Graph Signal Processing and Applications

Olivier Lézoray

Normandie Université, Université de Caen Normandie, GREYC UMR CNRS 6072

olivier.lezoray@unicaen.fr

https://lezoray.users.greyc.fr

1 Introduction

2 Graphs and difference operators

3 Construction of graphs - non locality

4 *p*-Laplacian nonlocal regularization on graphs

5 Multiscale hierarchical decomposition of graph signals

6 Adaptation of active contours on graphs

1 Introduction

2 Graphs and difference operators

3 Construction of graphs - non locality

4 p-Laplacian nonlocal regularization on graphs

5 Multiscale hierarchical decomposition of graph signals

6 Adaptation of active contours on graphs

The data deluge - Graphs everywhere

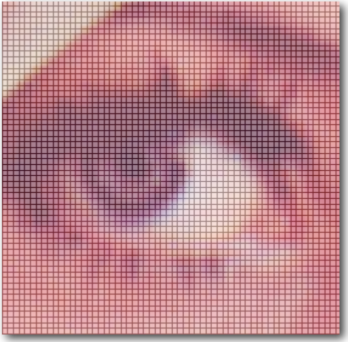
With the data deluge, graphs are everywhere: we are witnessing the rise of graphs in Big Data.

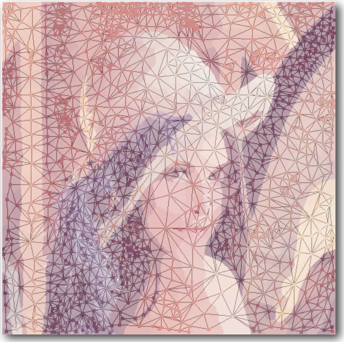
Graphs occur as a the most natural of representing arbitrary data by modeling the neighborhood properties between these data.

The data deluge - Graphs everywhere

With the data deluge, graphs are everywhere: we are witnessing the rise of graphs in Big Data.

Graphs occur as a the most natural of representing arbitrary data by modeling the neighborhood properties between these data.



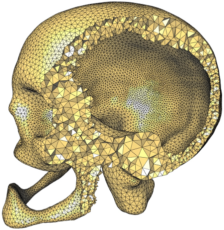


Images (grid graphs), Image partitions (superpixels graphs)

The data deluge - Graphs everywhere

With the data deluge, graphs are everywhere: we are witnessing the rise of graphs in Big Data.

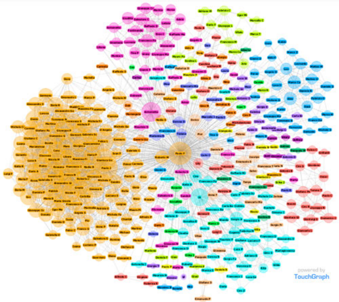
Graphs occur as a the most natural of representing arbitrary data by modeling the neighborhood properties between these data.

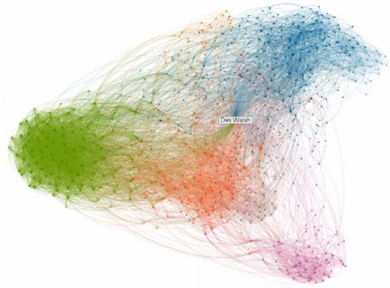


Meshes, 3D colored point clouds

The data deluge - Graphs everywhere

With the data deluge, graphs are everywhere: we are witnessing the rise of graphs in Big Data.
Graphs occur as a the most natural of representing arbitrary data by modeling the neighborhood properties between these data.



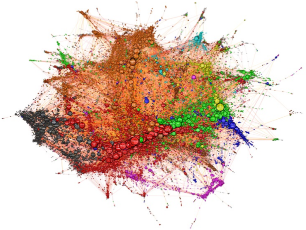
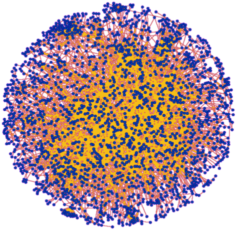
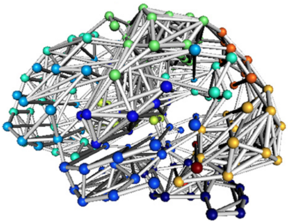


Social Networks: Facebook, LinkedIn

The data deluge - Graphs everywhere

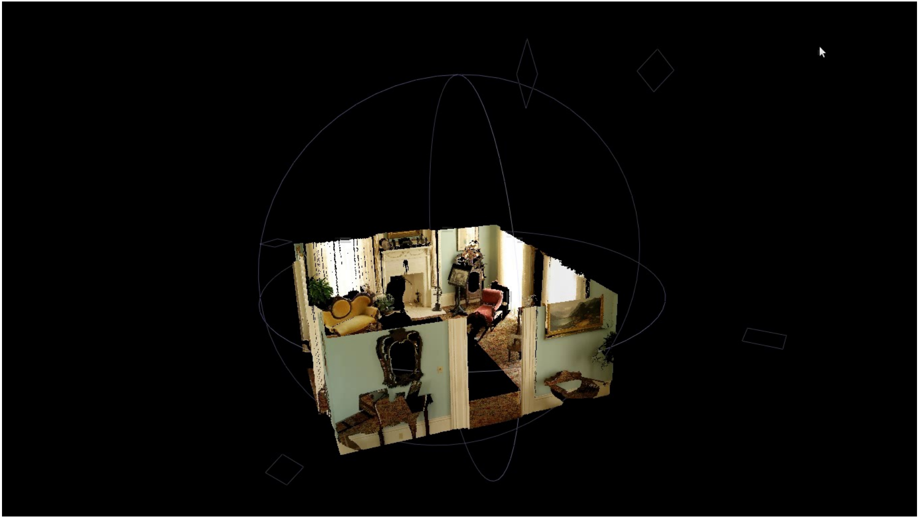
With the data deluge, graphs are everywhere: we are witnessing the rise of graphs in Big Data.

Graphs occur as a the most natural of representing arbitrary data by modeling the neighborhood properties between these data.

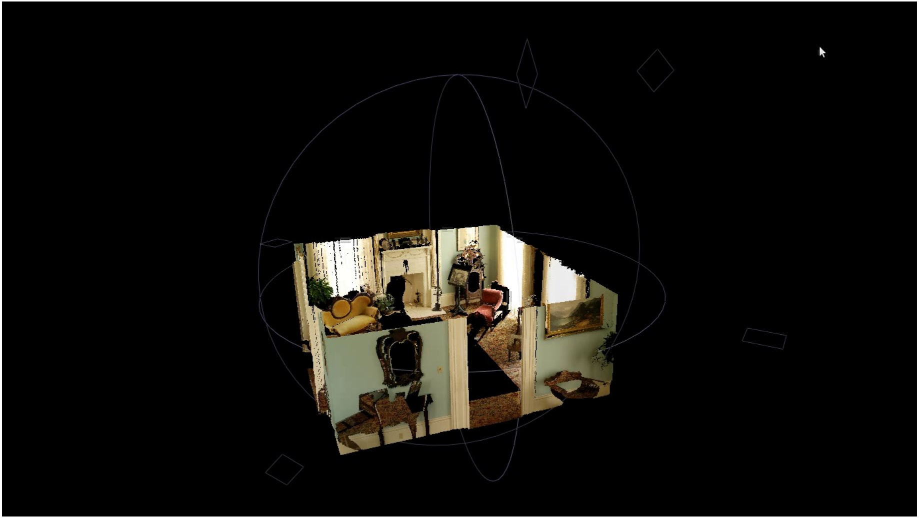




Internet, Biological Networks, Brain Graphs

3D Point Clouds ?



3D Point Clouds ?



Processing signals on specific Graphs

Usual ways to perform operations on graphs

- Graph theory, spectral analysis **(for data processing: similarity graphs)**
- Continuous variational methods **(for image/signal processing: grid graphs)**

Actual trends

- Emergence of a new research field called **Graph Signal Processing**
- Aim: development of algorithms that enable to process data that reside on the vertices or edges of a graph: **graph signals**
- Problem: how to process general (non Euclidean) graphs with image/signal processing techniques ?
- There are a lot of recent works that aim at extending image and signal processing tools for the processing of graph signals

Graph Signal processing

David I. Shuman, Sunil K. Narang, Pascal Frossard, Antonio Ortega, Pierre Vandergheynst, **The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains**. *IEEE Signal Process. Mag.* 30(3): 83-98 (2013)

Processing graph signals - some examples

- Signal processing side: graph wavelets
  - Diffusion wavelets (Coifman & Maggioni)
  - Spectral graph wavelets (Hammond, Vandergheynst & Gribonval)
  - Lifting Transforms on graphs (Narang & Ortega, Jansen & al.)
  - Multiscale Wavelets on Trees, Graphs (Gavish, Nadler & Coifman)
- Image processing side: graph PDEs
  - Mumford-Shah on graphs (Grady & Alvino)
  - Ginzburg-Landau graph functionals (Van Gennip & Bertozzi)
  - Nonlinear elliptic PDEs on graphs (Manfredi, Oberman)
  - Partial difference Equations (our works)

Partial difference Equations on graphs

Our line of research

- Our goal is to provide methods that *adapt* on graphs well-known PDE variational formulations under a functional analysis point of view.
- To do this we use Partial difference Equations (PdE) that mimic PDEs in domains having a graph structure.

