Density based graph denoising for manifold learning

Yves Michels, Etienne Baudrier, Loïc Mazo, Mohamed Tajine

Density based graph denoising for manifold learning

Yves Michels, Étienne Baudrier, Loïc Mazo, Mohamed Tajine

Abstract—Processing high dimension data often makes use of a dimension reduction step. Indeed, high dimension data generally rely on a low dimension underlying structure. When the data are noisy, dimension reduction may fail because of shortcuts appearing on the graph catching the underlying structure. Our paper presents a method to suppress shortcuts in the underlying structure graph. The method is based on a skeleton graph that approximates the data and that is built using a data probability density estimation. This approximating graph is then used to select the edges of the underlying structure graph used in the dimension reduction. The proposed algorithm is tested on the capacity to suppress shortcuts and to conserve the underlying structure geodesic distance. Our method outperforms the state-of-the-art methods in the experiments on six 3D synthetic dataset and one tomographic dataset with different noise levels.

Index Terms—Shortcut detection, manifold learning, unsupervised learning, structure learning, neighborhood graph

1 Introduction

Processing high dimensional datasets is a significant challenge in machine learning, data visualization and parameter estimation. In many applications, the data lies on a low dimensional smooth subset embedded in high dimensional space. Dimension reduction can be used as a preliminary step to make subsequent algorithms more efficient [1]. When data lie in an affine subset, a linear dimension reduction can be used. For instance, Principal Component Analysis reduces the dimensionality by projecting the data onto the maximum variance linear subspace. This paper focuses on the case where the data are not in a linear subspace. Then, one has to turn to non-linear dimension reduction also known as manifold learning.

Manifold learning methods aim to extract the intrinsic low dimensional geometry of the data. These methods can be classified in two categories: Methods based on an *a priori* low dimensional map embedded in the high dimension space to match the dataset [2], [3], [4], [5]. Setting the low dimensional map implies to know *a minima* the topology of the manifold. When only the data point set is known, manifold learning methods based on the conservation of local information can reduce dimensionality without *a priori* on the topology of the manifold. This local information can be similarities as in Laplacian Eigenmap [6], the local linear structure as in Locally Linear Embedding [7] and Local Tangent Space Alignment [8], or distances as in Isomap [9], [10]. Local information is embedded in a neighborhood graph connecting only points that are close in the high dimension space [11]. When the dataset is too sparse or too noisy, the neighborhood graph connects points geodesically far on the manifold. If there are such shortcuts, the dimension reduction is biased and generally fails to reveal the low dimension manifold.

This shortcut issue is studied in the literature. In [12], [13], [14], the authors propose to construct the graph in two steps: first they construct a neighborhood graph and then they detect and remove shortcuts from a nearest neighbor graph. The simplest approach to detect shortcuts is to compute local statistics as the Jaccard index [15]. Given an edge, the Jaccard index measures the similarity of the neighborhood of its two vertices. Cukierski and Foran [12] propose to use the Edges Betweenness Centrality (EBC) to detect shortcuts. The EBC of an edge is the number of shortest paths connecting each pair of vertices that contain the given edge. By definition, shortcuts connect geodesically distant vertices, thus the EBC of the shortcuts tends to be higher than the EBC of the other edges. Glasher and Martinez [14] propose to remove a minimal set of edges to cut all the atomic cycles larger than a given threshold. Indeed, they show that a graph with shortcut connections necessarily contains large cycles. However, this method is not adapted to manifolds with holes and does not detect shortcuts when they are dispersed in the manifold. Detecting shortcuts is a challenging problem and our experiments, described in Section 4, show that existing methods generally fail when the noise level is high.

The scope of this paper is to construct a shortcut free, though highly connected, neighborhood graph on a noisy dataset of points from a low dimensional smooth manifold embedded in a high dimensional Euclidean space. At this aim, we present a method called Density based Graph Denoising (DGD). Our method is based on the construction of a graph that reveals the manifold. This graph is constructed regarding to the estimated density of the data point set.

The organization of the paper is as follows. In Section 2, a formal definition of a shortcut is given. Section 3 exposes the steps of the proposed algorithm. Then, Section 4 presents our experiments and the results. A conclusion and some perspectives end this paper in Section 5.

• *The authors are in ICube, Université de Strasbourg, CNRS (UMR 7357) ; 300 boulevard Sébastien Brant, CS 10413, 67412 Illkirch, France. E-mails: y.michels@unistra.fr, baudrier@unistra.fr, loic.mazo@unistra.fr, tajine@unistra.fr*

2 NOTION OF SHORTCUT

2.1 Geodesic semi-metric

Let m be a positive integer. In the whole paper, the topology on the space $\mathbb{R}^m$ is the usual topology associated with the Euclidean distance $\mathrm{d_E}$. Recall that a semi-metric is a symmetric function $d : (\mathbb{R}^m)^2 \mapsto \mathbb{R}_+$ such that $\forall\, x, y,\ \ d(x,y) = 0 \implies x = y$. A semi-metric d is a distance if it respects the triangle inequality: $\forall\ x, y, z\ \ d(x,z) \leq d(x,y) + d(y,z)$.

Let d be a semi-metric on $\mathbb{R}^m$ and S be a subset of $\mathbb{R}^m$. On the set S we consider two families of nested symmetric and irreflexive binary relations $\mathcal{N}^d_{S,\varepsilon}$ and $\mathcal{N}^d_{S,k}$. The parameter ε is continuous, *i.e.* $\varepsilon \in (0, +\infty)$, and, two points x, y of S are adjacent relatively to the relation $\mathcal{N}^d_{S,\varepsilon}$ if $d(x,y) \leq \varepsilon$. The parameter k is discrete, *i.e.* $k \in \mathbb{N}$, and two points x, y of S are adjacent relatively to the relation $\mathcal{N}^d_{S,k}$ if y is among the k nearest points of x and/or x is among the k nearest points of y. In the following, if there is no ambiguity in the context, we use the same notation $\mathcal{N}^d_\alpha$ for both $\mathcal{N}^d_{S,\varepsilon}$ and $\mathcal{N}^d_{S,k}$ without referring to the set S and letting α be a positive real number ε in the continuous case or a positive integer k in the discrete case. Then, the graph of the relation $\mathcal{N}^d_\alpha$ is denoted by G^d_α.

Let $\mathcal{M}$ be a Riemannian manifold of $\mathbb{R}^m$ whose geodesic distance is $\mathrm{d}_\mathcal{M}$ and whose intrinsic dimension is $l > 0$ (for more detail concerning Riemannian manifold see [16]). The geodesic distance in $\mathcal{M}$ is extended to $\mathbb{R}^m \times \mathbb{R}^m$ by a semi-metric as follows.

