \text{for } u \in \partial A\,. \end{cases}$$



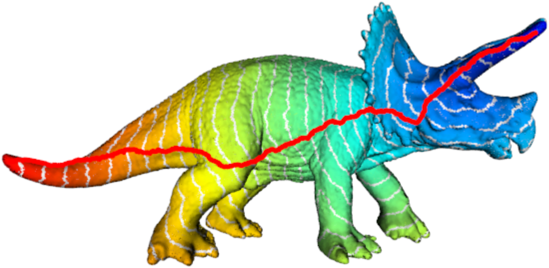


Computing general distances on point clouds

Special case: Eikonal equation for $g(u) \equiv 1$ and $p = 2$

We compute shortest paths on 3D point clouds by solving:

$$\begin{cases} ||\nabla_w^- f(u)||_2 = 1\,, & \text{for } u \in A\,, \\ f(u) = 0\,, & \text{for } u \in \partial A\,. \end{cases}$$

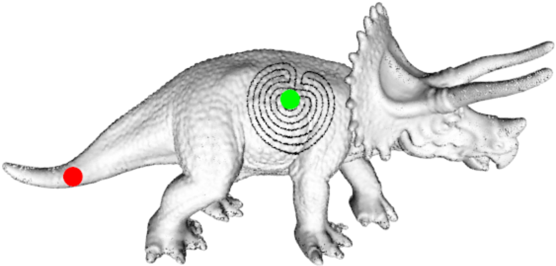


Computing general distances on point clouds

Special case: Eikonal equation for $g(u) \equiv 1$ and $p = 2$

We compute shortest paths on 3D point clouds by solving:

$$\begin{cases} \|\nabla_w^- f(u)\|_2 = 1\,, & \text{for } u \in A\,, \\ f(u) = 0\,, & \text{for } u \in \partial A\,. \end{cases}$$

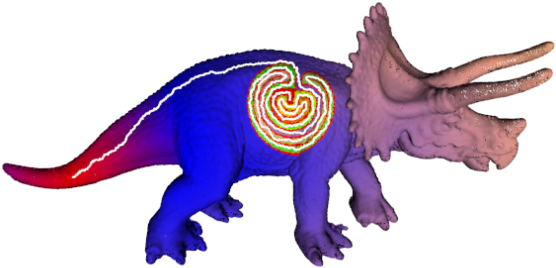


Computing general distances on point clouds

Special case: Eikonal equation for $g(u) \equiv 1$ and $p = 2$

We compute shortest paths on 3D point clouds by solving:

$$\begin{cases} \|\nabla_w^- f(u)\|_2 = 1\,, & \text{for } u \in A\,, \\ f(u) = 0\,, & \text{for } u \in \partial A\,. \end{cases}$$



Semi-supervised segmentation

Problem formulation:

- $G = (V, E, w)$ is symmetric k-NN proximity graph
- weight function $w\colon E \to [0, 1]$ is based on color patch distance
- $f_0\colon V \to \mathcal{L} \subset \mathbb{N}$ is given labels on vertices $u \in \partial A^0 \subset V$

Algorithm: Semi-supervised segmentation [30]

1. Solve interpolation problem (18) for vertices $u \in \partial^- A^n$ in border region
2. Add these vertices to set ∂A^n
3. Set $n \to n + 1$ and repeat algorithm until $A^n = \emptyset$

[57] F. Lozes, A. Elmoataz, O. Lezoray: *PDE-Based Graph Signal Processing for 3-D Color Point Clouds: Opportunities for Cultural Heritage*. IEEE SPMag (2015)

Semi-supervised segmentation

Original data User labels Segmented data

Outline

Introduction

- Motivation
- Related Work

Methods

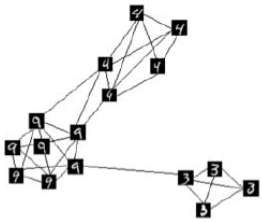
- Finite Weighted Graphs
- Graph Construction for 3D Point Cloud Data
- Partial Difference Operators on Weighted Graphs