Motivations

- Problems involving PDEs can be reduced to ones of a very much simpler structure by replacing the differentials by difference equations on graphs.
  R. Courant, K. Friedrichs, H. Lewy, On the partial difference equations of mathematical physics, Math. Ann. 100 (1928) 32-74.
- Instead of discretizing, we want equivalents on graphs of differential operators
- The analogue of PDEs on graphs is obtained by simply replacing the continuous operators by their discrete equivalent
- PdEs mimic PDEs in domains having a graph structure.

Interest of our proposals:

- To dispose of discrete analogues of differential geometry operators (integral, derivation, gradient, divergence, p-Laplacian, etc.)
- To use the framework of PdEs to transcribe PDEs on graphs,
- Provides a natural extension of variational methods on graphs,
- Can be used with arbitrary graphs,
- Provides a unification of local and nonlocal processing on images,
- Using weighted graphs provides Adaptive PDEs according to data geometry,
- Recovers exactly the discretization of PDEs on Euclidean domains.

PdEs on graphs - Adaptation examples

- *p*-Laplacian (isotropic, anisotropic) regularization on graphs,

A. Elmoataz, O. Lezoray, S. Bougleux, *Nonlocal Discrete Regularization on Weighted Graphs: a framework for Image and Manifold Processing*, IEEE transactions on Image Processing, Vol. 17(7), pp. 1047-1060, 2008.

- Mathematical morphology on graphs,

V.-T. Ta, A. Elmoataz and O. Lezoray, *Nonlocal PDEs-based Morphology on Weighted Graphs for Image and Data Processing*. IEEE transactions on Image Processing, 20(6) : pp. 1504-1516. 2011.

- Front Propagation on graphs,

X. Desquesnes, A. Elmoataz, O. Lezoray, *Eikonal equation adaptation on weighted graphs: fast geometric diffusion process for local and non-local image and data processing*, Journal of Mathematical Imaging and Vision, Vol. 46(2), pp. 238-257, 2013.

- Hierarchical decomposition of graph signals,

M. Hidane, O. Lezoray, A. Elmoataz, *Nonlinear Multilayered Representation of Graph-Signals*, Journal of Mathematical Imaging and Vision, Vol. 45(2), pp. 114-137, 2013.

- Active contours on graph signals,

O. Lezoray, A. Elmoataz, V.-T. Ta, *Nonlocal PdEs on graphs for active contours models with applications to image segmentation and data clustering*, International Conference on Acoustics, Speech, and Signal Processing (IEEE), pp. 873-876, 2012.

PdEs on graphs - Adaptation examples

- ***p*-Laplacian (isotropic, anisotropic) regularization on graphs**,

A. Elmoataz, O. Lezoray, S. Bougleux, *Nonlocal Discrete Regularization on Weighted Graphs: a framework for Image and Manifold Processing*, IEEE transactions on Image Processing, Vol. 17(7), pp. 1047-1060, 2008.

- Mathematical morphology on graphs,

V.-T. Ta, A. Elmoataz and O. Lezoray, *Nonlocal PDEs-based Morphology on Weighted Graphs for Image and Data Processing*. IEEE transactions on Image Processing, 20(6) : pp. 1504-1516. 2011.

- Front Propagation on graphs,

X. Desquesnes, A. Elmoataz, O. Lezoray, *Eikonal equation adaptation on weighted graphs: fast geometric diffusion process for local and non-local image and data processing*, Journal of Mathematical Imaging and Vision, Vol. 46(2), pp. 238-257, 2013.

- **Hierarchical decomposition of graph signals**,

M. Hidane, O. Lezoray, A. Elmoataz, *Nonlinear Multilayered Representation of Graph-Signals*, Journal of Mathematical Imaging and Vision, Vol. 45(2), pp. 114-137, 2013.

- **Active contours on graph signals**,

O. Lezoray, A. Elmoataz, V.-T. Ta, *Nonlocal PdEs on graphs for active contours models with applications to image segmentation and data clustering*, International Conference on Acoustics, Speech, and Signal Processing (IEEE), pp. 873-876, 2012.

1 Introduction

2 Graphs and difference operators

3 Construction of graphs - non locality

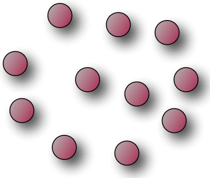
4 *p*-Laplacian nonlocal regularization on graphs

5 Multiscale hierarchical decomposition of graph signals

6 Adaptation of active contours on graphs

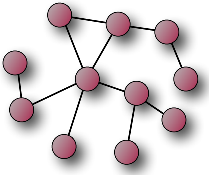
Weighted graphs Basics

- A weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$ consists in a finite set $\mathcal{V} = \{v_1, \dots, v_N\}$ of N vertices



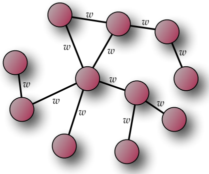
Weighted graphs Basics

- A weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$ consists in a finite set $\mathcal{V} = \{v_1, \ldots, v_N\}$ of N vertices
- and a finite set $\mathcal{E} = \{e_1, \ldots, e_{N'}\} \subset \mathcal{V} \times \mathcal{V}$ of N' weighted edges.



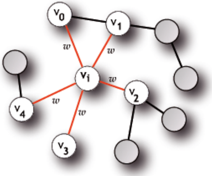
Weighted graphs Basics

- A weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$ consists in a finite set $\mathcal{V} = \{v_1, \ldots, v_N\}$ of N vertices
- and a finite set $\mathcal{E} = \{e_1, \ldots, e_{N'}\} \subset \mathcal{V} \times \mathcal{V}$ of N' weighted edges.
- $e_{ij} = (v_i, v_j)$ is the edge of $\mathcal{E}$ that connects vertices v_i and v_j of $\mathcal{V}$. Its weight, denoted by $w_{ij} = w(v_i, v_j)$, represents the similarity between its vertices.
- Similarities are usually computed by using a positive symmetric function $w : \mathcal{V} \times \mathcal{V} \to \mathbb{R}^+$ satisfying $w(v_i, v_j) = 0$ if $(v_i, v_j) \notin \mathcal{E}$.



Weighted graphs Basics

- A weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$ consists in a finite set $\mathcal{V} = \{v_1, \ldots, v_N\}$ of N vertices
- and a finite set $\mathcal{E} = \{e_1, \ldots, e_{N'}\} \subset \mathcal{V} \times \mathcal{V}$ of N' weighted edges.
- $e_{ij} = (v_i, v_j)$ is the edge of $\mathcal{E}$ that connects vertices v_i and v_j of $\mathcal{V}$. Its weight, denoted by $w_{ij} = w(v_i, v_j)$, represents the similarity between its vertices.
- Similarities are usually computed by using a positive symmetric function $w : \mathcal{V} \times \mathcal{V} \to \mathbb{R}^+$ satisfying $w(v_i, v_j) = 0$ if $(v_i, v_j) \notin \mathcal{E}$.
- The notation $v_i \sim v_j$ is used to denote two adjacent vertices.



Space of functions on Graphs

- $\mathcal{H}(\mathcal{V})$ and $\mathcal{H}(\mathcal{E})$ are the Hilbert spaces of graph signals: real-valued functions defined on the vertices or the edges of a graph $\mathcal{G}$.
- A function $f : \mathcal{V} \to \mathbb{R}$ of $\mathcal{H}(\mathcal{V})$ assigns a real value $x_i = f(v_i)$ to $v_i \in \mathcal{V}$.
- By analogy with functional analysis on continuous spaces, the integral of a function $f \in \mathcal{H}(\mathcal{V})$, over the set of vertices $\mathcal{V}$, is defined as

$$\int_{\mathcal{V}} f = \sum_{\mathcal{V}} f$$

- Both spaces $\mathcal{H}(\mathcal{V})$ and $\mathcal{H}(\mathcal{E})$ are endowed with the usual inner products:

$$\langle f, h \rangle_{\mathcal{H}(\mathcal{V})} = \sum_{v_i \in \mathcal{V}} f(v_i)h(v_i), \text{ where } f, h : \mathcal{V} \to \mathbb{R}$$

$$\langle F, H \rangle_{\mathcal{H}(\mathcal{E})} = \sum_{v_i \in \mathcal{V}} \sum_{v_j \sim v_i} F(v_i, v_j)H(v_i, v_j) \text{ where } F, H : \mathcal{E} \to \mathbb{R}$$

Difference operators on weighted graphs

➲ **Discrete analogue on graphs of classical continuous differential geometry**.