Definition 1 (Geodesic semi-metric)**.** *Let $\mathcal{M} \subset \mathbb{R}^m$ be a Riemannian manifold. For any $(x,y) \in \mathbb{R}^m \times \mathbb{R}^m$, we set*

$$\tilde{\mathrm{d}}_\mathcal{M}(x,y) = \min\Big(\mathrm{d_E}(x,x^*) + \mathrm{d}_\mathcal{M}(x^*,\ y^*) + \mathrm{d_E}(y^*,y) \mid x^* \in \underset{x_\mathcal{M}\in\mathcal{M}}{\operatorname{argmin}}\big(\mathrm{d_E}(x,x_\mathcal{M})\big)\ ,\ y^* \in \underset{y_\mathcal{M}\in\mathcal{M}}{\operatorname{argmin}}\big(\mathrm{d_E}(y,y_\mathcal{M})\big)\Big)\ .$$

The geodesic semi-metric, $\tilde{\mathrm{d}}_\mathcal{M}$ is not a distance in $\mathbb{R}^m$ because it does not necessarily respect the triangle inequality. The defined semi-metric is illustrated in Figure 1.

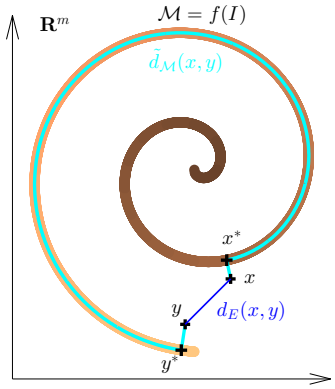


Fig. 1: Illustration of the geodesic measure.

2.2 Manifold shortcut

Let a finite subset Π of $\mathbb{R}^m$ be considered as a sampling of $\mathcal{M}$ with or without noise. The graphs of the relations $\mathcal{N}^{\mathrm{d_E}}_{\Pi,\alpha}$ and $\mathcal{N}^{\tilde{\mathrm{d}}_\mathcal{M}}_{\Pi,\alpha}$ are weighted by the corresponding distances: $w_{x,y} = \mathrm{d_E}(x,y)$ if $(x,y) \in \mathcal{N}^{\mathrm{d_E}}_\alpha$ and $w_{x,y} = \tilde{\mathrm{d}}_\mathcal{M}(x,y)$ if $(x,y) \in \mathcal{N}^{\tilde{\mathrm{d}}_\mathcal{M}}_\alpha$.

The graph $G^{\mathrm{d_E}}_\alpha$ may contain "shortcuts" which are edges that connect points that are not neighbors for the geodesic semi-metric.

Definition 2 (Shortcut relative to a manifold)**.** *Let $\beta > 0$. An edge of $G^{\mathrm{d_E}}_\alpha$ is a β-*shortcut relative to the manifold *$\mathcal{M}$ if it is not an edge of $G^{\tilde{\mathrm{d}}_\mathcal{M}}_\beta$.*

Figure 2 illustrates Definition 2. Observe that a β_1-shortcut of $G^{\mathrm{d_E}}_\alpha$ is a β_2-shortcut of $G^{\mathrm{d_E}}_\alpha$ whenever $\beta_2 < \beta_1$.

Unfortunately, the previous definition is not much useful since in practice the manifold, and therefore the semi-metric, is unknown. That is why we give thereafter another definition that will be used in the algorithm we propose.

Definition 3 (Shortcut relative to a semi-metric)**.** *Let d be a semi-metric on $\mathbb{R}^m$. Let $\beta > 0$. An edge (x,y) of $G^{\mathrm{d_E}}_\alpha$ is a β-*shortcut relative to the semi-metric *d if $d(x,y) > \beta$.*

Obviously, a β-shortcut relative to $\tilde{\mathrm{d}}_\mathcal{M}$ is a β-shortcut relative to $\mathcal{M}$, provided the ε-neighborhoods are used to define the graph $G^{\tilde{\mathrm{d}}_\mathcal{M}}_\beta$. Nevertheless, the latter definition will allow us to use a semi-metric computed to be close to the unknown geodesic semi-metric.

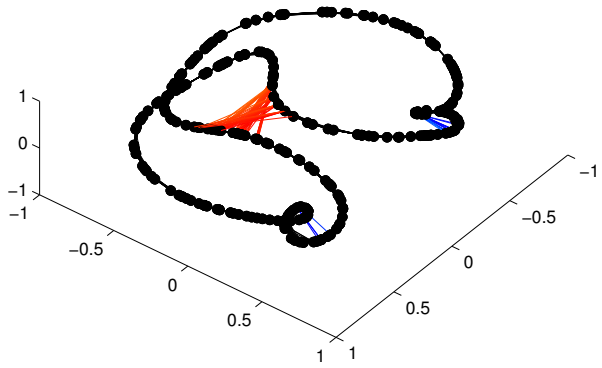


Fig. 2: A 300 sample point set on a 1-dimensional manifold $\mathcal{M}$ of $\mathbb{R}^3$. The $\mathcal{M}$-shortcuts are computed taking the union of nearest neighborhoods with $\alpha = 6$, $\beta = 15$. The blue edges involve small geodesic distance errors and the red ones involve large errors (color online).

3 SHORTCUT PRUNING ALGORITHM

Let $\mathcal{M}$ be a l-dimensional manifold of $\mathbb{R}^m$. Let Π be a finite subset of $\mathbb{R}^m$ sampled from $\mathcal{M}$ and noised by an additive noise. We assume that the probability density function of the noise is centered and varies inversely with the distance to the origin.

Let $\alpha, \beta > 0$. The aim of the algorithm exposed in this section is to remove from the neighborhood graph $G_\alpha^{\mathrm{d_E}}$ built on Π the β-shortcuts relative to the manifold $\mathcal{M}$. As $\mathcal{M}$ is unknown, we propose to replace the semi-metric $\tilde{\mathrm{d}}_{\mathcal{M}}$ by another semi-metric that varies directly with $\tilde{\mathrm{d}}_{\mathcal{M}}$, and to remove the β-shortcuts relative to this semi-metric. The new metric is build from a so called *skeleton-graph*, noted SkG, whose vertices and edges lie in the data high relative density areas (the vertices need not to be included in Π). Indeed, thanks to the assumptions on the probability density function of the noise, points lying in the higher density areas are close to the manifold $\mathcal{M}$ in probability. Then, a Voronoï diagramm allows to bind the data points to the vertices of the skeleton and to use the skeleton-graph geodesic distance as an approximation of the unknown $\tilde{\mathrm{d}}_{\mathcal{M}}$. The construction of SkG is detailed hereafter and illustrated in Figure 3.

Firstly, the density of the noisy manifold $\mathcal{M}$ is estimated from the sample Π (Subsection 3.1).

Then, the vertices of SkG are computed with the objective to sample as uniformly as possible the high density areas. For computational efficiency, the sample size, n_{sk}, is set to a small fraction of the data set Π. The sampling start from the output of a k-mean on Π and is then driven towards high density areas by minimizing a cost function which also favors spreading (Subsection 3.2).

