Partial Difference Operators on Weighted Graphs for Point Cloud Processing
From 3D Surfaces to Machine Learning

Abderrahim Elmoataz\textsuperscript{1,2}, Francois Lozes\textsuperscript{1}, Daniel Tenbrinck\textsuperscript{3}

\textsuperscript{1}Université de Caen Normandie, Caen, France
\textsuperscript{2}Université de Paris-Est Marne-la-Vallée, Paris, France
\textsuperscript{3}Westfälische Wilhelms-Universität Münster, Münster, Germany
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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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].

Recent technical advances

**Today:** 3D surface data acquisition with Microsoft Kinect devices

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Recent technical advances

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Recent technical advances

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

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]

New applications: Digital archaeology

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
New applications: Digital archaeology

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

[5] CyArk, non-profit organization (http://www.cyark.org/)

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Example: Social networks

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


Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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

Daniel Tenbrinck: "*Partial Difference Operators on Weighted Graphs for Point Cloud Processing*"
Yale face database for training/evaluation of face recognition algorithms [7]

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

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

Daniel Tenbrinck:  "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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)

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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)


Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Related work: Intrinsic methods

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

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

\[ \text{div}_M^d \nabla(p_i) = \frac{1}{\sum_l \text{Area}(T_l)} \sum_l \text{Area}(T_l) \text{div}_{T_l}^d \nabla(p_0) \]

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Related work: Intrinsic methods

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


Daniel Tenbrinck: "*Partial Difference Operators on Weighted Graphs for Point Cloud Processing*"
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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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 $w : E \rightarrow [0, 1]$
Vertex and edge functions

A vertex function \( f : V \to \mathbb{R}^m \) 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 \( f : V \rightarrow \mathbb{R}^m \) as:

\[
\int_V f = \sum_{u \in V} f(u)
\]

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

\[
\|f\|_p = \begin{cases} 
(\sum_{u \in V} \|f(u)\|^p)^{1/p}, & 1 \leq p < \infty \\
\max_{u \in V} \|f(u)\|, & p = \infty
\end{cases}
\]

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

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Choice of weight function

Construct weight functions \( w : E \rightarrow [0, 1] \) based on similarity of features.

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

- a symmetric distance function \( d : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}^+ \)
  
  e.g., Euclidean distance, patch distance, …

- a normalized similarity function \( s : \mathbb{R}^+ \rightarrow [0, 1] \)
  
  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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
General graph construction

$\epsilon$-ball graph

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
General graph construction

k-Nearest-Neighbor graph (k=2)

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
General graph construction

Distribution of 2D points
General graph construction

k-NN graph for $k=3$

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
General graph construction

k-NN graph for $k=15$

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric graph construction for 3D point clouds

Colored 3D point cloud data of a scanned chair

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric graph construction for 3D point clouds

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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

Observation:
Comparing patches implicitly induces local regularity!
→ How to construct patches on 3D point cloud data?


Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Patch construction for 3D point clouds

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

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Weighted finite differences

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

\[
d_wf(u, v) = \sqrt{w(u, v)}(f(v) - f(u))
\]  

(4)

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

\[
\nabla_w f(u) = (\partial_v f(u))_{v \in V} \quad \text{with} \quad \partial_v f(u) = d_wf(u, v)
\]  

(5)
Special case: local image processing

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

$$\partial_v f(u) = \sqrt{w(u, v)}(f(v) - f(u))$$

and

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

→ Weighted finite differences correspond to forward differences!

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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*!

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Special case: nonlocal image processing

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


Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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 \( \text{div}_w : H(E) \to H(V) \) of \( G \) in a vertex \( u \in V \) is given as:

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

We have in particular the following conservation law:

\[
\sum_{u \in V} \text{div}_w G(u) = 0 \tag{8}
\]
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:

\[
\Delta^i_{w,p} f(u) = \frac{1}{2} \text{div}_w (||\nabla_w f||^{p-2} d_w f) (u)
\]

(9)

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

\[
\Delta^a_{w,p} f(u) = \frac{1}{2} \text{div}_w (|d_w f|^{p-2} d_w f) (u)
\]

\[
= \sum_{v \sim u} (w(u, v))^{p/2} |f(v) - f(u)|^{p-2} (f(v) - f(u))
\]