The **difference operator** of f, $d_w : \mathcal{H}(\mathcal{V}) \to \mathcal{H}(\mathcal{E})$, is defined on an edge $e_{ij} = (v_i, v_j) \in \mathcal{E}$ by:

$$(d_w f)(e_{ij}) = (d_w f)(v_i, v_j) = w(v_i, v_j)^{1/2}(f(v_j) - f(v_i)) \ . \qquad (1)$$

The **adjoint** of the difference operator, $d_w^* : \mathcal{H}(\mathcal{E}) \to \mathcal{H}(\mathcal{V})$, is a linear operator defined by

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

and expressed by

$$(d_w^* H)(v_i) = -div_w(H)(v_i) = \sum_{v_j \sim v_i} w(v_i, v_j)^{1/2}(H(v_j, v_i) - H(v_i, v_j)) \ . \qquad (2)$$

M. Hein, J.-Y. Audibert, U. Von Luxburg, *From Graphs to Manifolds - Weak and Strong Pointwise Consistency of Graph Laplacians*. COLT 2005: 470-485

D. Zhou, J. Huang, B. Schlkopf, *Learning from labeled and unlabeled data on a directed graph*. ICML 2005: 1036-1043

Weighted gradient operator

The **weighted gradient operator** of a function $f \in \mathcal{H}(\mathcal{V})$, at a vertex $v_i \in \mathcal{V}$, is the vector operator defined by

$$(\nabla_{\mathbf{w}} \mathbf{f})(\mathbf{v_i}) = [(d_w f)(v_i, v_j) \, : \, v_j \in \mathcal{V}]^T. \quad (3)$$

➲ **The gradient considers all vertices $v_j \in \mathcal{V}$ and not only $v_j \sim v_i$.**

The $\mathcal{L}_p$ norm of this vector represents the *local variation* of the function f at a vertex of the graph (It is a semi-norm for $p \geq 1$):

$$\|(\nabla_{\mathbf{w}} \mathbf{f})(\mathbf{v_i})\|_p = \Big[\sum_{v_j \sim v_i} w_{ij}^{p/2} \big| f(v_j) - f(v_i) \big|^p \Big]^{1/p} . \quad (4)$$

A. Elmoataz, O. Lezoray, S. Bougleux, *Nonlocal Discrete Regularization on Weighted Graphs: a framework for Image and Manifold Processing*, IEEE transactions on Image Processing, Vol. 17(7), pp. 1047-1060, 2008.

Isotropic p-Laplacian

The *weighted p-Laplace isotropic operator* of a function $f \in \mathcal{H}(\mathcal{V})$, noted $\Delta^i_{w,p} : \mathcal{H}(\mathcal{V}) \to \mathcal{H}(\mathcal{V})$, is defined by:

$$(\Delta^i_{w,p} f)(v_i) = \frac{1}{2} d^*_w(\|(\nabla_\mathbf{w}\mathbf{f})(\mathbf{v_i})\|_2^{p-2}(d_w f)(v_i, v_j)) \ . \quad (5)$$

The isotropic p-Laplace operator of $f \in \mathcal{H}(\mathcal{V})$, at a vertex $v_i \in \mathcal{V}$, can be computed by:

$$(\Delta^i_{w,p} f)(v_i) = \frac{1}{2} \sum_{v_j \sim v_i} (\gamma^i_{w,p} f)(v_i, v_j)(f(v_i) - f(v_j)) \ , \quad (6)$$

with

$$(\gamma^i_{w,p} f)(v_i, v_j) = w_{ij} \left(\|(\nabla_\mathbf{w}\mathbf{f})(\mathbf{v_j})\|_2^{p-2} + \|(\nabla_\mathbf{w}\mathbf{f})(\mathbf{v_i})\|_2^{p-2} \right) \ . \quad (7)$$

The p-Laplace isotropic operator is nonlinear, except for $p = 2$ (corresponds to the combinatorial Laplacian). For $p = 1$, it corresponds to the *weighted curvature* of the function f on the graph.

A. Elmoataz, O. Lezoray, S. Bougleux, *Nonlocal Discrete Regularization on Weighted Graphs: a framework for Image and Manifold Processing*, IEEE transactions on Image Processing, Vol. 17(7), pp. 1047-1060, 2008.

Anisotropic p-Laplacian

The *weighted p-Laplace anisotropic operator* of a function $f \in \mathcal{H}(\mathcal{V})$, noted $\Delta^a_{w,p} : \mathcal{H}(\mathcal{V}) \to \mathcal{H}(\mathcal{V})$, is defined by:

$$(\Delta^a_{w,p} f)(v_i) = \frac{1}{2} d^*_w(|(d_w f)(v_i, v_j)|^{p-2}(d_w f)(v_i, v_j)) \ . \tag{8}$$

The anisotropic p-Laplace operator of $f \in \mathcal{H}(\mathcal{V})$, at a vertex $v_i \in \mathcal{V}$, can be computed by:

$$(\Delta^a_{w,p} f)(v_i) = \sum_{v_j \sim v_i} (\gamma^a_{w,p} f)(v_i, v_j)(f(v_i) - f(v_j)) \ . \tag{9}$$

with

$$(\gamma^a_{w,p} f)(v_i, v_j) = w_{ij}^{p/2} |f(v_i) - f(v_j)|^{p-2} \ . \tag{10}$$

O. Lezoray, V.T. Ta, A. Elmoataz, *Partial differences as tools for filtering data on graphs*, Pattern Recognition Letters, Vol. 31(14), pp. 2201-2213, 2010.

1 Introduction

2 Graphs and difference operators

3 Construction of graphs - non locality

4 *p*-Laplacian nonlocal regularization on graphs

5 Multiscale hierarchical decomposition of graph signals

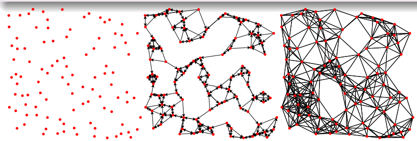
6 Adaptation of active contours on graphs

Constructing graphs

Any discrete domain can be modeled by a weighted graph where each data point is represented by a vertex $v_i \in \mathcal{V}$.

Unorganized data

An unorganized set of points $\mathcal{V} \subset \mathbb{R}^n$ can be seen as a function $f^0 : \mathcal{V} \to \mathbb{R}^m$.
The set of edges is defined by in modeling the neighborhood of each vertex based on similarity relationships between feature vectors.
Typical graphs: k-nearest neighbors graphs and ϵ-neighborhood graphs.



Organized data

Typical cases of organized data are signals, gray-scale or color images (in 2D or 3D).
The set of edges is defined by spatial relationships.
Such data can be seen as functions $f^0 : \mathcal{V} \subset \mathbb{Z}^n \to \mathbb{R}^m$.
Typical graphs: pixel or region graphs.

Weighting graphs

For an initial function $f^0 : \mathcal{V} \to \mathbb{R}^m$, similarity relationship between data can be incorporated within edges weights according to a measure of similarity $g : \mathcal{E} \to [0,1]$ with $w(e_{ij}) = g(e_{ij})$, $\forall e_{ij} \in \mathcal{E}$.
Each vertex v_i is associated with a feature vector $\mathbf{F}_{\boldsymbol{\tau}}^{\mathbf{f^0}} : \mathcal{V} \to \mathbb{R}^{m \times q}$ where q corresponds to this vector size:

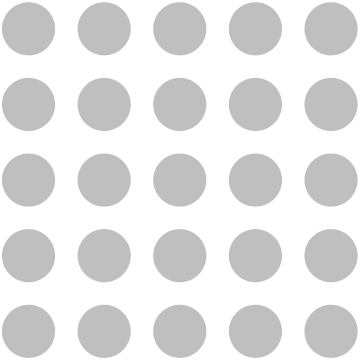
$$\mathbf{F}_{\boldsymbol{\tau}}^{\mathbf{f^0}}(v_i) = \Big(f^0(v_j) : v_j \in \mathcal{N}_\tau(v_i) \cup \{v_i\}\Big)^T \tag{11}$$

with $\mathcal{N}_\tau(v_i) = \{v_j \in \mathcal{V} \setminus \{v_i\} : \mu(v_i, v_j) \le \tau\}$.
For an edge e_{ij} and a distance measure $\rho : \mathbb{R}^{m\times q} \times \mathbb{R}^{m\times q} \to \mathbb{R}$ associated to $\mathbf{F}_{\boldsymbol{\tau}}^{\mathbf{f^0}}$, we can have:

$$\begin{aligned}
g_1(e_{ij}) &= 1 \text{ (unweighted case) },\\
g_2(e_{ij}) &= \exp\Big(-\rho\big(\mathbf{F}_{\boldsymbol{\tau}}^{\mathbf{f^0}}(v_i), \mathbf{F}_{\boldsymbol{\tau}}^{\mathbf{f^0}}(v_j)\big)^2/\sigma^2\Big) \text{ with } \sigma > 0 \ , \\
g_3(e_{ij}) &= 1/\Big(1 + \rho\big(\mathbf{F}_{\boldsymbol{\tau}}^{\mathbf{f^0}}(v_i), \mathbf{F}_{\boldsymbol{\tau}}^{\mathbf{f^0}}(v_j)\big)\Big)
\end{aligned} \tag{12}$$