Eventually, a connected neighborhood graph is constructed on the n_{sk} vertices by selecting the pairs of vertices that are close for the Euclidean distance and that can be linked by a straight segment lying in a relative high density region (Subsection 3.3). The reason to consider a relative density rather than an absolute density is to get a well-balanced set of edges linking the vertices, especially since we do not assume a uniform probability measure on the manifold (see Section 3.1).

Once, the graph SkG is built, a new metric $\mathrm{d_{SkG}}$ is defined on the graph $G_\alpha^{\mathrm{d_E}}$ such that $\mathrm{d_{SkG}}(x, y)$ is equal to the geodesic distance on SkG between the vertices of SkG closest to x and y. Finally, the β-shortcuts relative to this new metric are removed from the graph $G_\alpha^{\mathrm{d_E}}$ (Subsection 3.4).

3.1 Density estimation

We assume that the l-dimensional manifold $\mathcal{M}$ is the image of a compact and connected set of parameters $I \subset \mathbb{R}^l$ by a smooth injective function f from I to $\mathbb{R}^m$. We also assume a probability measure μ on I — which models the sampling process — together with an additive noise in $\mathbb{R}^m$. Then the distribution of the dataset in $\mathbb{R}^m$ is driven by the convolution ν of the image measure $f * \mu$ and the noise probability measure. Note that in absence of noise, $\nu = f * \mu$ and it has no density but one can estimate the measure on tiles of $\mathbb{R}^m$ (which amounts to an obvious count [17]).

The goal of the density estimation is to estimate ν from Π without *a priori* about the distribution. The estimated density is noted D_Π. The existing density estimation methods can be separated in two main approaches; parametric estimation and non-parametric estimation [18]. The parametric approach is based on a parametric function $g_\Phi \colon \mathbb{R}^m \to \mathbb{R}^+$, where Φ is a finite set of parameters. In the general case, where no *a priori* on the function g_Φ is known, we have to turn on non-parametric density estimation.

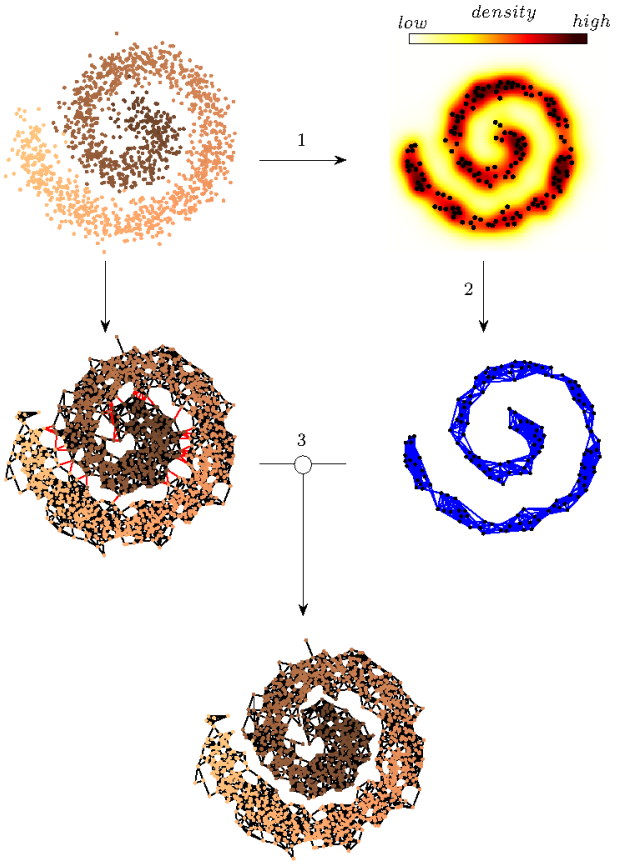


Fig. 3: Illustration of the proposed-method steps on a 2D 1500 point dataset from a spiral manifold perturbed with an additive white Gaussian noise whose standard deviation is $\sigma = 0.2$. The upper sub-figures represent from left to right the dataset and SkG points with the estimated density. The two middle sub-figures represent respectively the neighborhood graph where the detected shortcuts are marked in red and the SkG. The data graph denoised by DGD is shown in the lower sub-figure (color online).

Non-parametric density estimation can be gathered in two families [19]: histogram based estimations and Kernel Density Estimation (KDE). Histograms are used in the literature as a visualization tool and their use is limited to one or two dimensions. The second non-parametric approach is KDE described in the following paragraph.

Kernel Density Estimation: Given the finite dataset $\Pi = \{\pi_i\}_{i \in [\![1, n_p]\!]}$, the probability density is estimated by a convolution of the data modeled by a mixture of weighted Dirac's functions centered on each point and a fixed kernel function. The general form, D_Π, of the KDE is given below.

$$\forall x \in \mathbb{R}^m,\ D_\Pi(x) = \sum_{i=1}^{n_p} \frac{\omega_i}{\delta(x, \pi_i)^m}\, K\left(\frac{\pi_i - x}{\delta(x, \pi_i)}\right) ,$$

where $K(.)$ is the kernel function, $\{\omega_i\}$ are weights summing to 1, and δ is the window width. Several kernels have been proposed in the literature. Nevertheless, the kernel shape has a limited impact on the Integrated Mean Square Error (IMSE) regarding to the window width [18]. The kernel used in our experiment is the Gaussian kernel. However, proper selection of the kernel window width is

a critical step for KDE [20]. Therefore, three functions have been implemented and evaluated (see Appendix **??**), which leads us to choose the *sample point estimator* [21] for our experiment.

3.2 Skeleton-graph vertices computation

Initialization: A set S of n_{sk} points is computed from the data set Π with an adaptively constrained k-means algorithm [22] to ensure to start the optimization from a good summary of the data set (low intra-class variance and well-balanced class sizes).

Optimization: The objective function C is defined as follows:

$$\begin{aligned} C\colon \left(\mathbb{R}^m\right)^{n_{sk}} &\longrightarrow \mathbb{R} \\ (x_1, \cdots, x_{n_{sk}}) &\longmapsto \sum_{i=1}^{n_{sk}} \left(\lambda_1 \frac{\mathrm{d_E}(s_i, x_i)^2}{\Delta^2} \right. \\ &\quad + \lambda_2 \left(- \frac{D_\Pi(x_i)}{D_\Pi(s_i)} \right) \\ &\quad \left. + \lambda_3 \sum_{j \neq i} \frac{\Delta^2}{\mathrm{d_E}(s_j, x_i)^2} \right) , \end{aligned}$$

where $s_i \in S$, the constants λ_1, λ_2, λ_3 are positive weights summing to 1, Δ is a geometric scale factor added in order to homogenize the formula.