(10)

Higher-order operators

We introduce the weighted upwind gradient to mimic upwind differences:

\[
\nabla^+ f(u) = (\partial^+ v f(u))_{v \in V} = (\max(0, \partial_v f(u)))_{v \in V}
\]

\[
\nabla^- f(u) = (\partial^- v f(u))_{v \in V} = (\min(0, \partial_v f(u)))_{v \in V}
\]

(11)

The graph \(\infty\)-Laplacian operator in an vertex \(u \in V\) is given as:

\[
\Delta_{w, \infty} f(u) = \frac{1}{2} (||\nabla^+_w f(u)||_\infty - ||\nabla^-_w f(u)||_\infty)
\]

(12)

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 $\infty$-Laplacian

Discretization schemes for nonlocal Laplacian operators:
- Discretization of the nonlocal fractional $p$-Laplacian
- Discretization of the nonlocal Hölder $\infty$-Laplacian

Discretization schemes for gradient operators:
- Osher-Sethian upwind discretization scheme
- Gudonov discretization scheme

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?

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?


Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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:

\[
\begin{align*}
\frac{\partial f(u, t)}{\partial t} &= \Delta_w f(u, t), \\
 f(u, t = 0) &= f_0(u),
\end{align*}
\]

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Diffusion processes on graphs

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

\[
\begin{aligned}
\frac{\partial f(u,t)}{\partial t} &= \Delta_{w,p}^{a} f(u,t), \\
f(u, t = 0) &= f_{0}(u),
\end{aligned}
\]  

(14)

Applying forward Euler time discretization leads to an iterative scheme:

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

(15)

$l^{\infty}$-norm stability can be guaranteed under the following CFL condition:

\[
1 \geq \Delta t \sum_{v \sim u} (w(u, v)^{p/2}|f(v) - f(u)|^{p-2}
\]

(16)
Geometric filtering using height patches

Problem formulation:

- $G = (V, E, w)$ is symmetric $k$-NN proximity graph
- weight function $w : E \rightarrow [0, 1]$ is based on height patch similarity
- $f_0 : V \rightarrow \mathbb{R}^3$ 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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric filtering using height patches

3D point cloud perturbed by Gaussian noise

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric filtering using height patches

Denoised 3D point cloud

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric filtering using height patches

Original data

Noisy data

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric filtering using height patches

Original data

Filtered data

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric denoising for data simplification

Problem formulation:

- $G = (V, E, w)$ is symmetric $k$-NN proximity graph
- weight function $w : E \rightarrow [0, 1]$ is based on Euclidean distance
- $f_0 : V \rightarrow \mathbb{R}^3$ 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

Geometric denoising for data simplification

Original 3D point cloud data
(∼201,000 points)

Simplified 3D point cloud data
(∼45,000 points) → 77.6% reduction

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric denoising for data simplification

Mesh visualization of simplified data for different $p$ values

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Color denoising

Problem formulation:

- $G = (V, E, w)$ is symmetric $k$-NN proximity graph with:
  - $k = 8$ (local)
  - $k = 1000$ (nonlocal)
- weight function $w : E \rightarrow [0, 1]$ is either:
  - constant with $w(u_i, u_j) = 1$ for all $(u_i, u_j) \in E$ (local)
  - 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)

Color denoising

Original 3D point cloud data

Noisy 3D point cloud data

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Color denoising

Original 3D point cloud data
Filtered 3D point cloud data (local)

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Color denoising

Original 3D point cloud data

Filtered 3D point cloud data (nonlocal)

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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

\[ E : H(V) \rightarrow \mathbb{R}, \quad E(u) = \lambda ||u - f||_2^2 + ||u||_{TV} \]


Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Total variation denoising for color denoising

Noisy data
Result (local) 1200 iterations
Result (nonlocal) 1200 iterations

Total variation for geometric filtering

Noisy data

Filtered data

Interpolation problems

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

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

for which \( A \subset V \) is a subset of vertices and \( \partial A = V \setminus A \) 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.