Graph topology

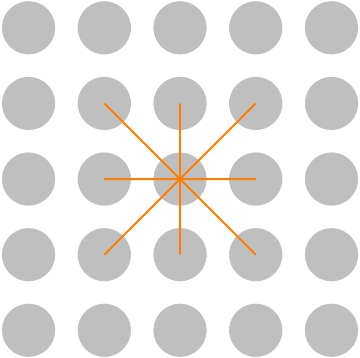
Digital Image



Graph topology

Digital Image

8-neighborhood : 3×3

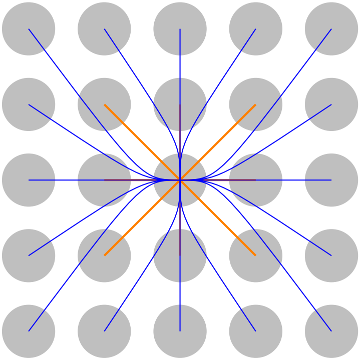


Graph topology

Digital Image

8-neighborhood : 3×3

24-neighborhood : 5×5



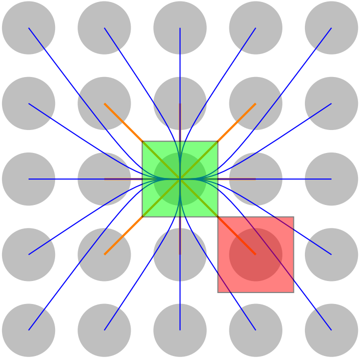
Graph topology

Digital Image

8-neighborhood : 3×3

24-neighborhood : 5×5

Local: a value is associated to vertices



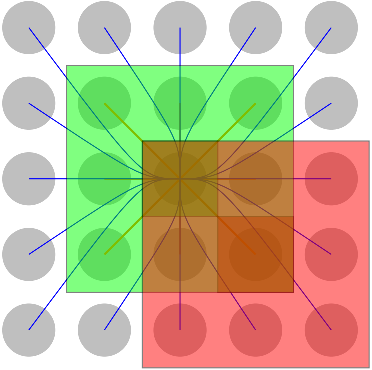
Graph topology

Digital Image

8-neighborhood : 3×3

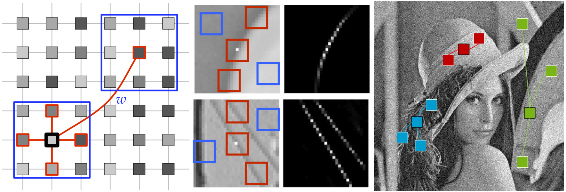
24-neighborhood : 5×5

Nonlocal: a patch (vector of values in a given neighborhood) is associated to vertices.



With Graphs

- Nonlocal behavior is directly expressed by the graph topology.
- Patches are used to measure similarity between vertices.



Consequences

- Nonlocal processing of images becomes local processing on similarity graphs.
- Our difference operators on graphs naturally enable local and nonlocal configurations (with the weight function and the graph topology)
- with specific graph topologies and weights, the discretization of continuous formulations can be recovered

1 Introduction

2 Graphs and difference operators

3 Construction of graphs - non locality

4 *p*-Laplacian nonlocal regularization on graphs

5 Multiscale hierarchical decomposition of graph signals

6 Adaptation of active contours on graphs

p-Laplacian nonlocal regularization on graphs

Let $f^0 : \mathcal{V} \to \mathbb{R}$ be the noisy version of a clean graph signal $g : \mathcal{V} \to \mathbb{R}$ defined on the vertices of a weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, w)$.
To recover g, seek for a function $f : \mathcal{V} \to \mathbb{R}$ regular enough on $\mathcal{G}$, and close enough to f^0, with the following variational problem:

$$g \approx \min_{f:\mathcal{V}\to\mathbb{R}} \left\{ E^*_{w,p}(f, f^0, \lambda) = R^*_{w,p}(f) + \frac{\lambda}{2}\|f - f^0\|_2^2 \right\}, \qquad (13)$$

where the regularization functional $R^*_{w,p} : \mathcal{H}(\mathcal{V}) \to \mathbb{R}$ can correspond to an isotropic $R^i_{w,p}$ or an anisotropic $R^a_{w,p}$ functionnal.

A. Elmoataz, O. Lezoray, S. Bougleux, *Nonlocal Discrete Regularization on Weighted Graphs: a framework for Image and Manifold Processing*, IEEE transactions on Image Processing, Vol. 17(7), pp. 1047-1060, 2008.
A. Elmoataz, O. Lezoray, V.-T. Ta, S. Bougleux, *Partial difference equations on graphs for local and nonlocal image processing*, In Image Processing and Analysing With Graphs: Theory and Practice, Editors: O. Lezoray and L. Grady, Publisher: CRC Press / Taylor and Francis, Series: Digital Imaging and Computer Vision, pp. 175-206, 2012.

Isotropic and anisotropic regularization terms

The isotropic regularization functionnal $R^i_{w,p}$ is defined by the $\mathcal{L}_2$ norm of the gradient and is the discrete p-Dirichlet form of the function $f \in \mathcal{H}(\mathcal{V})$:

$$\begin{aligned} R^i_{w,p}(f) &= \frac{1}{p} \sum_{v_i \in \mathcal{V}} \|(\nabla_\mathbf{w} \mathbf{f})(\mathbf{v_i})\|_2^p = \frac{1}{p} \langle f, \Delta^i_{w,p} f \rangle_{\mathcal{H}(\mathcal{V})} \\ &= \frac{1}{p} \sum_{v_i \in \mathcal{V}} \left[\sum_{v_j \sim v_i} w_{ij} (f(v_j) - f(v_i))^2 \right]^{\frac{p}{2}} . \end{aligned} \tag{14}$$

The anisotropic regularization functionnal $R^a_{w,p}$ is defined by the $\mathcal{L}_p$ norm of the gradient:

$$\begin{aligned} R^a_{w,p}(f) &= \frac{1}{p} \sum_{v_i \in \mathcal{V}} \|(\nabla_\mathbf{w} \mathbf{f})(\mathbf{v_i})\|_p^p = \frac{1}{p} \langle f, \Delta^a_{w,p} f \rangle_{\mathcal{H}(\mathcal{V})} \\ &= \frac{1}{p} \sum_{v_i \in \mathcal{V}} \sum_{v_j \sim v_i} w_{ij}^{p/2} |f(v_j) - f(v_i)|^p . \end{aligned} \tag{15}$$

When $p \geq 1$, the energy $E^*_{w,p}$ is a convex functional of functions of $\mathcal{H}(\mathcal{V})$.

Isotropic/Anisotropic diffusion processes

To get the solution of the minimizer, we consider the following system of equations:

$$\frac{\partial E^*_{w,p}(f, f^0, \lambda)}{\partial f(v_i)} = 0, \forall v_i \in \mathcal{V} \tag{16}$$

which is rewritten as:

$$\frac{\partial R^*_{w,p}(f)}{\partial f(v_i)} + \lambda(f(v_i) - f^0(v_i)) = 0, \quad \forall v_i \in \mathcal{V}. \tag{17}$$

Moreover, we can prove that

$$\frac{\partial R^i_{w,p}(f)}{\partial f(v_i)} = 2(\Delta^i_{w,p} f)(v_i) \text{ and } \frac{\partial R^a_{w,p}(f)}{\partial f(v_i)} = (\Delta^a_{w,p} f)(v_i) \ . \tag{18}$$

The system of equations is then rewritten as which is equivalent to the following system of equations:

$$\left(\lambda + \sum_{v_j \sim v_i} (\gamma^*_{w,p} f)(v_i, v_j)\right) f(v_i) - \sum_{v_j \sim v_i} (\gamma^*_{w,p} f)(v_i, v_j) f(v_j) = \lambda f^0(v_i). \tag{19}$$

Isotropic/Anisotropic diffusion processes

We can use the linearized Gauss-Jacobi iterative method to solve the previous systems. Let n be an iteration step, and let $f^{(n)}$ be the solution at the step n. Then, the method is given by the following algorithm:

$$\begin{cases} f^{(0)} = f^0 \\ f^{(n+1)}(v_i) = \dfrac{\lambda f^0(v_i) + \sum_{v_j \sim v_i} (\gamma^*_{w,p} f^{(n)})(v_i, v_j) f^{(n)}(v_j)}{\lambda + \sum_{v_j \sim v_i} (\gamma^*_{w,p} f^{(n)})(v_i, v_j)}, \forall v_i \in \mathcal{V}. \end{cases} \quad (20)$$

$$\text{with } (\gamma^i_{w,p} f)(v_i, v_j) = w_{ij} \left(\|(\nabla_\mathbf{w} \mathbf{f})(\mathbf{v_j})\|_2^{p-2} + \|(\nabla_\mathbf{w} \mathbf{f})(\mathbf{v_i})\|_2^{p-2} \right) , \quad (21)$$

$$\text{and } (\gamma^a_{w,p} f)(v_i, v_j) = w_{ij}^{p/2} |f(v_i) - f(v_j)|^{p-2} . \quad (22)$$

It describes a family of discrete diffusion processes, which is parameterized by the structure of the graph (topology and weight function), the parameter p, and the parameter λ.