This cost function breaks down into the sum of three terms:

- $\frac{\mathrm{d_E}(s_i,x_i)^2}{\Delta^2}$ is a penalization for a high distance to the initial center of the cluster;
- $-\frac{D_\Pi(x_i)}{D_\Pi(s_i)}$ is a reward for high density;
- $\sum_{j\neq i} \frac{\Delta^2}{\mathrm{d_E}(s_j,x_i)^2}$ is a penalization for SkG vertices clustering.

The output of the optimization, V, which is the point set that minimizes the objective function C, is taken as the vertex set of SkG.

Once vertices computed, we construct the edges of SkG as described below.

3.3 Skeleton-graph edges determination

SkG is aimed to only connect close vertices for the Euclidean distance while being a connected graph. Moreover, the edges must pass through relative high density areas. The construction of SkG edges is made in several steps.

Step 1 The first step deals with the Euclidean distance $\mathrm{d_E}$: the complete graph built upon the vertex set V is weighted by the Euclidean distance between its vertices. Then, a minimum spanning tree is computed and enriched with the intersection of the n_v nearest neighbors (n_v-NN), with n_v a parameter discussed in Section 4. At the end of this first step, $G_0 = (V, E_0)$ is a connected graph where vertices are connected iff they are close for $\mathrm{d_E}$.

Step 2 The second step deals with the estimated density D_Π. In order to avoid edges passing far from the manifold, a density coefficient $e_{s,t}$ is assigned to each edge $\{s,t\} \in E_0$ so as to give priority to edges passing through high relative density regions. Ideally, we should compute the minimum density along the edge. To keep low computation times, in our implementation we simply calculate the density at the center of the edge and compare it to the mean of the densities at extremities of the edge:

$$e_{s,t} = D_\Pi\left(\frac{s+t}{2}\right) \times \frac{2}{D_\Pi(s) + D_\Pi(t)} \ .$$

Then, the graph G_0 is equipped with the weights $1/e_{s,t}$ and we perform the same operations as in Step 1: a minimum spanning tree and a k-NN. The choice of k is a trade-off between the graph connectivity and the shortcut probability (note that necessarily $k < n_v$). At the end of the second step, we obtain a connected graph $G_1 = (V, E_1)$ where vertices are connected by an edge iff they are close for $\mathrm{d_E}$ and the edge does not pass through a low density area.

Step 3 When testing the proposed method, our experiments showed that the connectivity of the graph G_1 is not sufficient. Nevertheless, it is unsafe to increase the parameter k in Step 2 since this can create high shortcut edges in the graph as shown in Figure 4(b) where the k-NN graph constructed with the best parameter – $k = 10$ – contains a cut and a shortcut. So, noting that the probability that $\tilde{\mathrm{d}}_{\mathcal{M}}(s,t)$ is greater than a threshold β on the one hand increases as the number of edges between s and t in G_1 increases and, on the other hand, decreases as the density coefficient $e_{s,t}$ increases, we define the following predicate.

$$\begin{aligned} P(s,t) = \big(D_{s,t} < \big(n_v - r(s,t)\big)/2\big) \ \vee \\ \big(D_{t,s} < \big(n_v - r(t,s)\big)/2\big) \ , \end{aligned}$$

where $D_{s,t}$ is the number of edges between the vertices s and t, $r(s,t)$ is the rank of the vertex t in the neighborhood of the vertex s ranked by edge weight and $\vee$ is the logic operator "or".
The edges set E of SkG is the union of the set $E1$ and the edges of E_0 that satisfy the predicate P. The result is SkG $= (V, E)$.

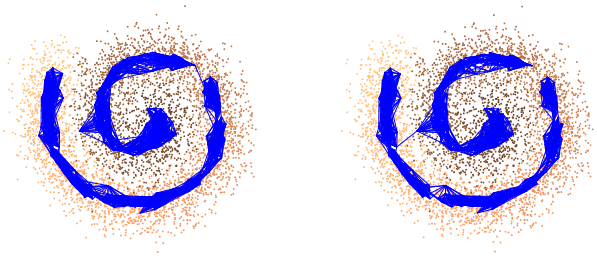
The SkG construction is illustrated in Figure 5.

3.4 Removing shortcuts in the data graph $G_\alpha^{\mathrm{d_E}}$

Let $s_1, \cdots, s_{n_{sk}}$ be the vertices of SkG. They induce a partition of the data space whose cells R_i are given by:

$$\begin{aligned} R_i = \{x \in \mathbb{R}^m \mid \forall j > i,\ \mathrm{d_E}(x, s_i) < \mathrm{d_E}(x, s_j) \text{ and} \\ \forall j < i,\ \mathrm{d_E}(x, s_i) \leq \mathrm{d_E}(x, s_j)\} \ . \end{aligned}$$

Each point π in the dataset Π belongs to exactly one cell. The *assignment function* $\sigma\colon |[1, n_p]| \to |[1, n_{sk}]|$ is defined such that $\pi_i \in R_{\sigma(i)}$. Then, we define the distance $\mathrm{d_{SkG}}$ on Π by $\mathrm{d_{SkG}}(\pi_i, \pi_j) = \mathrm{d_G}(s_{\sigma(i)}, s_{\sigma(j)}))$ where $\mathrm{d_G}$ is the unweighted graph distance on SkG. Then, according to Definition 3, an edge (π_i, π_j) of $G_\alpha^{\mathrm{d_E}}$ is detected as a β-shortcut relatively



(a) Skeleton graph: 2845 edges (b) 10-NN with weight $1/e_{s,t}$: 2426 edges

Fig. 4: SkG computed from 5000 points uniformly sampled from the Swiss Roll manifold, noised with white Gaussian noise with $\sigma = 0,28$. (a) SkG obtained by the three steps described in Section 3.3 with $n_v = 25$ and $k = 4$, (b) SkG without the third step but using a higher parameter k: $n_v = 25$ and $k = 10$.

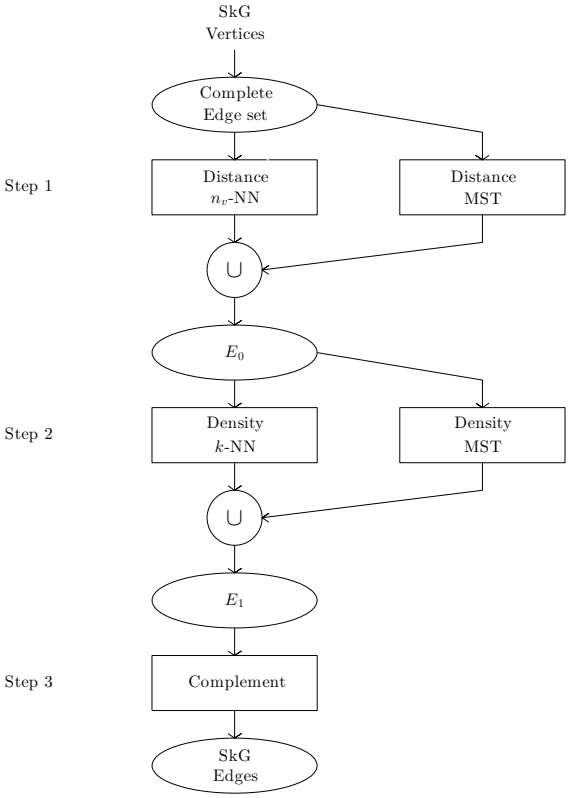


Fig. 5: SkG edge construction algorithm.

to $\mathrm{d_{SkG}}$ —and therefore pruned— if $\mathrm{d_{SkG}}(\pi_i, \pi_j) > \beta$. Algorithm 1 describes the detection process.