\[
\begin{aligned}
\frac{\partial f(u,t)}{\partial t} &= \Delta_{w,\infty} f(u,t), & \text{for } u \in A, \\
f(u) &= g(u), & \text{for } u \in \partial A.
\end{aligned}
\] (18)
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 \( \partial A \) and repeat with 1 until \( A = \emptyset \).
Interpolation problems

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

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

\[ + \min_{v \sim u} \sqrt{w(u, v)} (f(u) - f(v)) \]

2. Iterate until no more changes in border zone

3. Add border nodes to \( \partial A \) and repeat with 1 until \( A = \emptyset \).
Color inpainting

Problem formulation:
- $G = (V, E, w)$ is symmetric $k$-NN proximity graph
- weight function $w : E \to [0, 1]$ is based on color patch distance
- $f_0 : V \to C \subset \mathbb{N}$ 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 $\partial A^n$
3. Set $n \to n + 1$ and repeat algorithm until $A^n = \emptyset$


Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Color inpainting

3D point cloud of a scanned person

User-defined region for color inpainting

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Color inpainting

3D point cloud of a scanned person

Result of color inpainting (local)

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Color inpainting

3D point cloud of a scanned person

Result of color inpainting (nonlocal)

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Color inpainting

3D point cloud of a scanned ancient wall

User-defined region for color inpainting

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Color inpainting

3D point cloud of a scanned ancient wall

Result of color inpainting

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric inpainting

Problem formulation:

- $G = (V, E, w)$ is symmetric $k$-NN proximity graph
- weight function $w: E \rightarrow [0, 1]$ is based on height patch distance
- $f_0: V \rightarrow \mathbb{R}^3$ 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 $\partial A^n$
3. Set $n \rightarrow n + 1$ and repeat algorithm until $A^n = \emptyset$

Geometric inpainting

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric inpainting

Filled in points (red) after geometric inpainting

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric inpainting

Rendering of inpainted cup

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric inpainting

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric inpainting

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric inpainting

Scanned vase with hole

Geometric inpainting result

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Geometric inpainting

Scanned vase with hole

Color inpainting result

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Colorization

Problem formulation:

- $G = (V, E, w)$ is symmetric $k$-NN proximity graph
- weight function $w: E \rightarrow [0, 1]$ is based on height patch distance
- $f_0: V \rightarrow C \subset \mathbb{N}$ 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 $\partial A^n$
3. Set $n \rightarrow n + 1$ and repeat algorithm until $A^n = \emptyset$

Colorization

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Colorization

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Colorization

User-defined color scribbles on 3D point cloud

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Colorization result

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Colorization

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Computing general distances on point clouds

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

\[
\begin{align*}
\| \nabla_w f(u) \|_p &= g(u) , & \text{for } u \in A , \\
f(u) &= 0 , & \text{for } u \in \partial A .
\end{align*}
\]  

(19)

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

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:

$$\left\{ \begin{array}{c}
\|\nabla_w f(u)\|_2 = 1, \\
f(u) = 0,
\end{array} \right. \quad \text{for } u \in A,$$

$$f(u) = 0, \quad \text{for } u \in \partial A.$$
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:

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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:

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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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:

\[
\begin{align*}
\| \nabla_w f(u) \|_2 &= 1, & \text{for } u \in A, \\
f(u) &= 0, & \text{for } u \in \partial A.
\end{align*}
\]
Semi-supervised segmentation

Problem formulation:
- $G = (V, E, w)$ is symmetric $k$-NN proximity graph
- weight function $w : E \rightarrow [0, 1]$ is based on color patch distance
- $f_0 : V \rightarrow \mathcal{L} \subset \mathbb{N}$ 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 $\partial A^n$
3. Set $n \rightarrow n + 1$ and repeat algorithm until $A^n = \emptyset$

Semi-supervised segmentation

Original data

User labels

Segmented data

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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

$p$-Laplacian diffusion for pre-processing

![Original](image1)

![$p = 1$](image2)

![$p = 2$](image3)

![$p = \infty$](image4)
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.


Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Cell classification in histology

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

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.


Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
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

Daniel Tenbrinck: "Partial Difference Operators on Weighted Graphs for Point Cloud Processing"
Thank you for your attention!

Any questions?