Applications

- 3D Point Cloud Processing
- Machine Learning

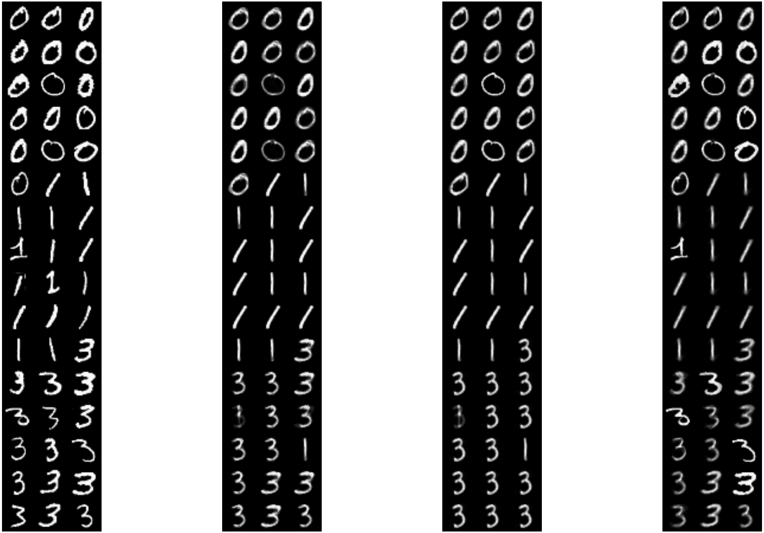
Digit classification

- use MNIST digit database [35] for semi-supervised classification
- build a graph using two-sided tangent distance of digit patches
- use weighted graph framework for classification



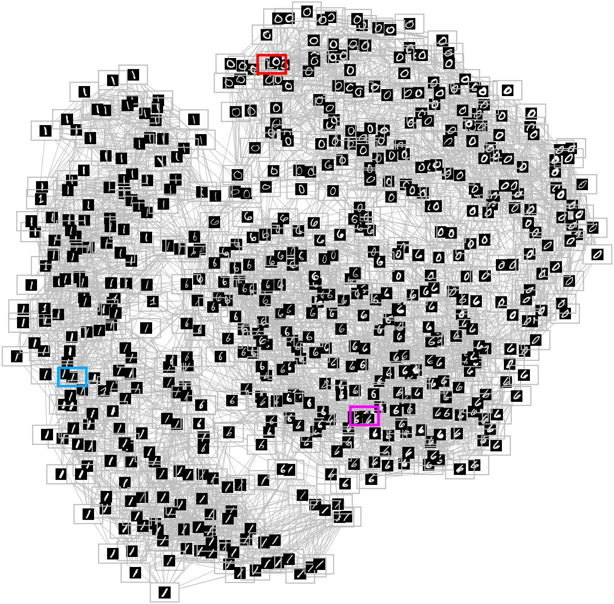
[63] Y. LeCun, C. Cortes, C. Burges: *The MNIST database of handwritten digits*. http://yann.lecun.com/exdb/mnist/ [64] M. Toutain, A. Elmoataz, O. Lézoray: *Geometric PDEs on Weighted Graphs for Semi-Supervised Classification*. ICMLA (2014)