Up to the boundary, the subsets $R_1, \ldots R_{n_{sk}}$ can be seen as the Voronoi cells of the SkG vertex set V. Assuming that the frontier between two cells is weighted by their $\mathrm{d_{SkG}}$ distance, an edge is considered as a shortcut (and removed) when it crosses a frontier whose weight is higher than the fixed threshold β (see Figure 6 for an illustration).

Algorithm 1 Shortcut pruning

Input Data:
  Π; ▷ data point set
  SkG; ▷ skeleton graph
Input Parameters:
  n_v; ▷ maximum number of neighbor
  β; ▷ threshold for the shortcut detection
A = n_v-nearest neighbor graph(Π); ▷ Construct the neighborhood graph
Compute the assignment function σ;
D = graph distance matrix(SkG); ▷ graph distances used for the detection
for $\{\pi_i, \pi_j\} \in A$ **do**
  if $D_{\sigma(i),\sigma(j)} > \beta$ **then**
    remove $\{\pi_i, \pi_j\}$ from A ▷ remove detected shortcuts
  end if
end for
return A

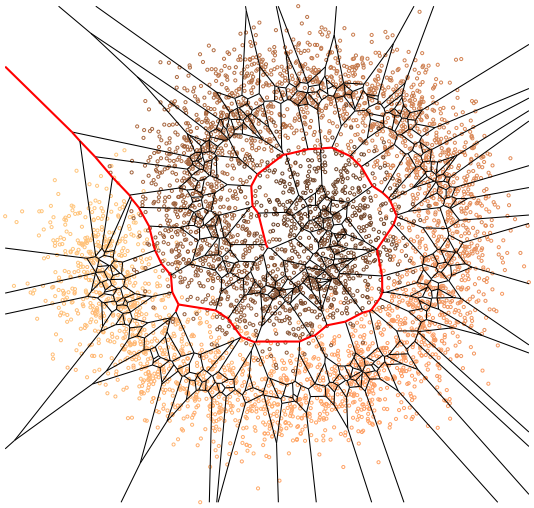


Fig. 6: Voronoi diagram on SkG vertices constructed on 5000 points uniformly sampled from a 2D spiral manifold, noised with white Gaussian noise with $\sigma = 0,28$. The frontiers whose weights (see text) are higher than 4 are marked in red.

4 Experiments

Our method is tested and compared on synthetic manifolds with a simple geometry (few curvature variations) and real world datasets. We evaluate the ability of the shortcut removal to recover the underlying structure geodesic distances and to cut as few edges as possible.

4.1 Graph pruning

The meta-parameters of the proposed method have been chosen experimentally. The same set of parameters is used for all the experiments.

- The maximum number of neighbors has been chosen to minimize the pairwise estimated geodesic distance error in noise free datasets. For sets of 5000 points from our synthetic manifolds, it has been fixed to $n_v = 25$.
- mputation times. The chosen number of vertices is $n_{sk} = 400$.
- The window parameter used in the kernel density estimation is $\lambda_l = 0.7$. It has been chosen to minimize the average integral mean square error of the estimated density of the noisy synthetic datasets.
- The parameters used for the computation of SkG points are $\lambda_1 = \lambda_2 = 1/3$ to gives the same weight to the 3 normalized terms of the cost function. It has been shown experimentally that these values gives good performances to move SkG vertices near to the manifold without creating clusters.
- The scale factor is given by

$$\Delta = \sqrt{\frac{1}{n_p}\sum_{i=1}^{n_p}\frac{1}{n_v}\sum_{\pi\in\mathcal{N}_{n_v}^{\mathrm{d_E}}(\pi_i)} \mathrm{d_E}(\pi_i,\pi)^2} \ .$$

Moreover, in order to observe shortcuts at comparable scales, we sized the tested synthetic manifolds such that the critical length — the radius of the smallest ball B whose centre is in the smooth manifold $\mathcal{M}$ and such that $\mathcal{M}\cap B$ has at least two connected components — has value 1. An illustration of the critical length is given in Figure 7.

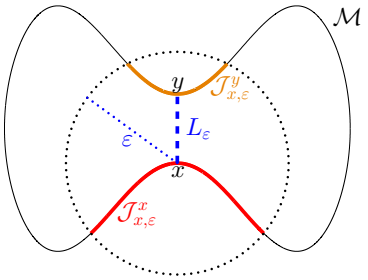


Fig. 7: Illustration of the critical length L_ε. The smallest balls with at least two connected components – here $\mathcal{J}^x_{x,\varepsilon}$ and $\mathcal{J}^y_{x,\varepsilon}$ – are centered in x or y and with radius $\varepsilon = L_\varepsilon$.

The shortcut detection methods are evaluated on the accuracy of the pairwise geodesic distances estimation. Indeed, the shortest-path based geodesic-distance estimation is highly impacted by the presence of shortcuts in the used graph. To measure the accuracy of the geodesic distance estimation, the matrix of pairwise geodesic distances is compared to reference distances. These reference distances are computed by a shortest path algorithm on the graph obtained by the $\mathcal{N}^{\tilde{\mathrm{d}}_{\mathcal{M}}}_{S,k}$ neighborhood, with $k = 25$, on the noise-free dataset. To be less sensitive to the distance fluctuations due to the noise, which is independent of the graph quality, the sets of geodesic distances are centered and normalized. The error is then given by the Frobenius distance between the computed and the reference normalized geodesic distance matrices:

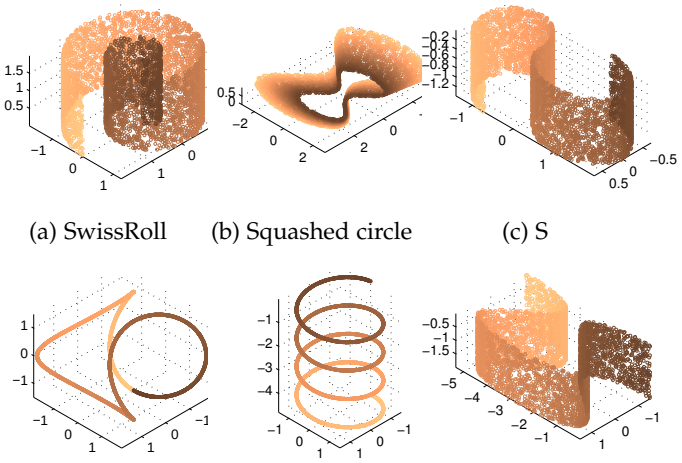
$$err = \frac{1}{n_p}\sqrt{\sum_{i=1}^{n_p}\sum_{j=1}^{n_p} e_{i,j}^2}$$

where $e_{i,j} = \frac{D_G(i,j)-\mathrm{E}(D_G)}{\sigma(D_G)} - \frac{D_{G_0}(i,j)-\mathrm{E}(D_{G_0})}{\sigma(D_{G_0})}$ and D_G is the matrix of the pairwise geodesic distances computed on the tested graph, D_{G_0} is the matrix of the pairwise reference geodesic distances, $\mathrm{E}(X)$ is the mean of X and $\sigma(X)$ is the standard deviation of X.