λ	w	Graph	$p = 1$	$p = 2$	$p \in]0, 1[$
0	*exp*()	semi-local	Ours	Bilateral	Our
0	*exp*()	nonlocal	Ours	NLMeans	Our
$\neq 0$	constant	local	TV Digital	L_2 Digital	Ours
$\neq 0$	any	nonlocal	Ours	Ours	Ours

More efficient minimization

Previous scheme is very slow and introduces a smoothing parameter when $p = 1$.
Better to use primal-dual algorithms: the Chambolle and Pock that exhibits very good numerical performance.
To solve the general optimization problem $\min_{x \in \mathcal{H}(\mathcal{V})} F(Kx) + G(x)$, they have proposed the algorithm:

$$\begin{cases} x^0 = \bar{x}^0 = f, \quad y^0 = 0 \\ y^{n+1} = \text{prox}_{\sigma F^*}(y^n + \sigma K \bar{x}^n), \\ x^{n+1} = \text{prox}_{\tau G}(x^n - \tau K^* y^{n+1}), \\ \bar{x}^{n+1} = x^{n+1} + \theta(x^{n+1} - x^n), \end{cases} \tag{23}$$

where F^* is the conjugate of F, K^* is the adjoint operator of K, and prox the proximal operator.
To apply it to our case, we have to set e.g., for the isotropic case, $F = ||.||_2^1$, $K = \nabla_{\mathbf{w}}$, $K^* = -div_w$ and $G = \frac{\lambda}{2}||. - f||_2^2$.

M. Hidane, O. Lezoray, A. Elmoataz, *Nonlinear Multilayered Representation of Graph-Signals*, Journal of Mathematical Imaging and Vision, Vol. 45(2), pp. 114-137, 2013.

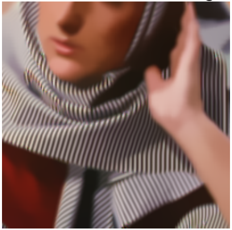
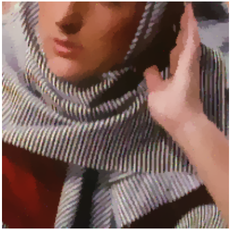
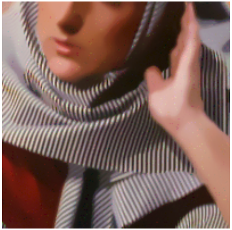
Examples: Image denoising

Original image

Noisy image (Gaussian noise with $\sigma = 15$)

$f^0 : \mathcal{V} \to \mathbb{R}^3$ PSNR=29.38dB

Examples: Image denoising

	Isotropic $\mathcal{G}_1$, $\mathbf{F}_0^{\mathbf{f}^0} = f^0$	Isotropic $\mathcal{G}_7$, $\mathbf{F}_3^{\mathbf{f}^0}$	Anisotropic $\mathcal{G}_7$, $\mathbf{F}_3^{\mathbf{f}^0}$
$p = 2$	 PSNR=28.52db	 PSNR=31.79dB	 PSNR=31.79dB
$p = 1$	 PSNR=31.25dB	 PSNR=34.74dB	 PSNR=31.81dB

Original Mesh
$f^0 : \mathcal{V} \rightarrow \mathbb{R}^3$

Isotropic, $p = 2$

Isotropic, $p = 1$,

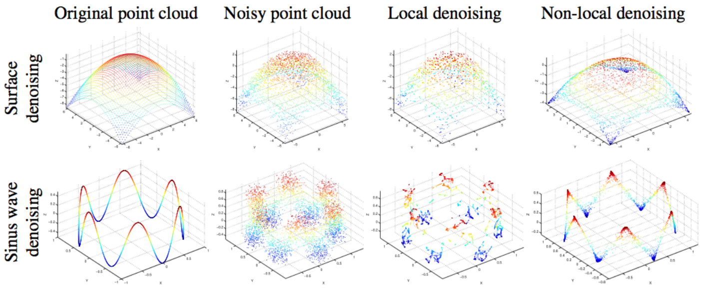
Anisotropic, $p = 1$

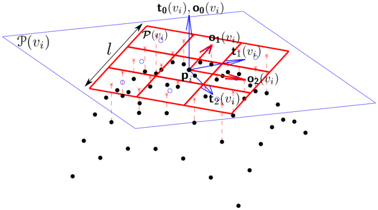
Examples: Colored Mesh simplification

Original Colored Mesh
$f^0 : \mathcal{V} \rightarrow \mathbb{R}^3$

$\lambda = 1$

$\lambda = 0.5$





2D Patches on 3D Point clouds

Examples: Colored Point Cloud denoising

Initial Point cloud
$f^0 : \mathcal{V} \to \mathbb{R}^3$
127039 points

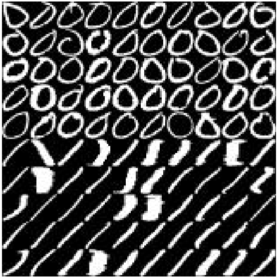
Noisy

Local Graph
4-NNG

Non Local Graph
200-NNG, $\mathbf{F}_9^{\mathbf{f}^0}$

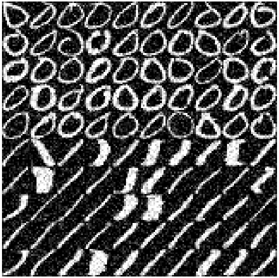
F. Lozes, A. Elmoataz, O. Lezoray, *Nonlocal processing of 3D colored point clouds*, International Conference on Pattern Recognition (ICPR), pp. 1968-1971, 2012.

Examples: Image Database denoising

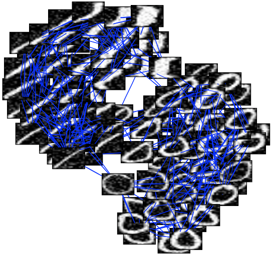


Initial data

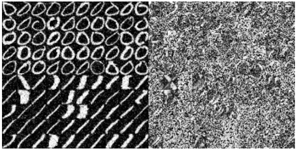
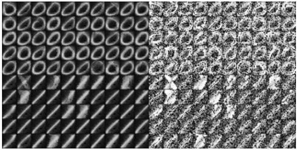
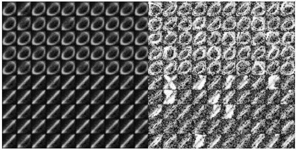
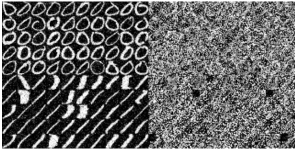
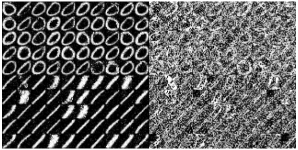
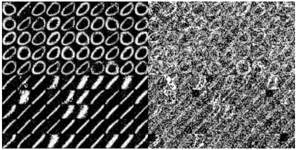
$f^0 : \mathcal{V} \to \mathbb{R}^{16\times16}$



Noisy data

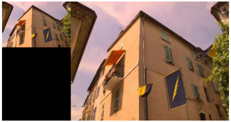


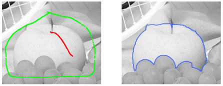
10-NNG

	$\lambda = 1$	$\lambda = 0.01$	$\lambda = 0$
		Isotropic	
$p = 1$			
	PSNR=18.80dB	PSNR=13.54dB	PSNR=10.52dB
		Anisotropic	
$p = 1$			
	PSNR=18.96dB	PSNR=15.19dB	PSNR=14.41dB

Interpolation of missing data on graphs

Let $f^0 : \mathcal{V}_0 \to \mathbb{R}$ be a function with $\mathcal{V}_0 \subset \mathcal{V}$ be the subset of vertices from the whole graph with known values.





The interpolation consists in recovering values of f for the vertices of $\mathcal{V} \setminus \mathcal{V}_0$ given values for vertices of $\mathcal{V}_0$ formulated by:

$$\min_{f:\mathcal{V}\to\mathbb{R}} R^*_{w,p}(f) + \lambda(v_i)\|f(v_i) - f^0(v_i)\|_2^2 \ . \tag{24}$$

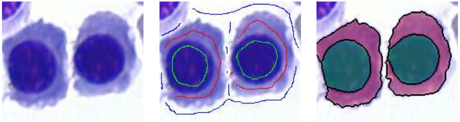
Since $f^0(v_i)$ is known only for vertices of $\mathcal{V}_0$, the Lagrange parameter is defined as $\lambda : \mathcal{V} \to \mathbb{R}$:

$$\lambda(v_i) = \begin{cases} \lambda & \text{if } v_i \in \mathcal{V}_0 \\ 0 & \text{otherwise.} \end{cases} \tag{25}$$

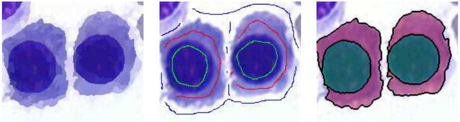
This comes to consider $\Delta^*_{w,p} f(v_i) = 0$ on $\mathcal{V} \setminus \mathcal{V}_0$.
Our isotropic and anisotropic diffusion processes can be directly used to perform the interpolation.

Examples: Image segmentation

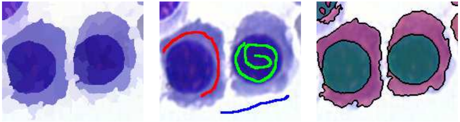
Solve $\Delta^*_{w,p} f(v_i) = 0$ on $\mathcal{V} \setminus \mathcal{V}_0$.