p-Laplacian diffusion for pre-processing

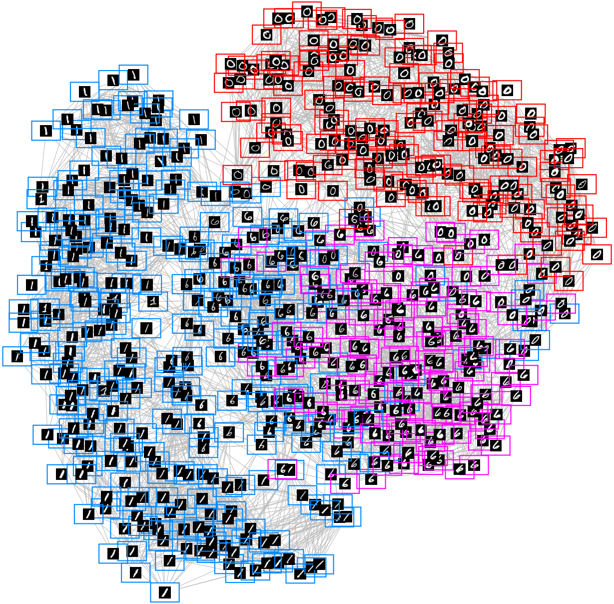


Original $\quad$ $p = 1$ $\quad$ $p = 2$ $\quad$ $p = \infty$

Eikonal equation for semi-supervised classification

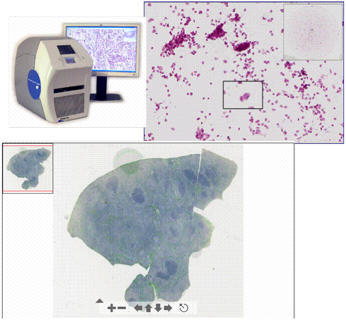


Eikonal equation for semi-supervised classification



Cell classification in histology

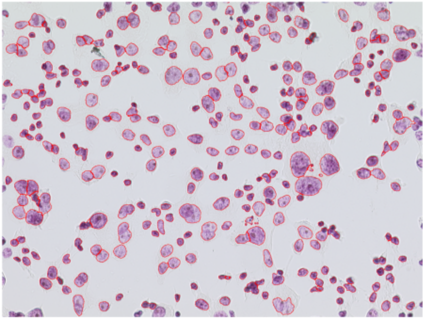
Classify different cell types using very few expert interactions
→ especially: healthy and pathological cells for cervical cancer.



[65] Datexim. http://www.datexim.com/en/

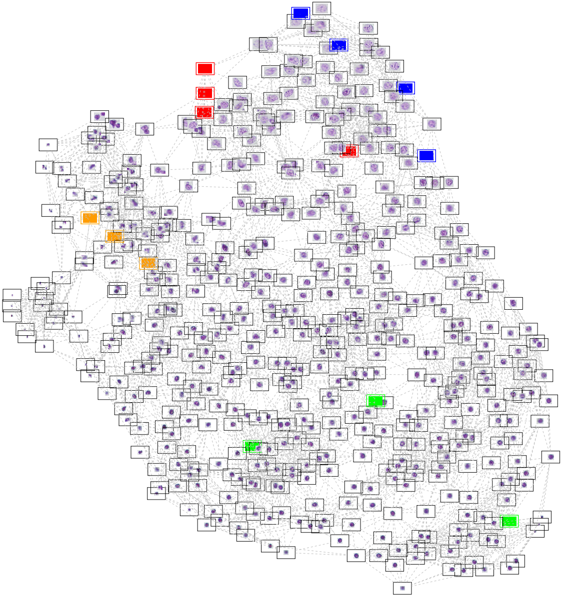
Cell classification in histology

Classify different cell types using very few expert interactions
→ especially: healthy and pathological cells for cervical cancer.

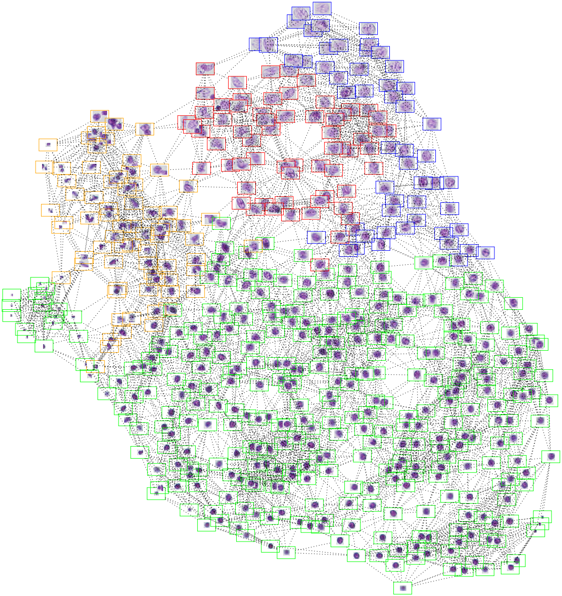


[65] Datexim. http://www.datexim.com/en/

Cell classification in histology

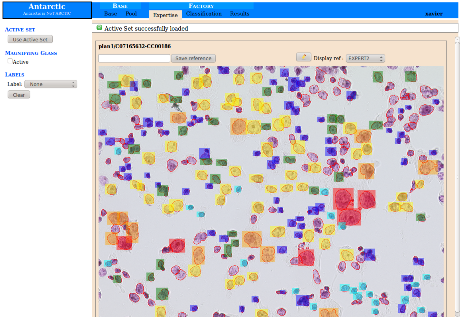


Cell classification in histology



Cell classification in histology

Classify different cell types using very few expert interactions
→ especially: healthy and pathological cells for cervical cancer.



[37] Datexim. http://www.datexim.com/en/

Summary

1. Graph framework unifies local and nonlocal methods
2. Patch construction for 3D point cloud data introduced
3. PDEs / variational models translated to data of arbitrary topology
4. Experimental results were demonstrated for:
    - Filtering of color and geometry
    - Data simplification
    - Inpainting for color and geometry
    - Colorization
    - Semi-supervised segmentation / classification

Thank you for your attention!

Any questions?