The datasets used for the experiments are composed of 5000 points uniformly distributed in the parameter space with additive white Gaussian noise. The evaluation is done on 6 synthetic manifolds and a tomographic dataset with 4 levels of noise. Each experiment is repeated 100 times with different data samplings and noise realizations.

Synthetic manifolds

The shortcut detection methods have been tested on 6 synthetic non-linear manifolds in the three-dimensional space with intrinsic dimension 1 or 2. The manifolds are defined in Appendix **??** and shown in Figure 8. The synthetic manifold sizes have been chosen to have the manifold critical lengths around 1, and the noise standard deviation belongs to $\{0, 0.15, 0.2, 0.25\}$.



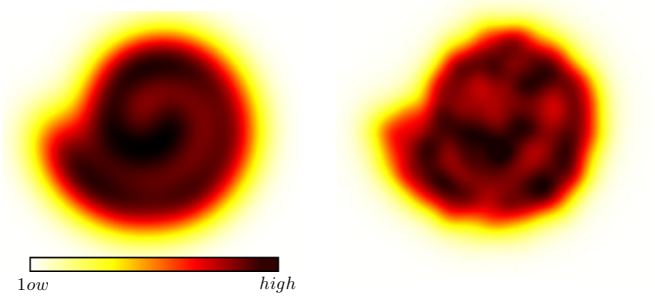
(a) SwissRoll (b) Squashed circle (c) S

(d) Tennis (e) Spring (f) Omega

Fig. 8: The 6 synthetic manifolds (without noise).

The critical length quantifies the difficulty of the shortcut detection. Our experiments on synthetic manifolds show that detecting shortcuts for a noise standard deviation over a quarter of the critical length is a difficult task. Moreover, it is plain that for a Gaussian noise, the density cannot reveal the manifold for a noise standard deviation above half of the critical length. The main reason is that for such a noise level, the areas on the manifold far for the geodesic distance and close for the Euclidean distance, have a higher density between the two distinct parts of the manifold than near each part. This implies that locally, the average number of

vertices and edges is higher between the distinct parts of the manifold than near each part. Therefore methods based on max flow/min cuts, on Jaccard index, on EBC or on density fail to detect shortcuts. Theoretical density and estimated density on 5000 points from the SwissRoll manifold noised with $\sigma = 0.4$, just lower than half of the critical length, are given in Figure 9. Note that the SwissRoll critical length is equal to 1.



(a) Theoretical density (b) Estimated density from 5000 points

Fig. 9: Density of the 2D SwissRoll manifold perturbed by a white Gaussian noise with $\sigma = 0.4$. Even if the manifold can be approximated by the maximums of the true density (a), it is not possible to find the manifold from the estimated density (b).

Tomographic manifold

We consider the problem of the parameter estimation from a dataset of random tomographic projections acquired from a 2D object. The set of projections lies in a closed smooth 1D manifold parametrized by the orientation, $\theta \in [0, 2\pi]$. The function f is the Radon transform of the planar object. It has been shown in [23] that non-linear dimension reduction can revealed the orientation of each projection. Generally the geometry of the data is embedded in the first principal components. As graph based non-linear dimension reduction relies on distance computation and as the distances are sensitive to the curse of dimensionality, a first linear dimension reduction is applied. For independent noise, the conserved signal variance can be controlled without additional *a priori*. In our example, 90% of the variance of the signal lies on the 5 principal components. Therefore, our experiments are led on the 5 principal components of our dataset. The noise levels are set to $\sigma \in \{0, 0.15, 0.25, 0.3\}$, where 0.3 is around a quarter of the critical length. Figure 10 shows a realization of a noisy tomographic dataset.

Results

Figures 11 and 12 give a comparison of the error obtained for each 25-Intersection of the Nearest Neighbors graph (IkNN) with different shortcut detection methods:

- Jaccard index thresholding with a threshold fixed to 0.4.
- EBC with the stopping criterion $\big(e\big(D_G(n) - D_G(n+1)\big) < 0.3\big) \vee (n = 15)$.
- Our method with the parameters given previously.
- IkNN which stands for undenoised graph.

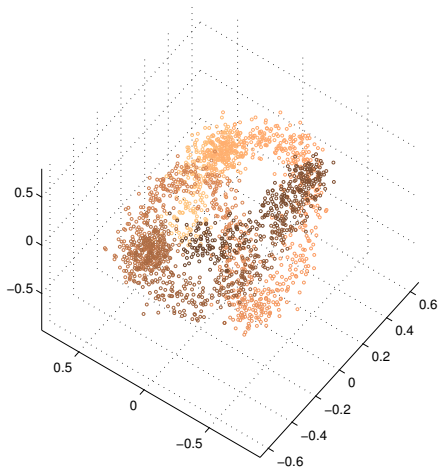


Fig. 10: First three principal components of a planar tomographic dataset composed of 5000 noisy 1D projections in dimension 125 ($\sigma = 0.3$).

The ground truth has been obtained by removing edges connecting vertices where the geodesic measure is higher than 20% of the highest geodesic measure in the dataset. As the experiment is composed of 9600 graph constructions, the results in Figures 11 and 12 are presented as smoothed histograms of the estimated geodesic distance errors. The peaks correspond to sets of constructed graphs with similar geodesic distances estimation error.

Let us give some general comments. For each manifold, in the noise free case ($\sigma = 0$), there is just one peak on the left, close to zero (no error). Therefore, the different methods do not introduced bias in the construction, except the Jaccard index that may cut the graph in the one dimensional manifolds (Tennis and Spring). In presence of noise ($\sigma > 0$), high error value peaks appear on the right, corresponding to the presence of large graph shortcuts and/or graph cuts. Indeed, the presence of large shortcuts and/or cuts — even a few of them — may introduce biases in a large proportion of estimated geodesic distances. Therefore, the higher the peak close to zero, the better the method. All the tested methods improve the estimated geodesic distances as their peaks are on the left of the IkNN ones (the average errors are smaller after using a denoising method than before).