(a) 27 512 pixels (b) Original+Labels (c) $t = 50$ (11 seconds)



(d) 639 zones (98% of reduction) (e) Original+Labels (f) $t = 5$ (< 1 second)



(g) 639 zones (98% of reduction) (h) Original+Labels (i) $t = 2$ (< 1 second)

Examples: Image colorization

Gray level image

Color scribbles

Compute Weights from the gray-level image, interpolation is performed in a chrominance color space from the seeds: $\mathbf{f^c(v_i)} = \left[\frac{f_1^s(v_i)}{f^l(v_i)}, \frac{f_2^s(v_i)}{f^l(v_i)}, \frac{f_3^s(v_i)}{f^l(v_i)} \right]^T$

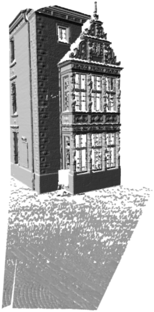
O. Lezoray, A. Elmoataz, V.T. Ta, *Nonlocal graph regularization for image colorization*, International Conference on Pattern Recognition (ICPR), 2008.

Examples: Image colorization

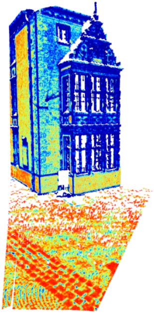
$p = 1,\ \mathcal{G}_1,\ \mathbf{F}_{\mathbf{0}}^{\mathbf{f}^0} = f^0$

$p = 1,\ \mathcal{G}_5,\ \mathbf{F}_{\mathbf{2}}^{\mathbf{f}^0}$

Examples: 3D Point Cloud colorization



3D coordinates



Saliency from height patches

Similarity from saliency patches

Saliency of a vertex is defined as its degree: provides an equivalent of grayscale values for image colorization.

Examples: 3D Point Cloud colorization

$$p = 1,\ \mathcal{G}_{25},\ \mathbf{F}_{9}^{\mathbf{f}^0}$$

F. Lozes, A. Elmoataz, O. Lézoray, *PDE-based Graph Signal Processing for 3D Color Point Clouds: Opportunities for Cultural Heritage*, IEEE Signal Processing Magazine, Vol. 32, n4, pp. 103-111, 2015.

1 Introduction

2 Graphs and difference operators

3 Construction of graphs - non locality

4 p-Laplacian nonlocal regularization on graphs

5 Multiscale hierarchical decomposition of graph signals

6 Adaptation of active contours on graphs

Motivations

- For images, structure-texture decomposition highly depends on the analysis scale;
- It is more natural to consider several levels of decompositions
- We consider the TNV approach that proposes to decompose images into several layers with an iterative variational approach;

E. Tadmor, S. Nezzar, L. Vese, *A multiscale image representation using hierarchical (BV,L2) decompositions*, Multiscale Modeling & Simulation 2 (4) (2004) 554-579.

The TNV approach: principles

Let

- f be a scalar image;
- E an energy of the form $E(u; f, \lambda) = \lambda R(u) + D(u, f)$; where R is a regularity term and D a fidelity term.
- $\lambda_0 > \lambda_1 > \ldots > \lambda_n > 0$ a sequence of scale parameters;

The application of the following algorithm

$$\begin{cases} v_{-1} & = f, \\ u_i & = \underset{u}{\operatorname{argmin}} \; E(u; v_{i-1}; \lambda_i), \; 0 \leq i \leq n, \\ v_i & = v_{i-1} - u_i, \; 0 \leq i \leq n. \end{cases}$$

enables to obtain the following multi-scale decomposition

$$\boxed{\mathbf{f} = \sum_{i=0}^{n} \mathbf{u}_i + \mathbf{v}_n.}$$

The TNV approach: remarks and extension

Remarks

- The algorithm decomposes the successive residues at finer and finer scales;
- The sequence $(\lambda_i)_{i \geq 0}$ has to be decreasing: the first extracted layers do represent a coarse representation of the initial signal f;
- In TNV, the authors have chosen a sequence of dyadic scales: $\lambda_i = \lambda_{i-1}/2$;
- In TNV, the considered functional is TV-L_2;
- In TNV, convergence guarantees are provided;

Extension to weighted graphs

- We propose to adapt the TNV approach for graph signals with isotropic p-Laplacian regularization;
- For image processing, this enables to integrate a nonlocal behavior into the decomposition;
- The convergence can be studied as well as the parameters;
- Innovative applications in detail enhancement for graph signals can be obtained;

TNV on graphs

$$\lambda_0 > \lambda_1 > \ldots > \lambda_n > 0$$

$$\begin{cases} \mathbf{v}_{-1} & = \mathbf{f}, \\ \mathbf{u}_i & = \underset{\mathbf{u}\in\mathcal{H}(\mathcal{V})}{\operatorname{argmin}} \; E^i_{w,p}(\mathbf{u}; \mathbf{v}_{i-1}; \lambda_i), \; 0 \leq i \leq n, \\ \mathbf{v}_i & = \mathbf{v}_{i-1} - \mathbf{u}_i, \; 0 \leq i \leq n. \end{cases}$$

$$\boxed{\mathbf{f} = \sum_{i=0}^{n} \mathbf{u}_i + \mathbf{v}_n.}$$

- Convergence ?
- Scale parameters ?

M. Hidane, O. Lezoray, A. Elmoataz, *Nonlinear Multilayered Representation of Graph-Signals*, Journal of Mathematical Imaging and Vision, Vol. 45(2), pp. 114-137, 2013.

Characterization of the minimizer of $E^i_{w,1}$

First we introduce the analog on graphs of the Meyer G space to represent oscillating patterns:

$$G_w = \{\mathbf{u} \in \mathcal{H}(\mathcal{V}) : \exists \mathbf{f} \in \mathcal{H}(\mathcal{E}), \mathbf{u} = div_w \mathbf{f}\}$$

with the following norm, $\forall \mathbf{u} \in G_w$:

$$\|\mathbf{u}\|_{G_w} = \inf\{\|\mathbf{F}\|_\infty, \mathbf{F} \in \mathcal{H}(\mathcal{E}), div_w(\mathbf{F}) = \mathbf{u}\}$$

The Moreau identity enables to characterize the solution of $E^i_{w,1}$

$$\mathbf{f} = \text{prox}_{\lambda R^i_{w,1}} + \lambda \text{prox}_{(R^i_{w,1})*/\lambda}(\mathbf{f}/\lambda) = \hat{\mathbf{u}} + \text{proj}_{B_{G_w}(\lambda)}(\mathbf{f})$$

which gives

$$\hat{\mathbf{u}} = \mathbf{f} - \text{proj}_{B_{G_w}(\lambda)}(\mathbf{f})$$

with the projection on $B_{G_w}(\lambda)$, a ball of radius λ for the norm G_w:

$$B_{G_w}(\lambda) = \{div_w \mathbf{F}, \mathbf{F} \in \mathcal{H}(\mathcal{E}), \|\mathbf{F}\|_\infty \leq \lambda\}$$

Convergence

From the characterization , we have

$$\mathbf{f} - \sum_{i=0}^{n} \mathbf{u}_i = \mathbf{v}_n = \text{proj}_{B_{G_w}(\lambda_n)}(\mathbf{v}_{n-1})$$

and

$$\|\mathbf{f} - \sum_{i=0}^{n} \mathbf{u_i}\|_{G_w} \leq \lambda_n$$

If the sequence (λ_n) is decreasing, then

$$\lim_{n\to+\infty} \lambda_n = 0 \implies \lim_{n\to+\infty} \|\mathbf{f} - \sum_{i=0}^{n} \mathbf{u_i}\|_{G_w} = 0 \implies \mathbf{f} = \sum_{i=0}^{+\infty} \mathbf{u}_i.$$

M. Hidane, O. Lezoray, A. Elmoataz, *Nonlinear Multilayered Representation of Graph-Signals*, Journal of Mathematical Imaging and Vision, Vol. 45(2), pp. 114-137, 2013.

Parameters

If λ_n is too small, few details are captured and a trivial decomposition is obtained:

$$\mathbf{f} = \bar{\mathbf{f}} + \mathbf{v}$$

One can show that there exist a critical value of λ above which any decomposition is trivial.
A decomposition is non trivial iff $\lambda \in]0, \|\mathbf{f} - \bar{\mathbf{f}}\|_{G_w}[$
With the parameters following a dyadic progression $\lambda_i = \lambda_{i-1}/2$, we have

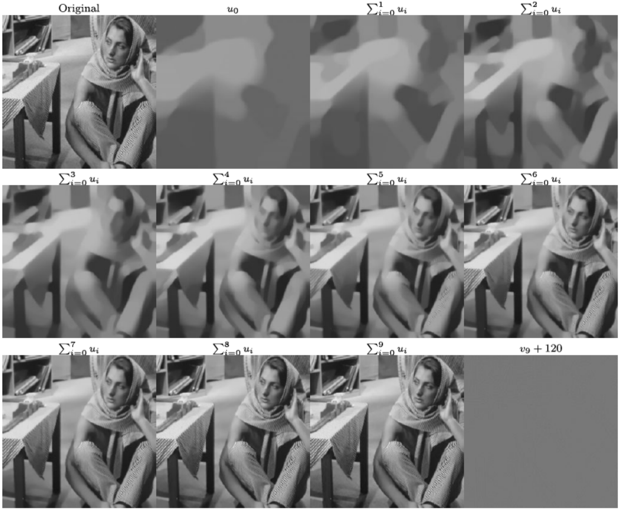
$$\frac{1}{2}\|\mathbf{f} - \bar{\mathbf{f}}\|_{G_w} \leq \lambda_0 < \|\mathbf{f} - \bar{\mathbf{f}}\|_{G_w}.$$

We choose to take the midpoint

$$\lambda_0 = \frac{3\|\mathbf{f} - \bar{\mathbf{f}}\|_{G_w}}{4}.$$

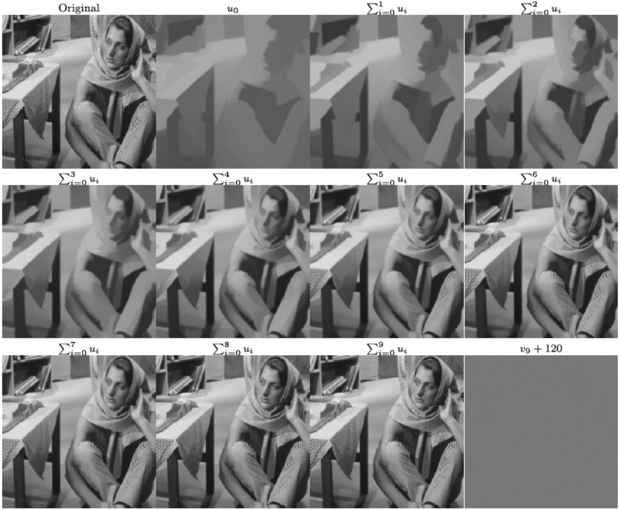
The Chambolle-Pock algorithm is used to compute the G_w norm.