Now, we give a detailed analysis. It can be seen in Figure 11 that DGD is the most appropriate to detect large shortcuts for the SwissRool, the Tennis and the Spring manifold, and gives similar results as EBC for the Squashed circle and the Omega manifold. Because of the high curvature of the S manifold, EBC denoising obtained better results than DGD on the geodesic distances estimation.

For the noise level $\sigma = 0.3$ in the tomographic datasets, presented in Figure 12, our method clearly detects the large shortcuts in 77% of cases. In comparison, EBC detects the large shortcuts in 64% of cases (given by the number of graph construction error in the left peak). In addition, it can be seen in Table 1 that DGD is more selective than the other tested methods in the sense that the number of removed

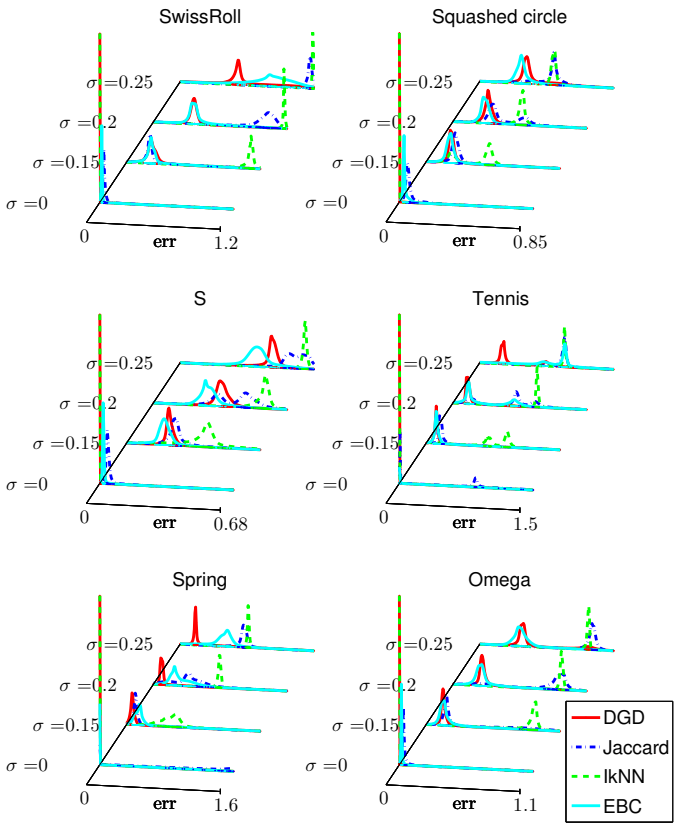


Fig. 11: Graph construction error for the 6 synthetic manifolds with 4 levels of noise, $\sigma \in \{0, 0.15, 0.2, 0.25\}$.

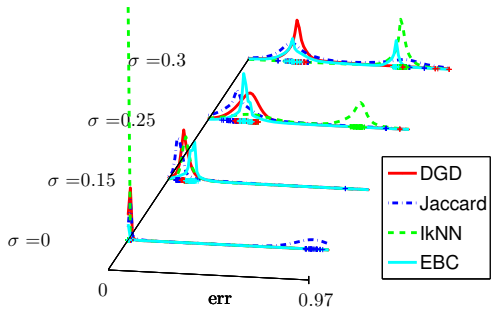


Fig. 12: Graph construction error for the tomographic dataset with 4 levels of noise, $\sigma \in \{0, 0.15, 0.25, 0.3\}$.

edges is smaller than the number of edges removed by the other denoising methods for all the 100 experiments.

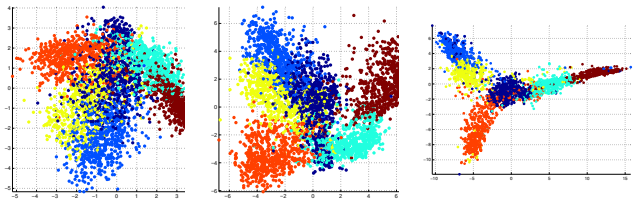
σ	Ground Truth	DGD	EBC	Jaccard
0	*0*	**0**	1.1 $.10^{-2}$	8.15
0.15	*1.3* $.10^{-2}$	**1.5** $.10^{-2}$	1.5	17.8
0.2	*0.136*	**0.153**	4.04	21.1
0.25	*0.459*	**0.468**	8.1	24.2

TABLE 1: Percentage of suppressed edges regarding the number of edges in the original k-INN graph with $k = 25$.

As a conclusion of these tests, it can be said that DGD gives results at least as good as the other tested methods while suppressing less edges than them.

4.2 Clustering

Our method can also be used for the classification. In this case, the steps for keeping the connectivity should be removed. Graph based clustering methods can be composed on a neighborhood-graph dimension reduction followed by a clustering. Dimension reduction based on the graph denoised by our method can improve the clustering. To illustrate this assertion, we compare the dimension reduction obtained by Isomap after performing our shortcut detection algorithm, the original Isomap algorithm and a linear dimension reduction (PCA) taken as a reference. The comparison is performed on the US postal dataset which is composed of 3906 gray level images with a resolution of 16 by 16 pixels, each image containing one hand written digit. To avoid issues due to a high density variation on the manifold, the experiment was done on the digits '2' to '7'. The reduction to two dimensions by the tested methods is given in Figure 13. It can be seen that the different digits (representing by different colors) are more separated when our shortcut detection step is done.



(a) PCA (b) Isomap (c) DGD

Fig. 13: Comparison of 2 dimension reductions on the digits '2' to '7' from the US postal dataset (color online).

4.3 Complexity

The asymptotic number of operations of the proposed algorithm is in $O(n_{sk}^2.n_v)$ due to the Djikstra's shortest path algorithm that is used twice. When the number of SkG points in the dataset is under 1000, the more time-consuming steps are the k-means and the gradient optimization with $O(n_{sk}.n_p.d.i)$ operations. The number of iterations for both k-mean and the gradient descent is in the order of hundreds.

A comparison of calculation times is presented in Table 2. The two measurements are the time spent to construct and to denoise the graph and the time spent to reduce the dimension, including graph construction, graph denoising, and dimension reduction. The dimension reduction is done by the Isomap method, using Djikstra's shortest path algorithm. The algorithms are implemented in Matlab and compared on a quad core i7-3770 @ 3.40GHz.

The Jaccard index based detection has the best time performance (it increases the graph computation time by 23%). However it has a poor detection performance regarding to EBC and our proposed method. The state-of-the-art EBC is slower than the proposed method to detect the shortcuts. In addition, EBC time performance is highly dependent on the noise level while our method is not.

Method	IkNN	Jaccard	EBC	DGD
Graph (s)	9.3 ±0.4	11.1 ±0.4	984 ±726	83.6 ±5.9
DR (s)	163.4 ±9.6	162.1 ±10.2	987 ±727	247.3 ±13.7

TABLE 2: Calculation times spent to construct the graph and to reduce dimensionality. The dataset used contained 5000 points from our synthetic manifolds.