Local Decomposition



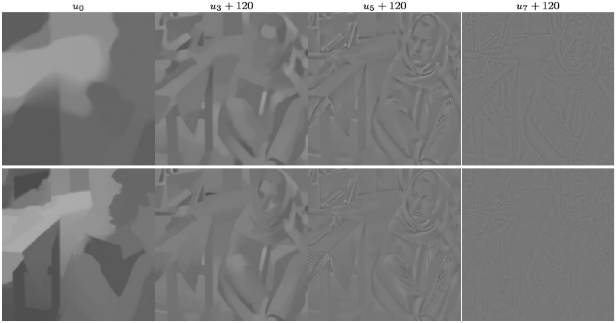
Local decomposition with an unweighted 4-grid graph $\mathcal{G}_1$, $\mathbf{F}_{\mathbf{0}}^{\mathbf{f^0}} = f^0$ (the classical TNV approach)

Nonlocal Decomposition



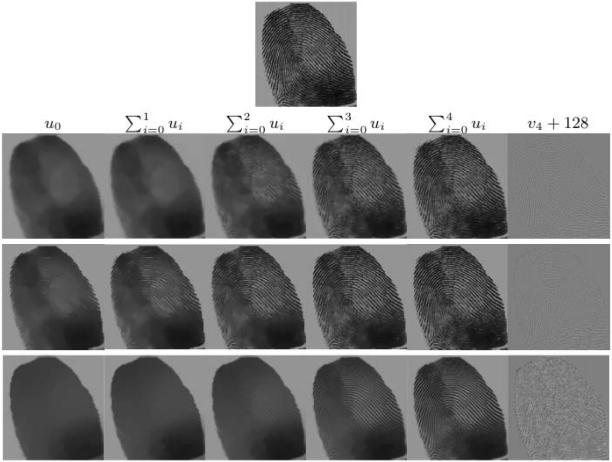
Nonlocal decomposition with a 10-NNG, $\mathbf{F_2^{f^0}}$.

Local versus Nonlocal Decomposition



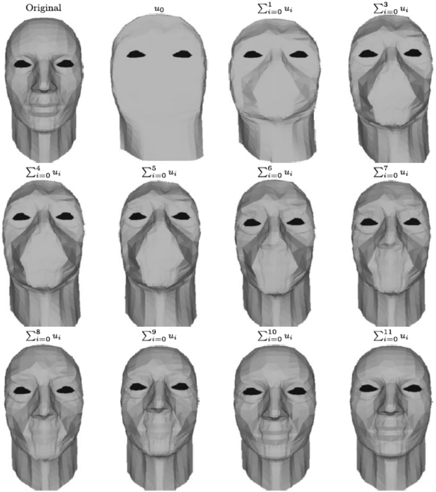
The layers extracted with the local (top) and nonlocal (approaches).

Local versus Nonlocal Decomposition



Top: 4-grid, $w = 1$; Middle: 8-grid, $w = exp()$; Bottom: 10-NNG, $\mathbf{F}_5^{\mathbf{f}^0}$

Mesh decomposition



Colored Mesh decomposition

Manipulating the decomposition

Aim

Attenuate or enhance details in a graph signal by applying coefficients to the extracted layers.

Original Image | BLF | WLS | Our approach (weighted 8-grid graph)

Removing layers

Original Image

Removing layers $\mathbf{u}_1$ to $\mathbf{u}_3$ removes acne

and $\mathbf{u}_6$ to $\mathbf{u}_9$ removes freckles

Mesh enhancement

Original Mesh

Enhanced Mesh

M. Hidane, O. Lezoray, A. Elmoataz, *Graph signal decomposition for multi-scale detail manipulation*, International Conference on Image Processing (IEEE), pp. 2041-2045, 2014. **Award finalist for the ICIP Best Paper Award.**

Low Quality Colored Mesh enhancement

Original Colored Mesh

Enhanced Mesh

High Quality Colored Mesh enhancement

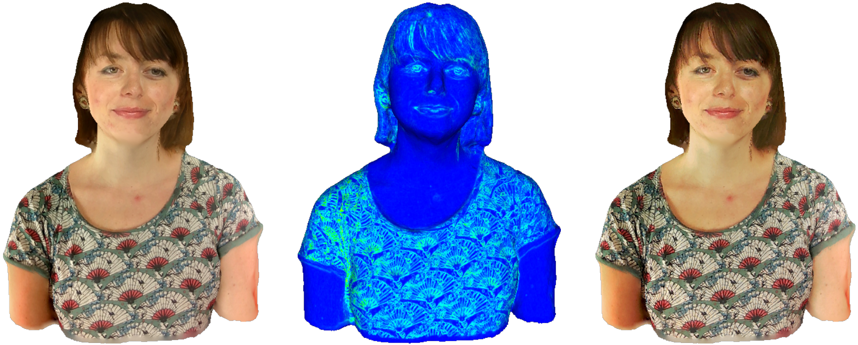
553053 vertices, 1105611 faces

Original scan

Enhanced scan

3D Selfie enhancement

3D selfie obtained with `http://reconstructme.net`



Original 3D selfie — 3D selfie structure mask — Enhanced 3D selfie

O. Lezoray, *3d colored mesh graph signals multi-layer enhancement*, ICASSP 2017, submitted.

1 Introduction

2 Graphs and difference operators

3 Construction of graphs - non locality

4 *p*-Laplacian nonlocal regularization on graphs

5 Multiscale hierarchical decomposition of graph signals

6 Adaptation of active contours on graphs

We consider a recent method proposed by Bresson and Chan to redefine the active contour model into a model which gives global minimizers

$$\underset{f(x)\in\{0,1\}}{\arg\min} \left\{ \int_\Omega ||\nabla f(x)||_1 dx + \lambda \int_\Omega g(f^0)(x) f(x) dx \right\}. \tag{26}$$

This can be adapted on graphs with PdEs:

$$\bar{f} \in \underset{f:\mathcal{V}\to\{0,1\}}{\operatorname{Arg\,min}} \left\{ \sum_{v_i\in\mathcal{V}} \|(\boldsymbol{\nabla}_{\mathbf{w}}\mathbf{f})(\mathbf{v_i})\|_p^p + \lambda \sum_{v_i\in\mathcal{V}} g(f^0)(v_i) f(v_i) \right\}, \tag{27}$$

where f is a labeling function and f^0 the signal on the graph.

X. Bresson and T.F. Chan, *Non-local unsupervised variational image segmentation models*, UCLA CAM Report 08-67, 2008.

Convex Relaxation

Previous problem (27) is non-convex and can be reformulated through a convex relaxation.

$$\hat{f} = \underset{f:\mathcal{V}\to[0,1]}{\arg\min} \left\{ \sum_{v_i\in\mathcal{V}} \|(\nabla_{\mathbf{w}}\mathbf{f})(\mathbf{v_i})\|_p^p + \lambda \sum_{v_i\in\mathcal{V}} g(f^0)(v_i)f(v_i) \right\}. \qquad (28)$$

- Global solution $\bar{f} : \mathcal{V} \to \{0,1\}$ is obtained by thresholding $\hat{f} : \mathcal{V} \to [0,1]$
- One has $\bar{f}(v_i) = \chi_{\mathcal{S}}(v_i)$, where $\mathcal{S} = \{v_i \in \mathcal{V} : \hat{f}(v_i) > t\}$ with $t \in [0,1]$
- For a given vertex, if $v_i \in \mathcal{A}$, then $\chi_{\mathcal{A}}(v_i) = 1$ and $\chi_{\mathcal{A}}(v_i) = 0$ otherwise
- ➲ First part of the energy (28) has to verify the co-area formula.

T.Chan, S.Esedoglu, and M.Nikolova,*Algorithms for Finding Global Minimizers of Image Segmentation and Denoising Models*, SIAM J. Appl. Math., vol. 66, no. 5, pp. 1632-1648, 2006.
O. Lézoray, A. Elmoataz, V.T. Ta, *Nonlocal PdEs on graphs for active contours models with applications to image segmentation and data clustering*, International Conference on Acoustics, Speech, and Signal Processing (IEEE), 2012.