5 Conclusion

A new method, so called Density based Graph Denoising (DGD), is introduced in this paper to detect shortcuts in the dataset neighborhood graph. The data underlying structure geometry is revealed by a skeleton graph that lies in high density areas. The use of a skeleton graph allows us to remove a minimal set of edges that does not follow the geodesic structure of the data. Our experiments show that DGD makes graph based non-linear dimension-reduction algorithms more robust to noise. The calculation time is several times lower than the state-of-the-art shortcut detection methods and does not depend on the noise level. Even if the data does not lie in a smooth manifold as in clustering problems, DGD can be used as a first processing step since it is not time consuming regarding to the complex high dimensional processing.

The density estimation is a key point of the method and future works will focus on the use of a priori information on the manifold so as to make the density estimation more robust in the case of high noise.

Acknowledgments

This work was partially supported by the *Agence Nationale de la Recherche* through contract ANR-14-CE27-0012-01.

References

[1] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, 2013. [Online]. Available: http://arxiv.org/abs/1206.5538

[2] T. Kohonen, "The self-organizing map," *Proc. IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[3] B. Kègl, A. Krzyzàk, T. Lindner, and K. Zeger, "Learning and Design of Principal Curves," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 0407, no. 3, pp. 281–297, 1997.

[4] B. Kégl and A. Krzyzak, "Piecewise linear skeletonization using principal curves," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 1, pp. 59–74, 2002.

[5] C. M. Bishop, M. Svensen, and C. K. I. Williams, "GTM: the generative topographic mapping," *Neural Comput.*, vol. 10, no. 1, pp. 215–234, 1998.

[6] M. Belkin and P. Niyrogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," 2002.

[7] D. L. Donoho and C. Grimes, "Hessian eigenmaps: locally linear embedding techniques for high-dimensional data." *Proc. Natl. Acad. Sci. U. S. A.*, vol. 100, no. 10, pp. 5591–5596, 2003.

[8] Z. Zhang and H. Zha, "Nonlinear Dimension Reduction via Local Tangent Space Alignment," *4th Int. Conf. IDEAL 2003*, vol. 2690, pp. 477–481, 2003.

[9] J. B. Tenenbaum, V. D. Silva, and J. C. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction," *Science (80-.).*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[10] M. Law and a.K. Jain, "Incremental nonlinear dimensionality reduction by manifold learning." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 3, pp. 377–391, 2006. [Online]. Available: http://ieeexplore.ieee.org/articleDetails.jsp?arnumber=1580483

[11] T. Lin and H. Zha, "Riemannian manifold learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 5, pp. 796–809, 2008.

[12] W. J. Cukierski and D. J. Foran, "Using betweenness centrality to identify manifold shortcuts," in *Proc. - IEEE Int. Conf. Data Min. Work. ICDM Work. 2008*, 2008, pp. 949–958.

[13] E. Brevdo and P. J. Ramadge, "Bridge Detection and Robust Geodesics Estimation via Random Walks," in *Conf. Acoust. Speech Signal Process.*, 2010.

[14] M. Gashler and T. Martinez, "Robust Manifold Learning With CycleCut," *Conn. Sci.*, vol. 24, no. 01, pp. 57–69, 2012.

[15] A. Singer and H. Wu, "Two-Dimensional Tomography from Noisy Projections Taken at Unknown Random Directions," *SIAM J. Imaging Sci.*, vol. 6, no. 1, pp. 136–175, 2013. [Online]. Available: http://epubs.siam.org/doi/abs/10.1137/090764657

[16] J. M. Lee, *Riemannian Manifolds: An Introduction to Curvature*. Springer-Verlag New York, Inc. (Graduate texts in mathematics ; 176), 1997.

[17] B. Sriperumbudur, "On the optimal estimation of probability measures in weak and strong topologies," *Bernoulli*, vol. 22, no. 3, pp. 1839–1893, 08 2016. [Online]. Available: http://dx.doi.org/10.3150/15-BEJ713

[18] A. Izenman, *Modern Multivariate Statistical Techniques*, G. Casella, S. Fienberg, and I. Olkin, Eds. Springer-Verlag New York, 2008.

[19] D. W. Scott and S. R. Sain, *Multidimensional Density Estimation*. Elsevier Masson SAS, 2005, vol. 24. [Online]. Available: http://dx.doi.org/10.1016/S0169-7161(04)24009-3

[20] D. Comaniciu, "An algorithm for data-driven bandwidth selection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 281–288, 2003.

[21] M. C. Jones, "Variable Kernel Density Estimates and Variable Kernel Density Estimates," *Aust. J. Stat.*, vol. 32, no. 3, pp. 361–371, 1990.

[22] Y. Xu, J. Wu, C.-c. Yin, and Y. Mao, "Unsupervised cryo-EM data clustering through adaptively constrained K-means algorithm," *ArXiv e-prints*, pp. 1–36, 2016.

[23] R. R. Coifman, Y. Shkolnisky, F. J. Sigworth, and A. Singer, "Graph Laplacian tomography from unknown random projections." *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1891–9, 2008. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/18784036

Yves Michels Received the Master of Science degree in Optics, Photonics, Signals and Images from École Centrale Marseille, France, in 2015. Is currently preparing a PhD in Signal, Image, automatic, robotics and Remote sensing from the University of Strasbourg. His research interest cover image processing, manifold learning, and tomography. He especially focuses on Cryo-Electron microscopy for deformable macro molecules.

Étienne Baudrier Étienne Baudrier studied mathematics at University Paris Sud, France and received his PhD in 2005 in Image Processing at the University of Reims, France. He is Assistant Professor since 2008 at the Univeristy of Strasbourg, France. His main research interests are digital geometry and inverse problems.

Loïc Mazo Loic Mazo studied Mathematics at the École Normale Supérieure de Cachan (Paris, France) and received the Agrégation of mathematics in 1978. He began to study Computer Science at the Université de Strasbourg in 2006 and received his Ph.D. degree in 2011. Since September 2012, he has been Assistant Professor of Computer Science at the University of Strasbourg. His scientific interests include discrete mathematics and tomography.

Mohamed Tajine Mohamed Tajine studied pure mathematics and computer science. He obtained his Ph.D. in 1992, specializing in logic and complexity theory. Between 1992 and 1999, he was an Assistant Professor at the Universit Louis Pasteur of Strasbourg, where he conducted research on combinatorics, discrete geometry, fractal geometry, image synthesis, neural networks and proof theory. In 1999, he obtained his Habilitation diploma. Since September 1999, he has been full Professor of Computer Science at university of Strasbourg. Since 1997, he is the head of the research group in discrete geometry of the Laboratoire ICube UMR 7357-CNRS. His scientific interests include discrete geometry, imaging theory, logic and complexity theory and neural networks.