Perimeters & Co-area formula on graphs

Co-area

- For $t \in \mathbb{R}$, let $\mathcal{A}_t = \{u \in \mathcal{V} : f(u) > t\}$.
- The co-area formula is verified for $p = 1$ since

$$Per_{w,1}(\mathcal{A}) = \int_{-\infty}^{\infty} Per_{w,1}(\mathcal{A}_t)dt$$

- The proof is direct since $|a - b| = \int_{-\infty}^{+\infty} |\chi_{\{a>t\}} - \chi_{\{b>t\}}|dt$.

We consider only the case of $p = 1$ since $R^a_{w,1}$ does verify the co-area formulae.

Perimeters

Given a sub-graph $\mathcal{A} \subset \mathcal{V}$, we can show that

$$R^a_{w,p}(\chi_{\mathcal{A}}) = \sum_{v_i \in \mathcal{V}} \|(\nabla_{\mathbf{w}} \chi_{\mathcal{A}})(\mathbf{v_i})\|_p^p = \text{vol}(\partial \mathcal{A}) = Per_{w,p}(\mathcal{A}) = \text{cut}(\mathcal{A}, \mathcal{A}^c)$$

O. Lézoray, A. Elmoataz, V.T. Ta, *Nonlocal PdEs on graphs for active contours models with applications to image segmentation and data clustering*, International Conference on Acoustics, Speech, and Signal Processing (IEEE), 2012.

Chan-Vese on graphs

We can directly express the discrete analogue on graphs of the CV model (a kind of regularized k-means), with

$$g(f^0(v_i)) = (\bar{c}_1 - f^0(v_i))^2 - (\bar{c}_2 - f^0(v_i))^2$$

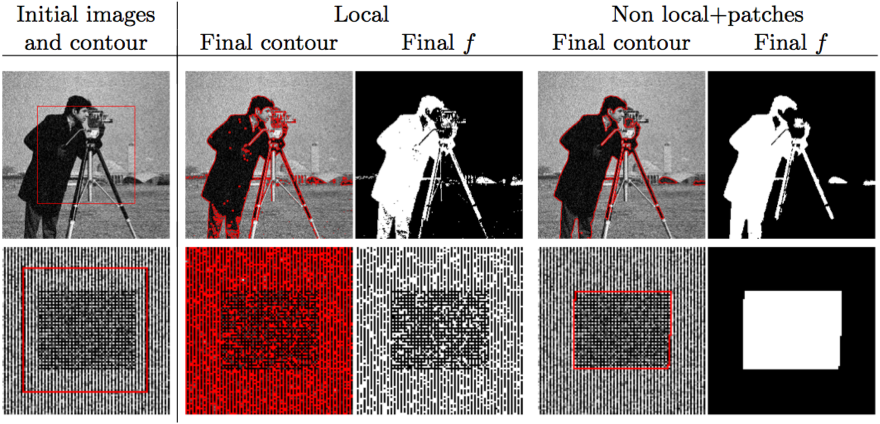
where $\bar{c}_1$ and $\bar{c}_2$ the average values inside and outside the object, then $\forall v_i \in V$:

Minimization

We use the Chambolle Pock algorithm with

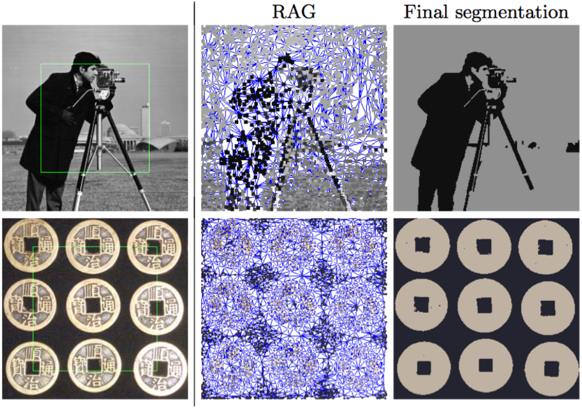
- $F = ||.||_1^1$
- $K = \mathbf{\nabla_w}$
- $G = \lambda\langle ., g(f^0)\rangle$

CV on images



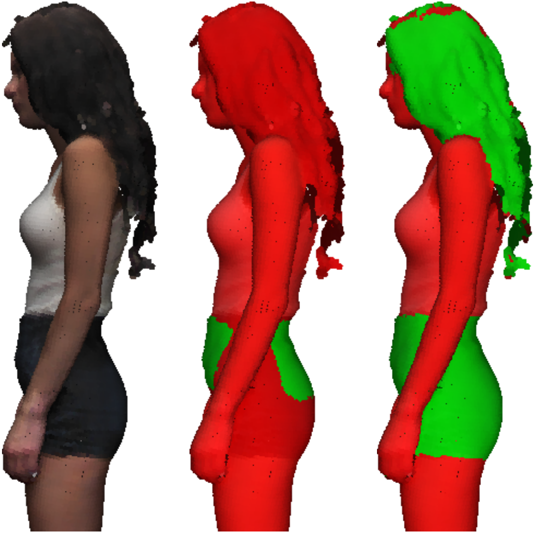
Segmentation result on local (4-adjacency graph with Gaussian weights computed on pixel values) and nonlocal (4-adjacency graph coupled with a 4-Nearest Neighbor graph selected in a 9×9 window and Gaussian weights computed on 3×3 patches) graphs.

CV on Region Adjacency Graphs

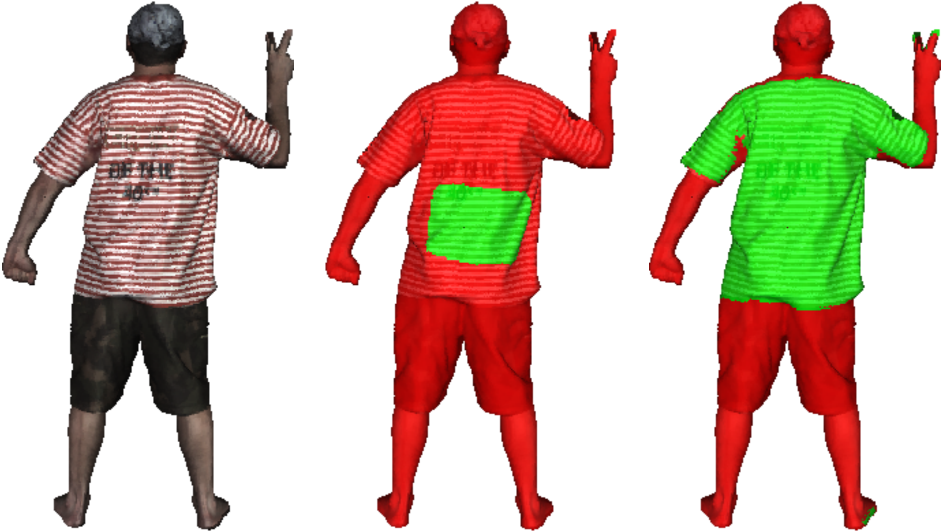


An initial contour, the considered graph (a super-pixel graph obtained from an over-segmentation with Gaussian weights on region mean values), and the obtained partition of the super-pixel graph.

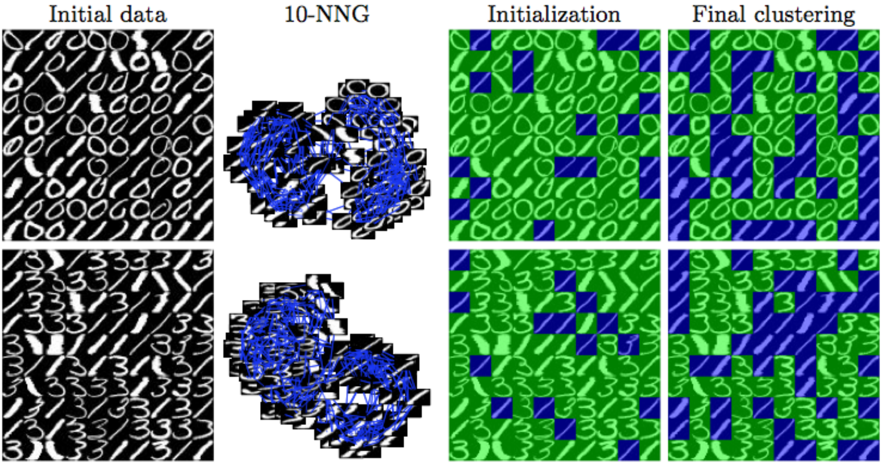
CV on Point clouds (Local)



CV on Point clouds (Nonlocal)



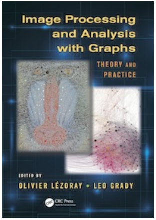
CV on Image Database Graphs



An image database with an initial random partition, the considered graph (a 10 nearest neighbors graph weighted with Gaussian weights on 16×16 vectors associated to the image of each vertex).

The End. Thanks.

Publications available at :
http://lezoray.users.greyc.fr



O. Lézoray and Leo Grady
Image Processing and Analysis with Graphs: Theory and Practice, CRC Press, July 2012.
https://lezoray.users.greyc.fr/IPAG/